



ELSEVIER

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp



A finite elements method to solve the Bloch–Torrey equation applied to diffusion magnetic resonance imaging



Dang Van Nguyen^{a,c}, Jing-Rebecca Li^{a,c,*}, Denis Grebenkov^b, Denis Le Bihan^c

^a INRIA Saclay, Equipe DEFI, CMAP, Ecole Polytechnique, Route de Saclay, 91128 Palaiseau Cedex, France

^b Laboratoire de Physique de la Matière Condensée, CNRS, Ecole Polytechnique, 91128 Palaiseau Cedex, France

^c NeuroSpin, Bat145, Point Courrier 156, CEA Saclay Center, 91191 Gif-sur-Yvette Cedex, France

ARTICLE INFO

Article history:

Received 28 July 2013

Received in revised form 23 December 2013

Accepted 6 January 2014

Available online 21 January 2014

Keywords:

Bloch–Torrey equation

Diffusion magnetic resonance imaging

Finite elements

RKC

Pseudo-periodic

Double-node

Interface problem

ABSTRACT

The complex transverse water proton magnetization subject to diffusion-encoding magnetic field gradient pulses in a heterogeneous medium can be modeled by the multiple compartment Bloch–Torrey partial differential equation (PDE). In addition, steady-state Laplace PDEs can be formulated to produce the homogenized diffusion tensor that describes the diffusion characteristics of the medium in the long time limit. In spatial domains that model biological tissues at the cellular level, these two types of PDEs have to be completed with permeability conditions on the cellular interfaces. To solve these PDEs, we implemented a finite elements method that allows jumps in the solution at the cell interfaces by using double nodes. Using a transformation of the Bloch–Torrey PDE we reduced oscillations in the searched-for solution and simplified the implementation of the boundary conditions. The spatial discretization was then coupled to the adaptive explicit Runge–Kutta–Chebyshev time-stepping method. Our proposed method is second order accurate in space and second order accurate in time. We implemented this method on the FEniCS C++ platform and show time and spatial convergence results. Finally, this method is applied to study some relevant questions in diffusion MRI.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

Biological tissue is a heterogeneous medium, consisting of cells of various sizes and shapes distributed in the extra-cellular space. The cells are separated from each other and from the extra-cellular space by the cell membranes. Diffusion magnetic resonance imaging (dMRI) is an imaging modality that uses magnetic field gradient pulses in order to access the diffusion characteristics of water molecules over a time period on the order of tens of milliseconds (see a recent review in [1]).

While there have been numerous works on the analysis of the dMRI signal under simplifying assumptions (on the geometry, membrane permeability, pulse sequence), see, e.g., [2–8], in this paper, we focus on a more complete model of the water proton magnetization called the multiple compartment Bloch–Torrey partial differential equation (PDE), that allows the inclusion of general tissue geometries, permeable cell membranes, and arbitrary diffusion-encoding gradient sequences. This numerical model is a generalization of the Bloch–Torrey equation [9] to heterogeneous domains [5,10] and it models the complex transverse water proton magnetization subject to diffusion-encoding magnetic field gradient pulses. The dMRI signal is given as the integral of the magnetization.

* Corresponding author at: INRIA Saclay, Equipe DEFI, CMAP, Ecole Polytechnique, Route de Saclay, 91128 Palaiseau Cedex, France.

E-mail address: jingrebecca.li@inria.fr (J.-R. Li).

Some previous works that solved the *diffusion equation* to obtain the dMRI signal in the narrow gradient pulse limit (the duration of the pulses is small compared to the measured diffusion time) are [11,12,8], where the spatial discretization is finite elements. To account for the short gradient pulses, the magnetization is pre- and post-multiplied by a spatially-dependent complex factor. The initial condition to the PDE is either the delta function distribution (to obtain the diffusion propagator) or a complex-valued initial distribution describing the magnetization just after the first applied gradient pulse. The PDE solved is the pure diffusion equation (obtained when setting the gradient to zero in the Bloch–Torrey equation), and the simulation starts after the application of the first gradient pulse and ends before the application of the next gradient pulse.

The focus of the work in [11,12,8] is on simulating diffraction patterns of restrictive (or lowly permeable) porous systems, to determine pore size, for example. To produce such diffraction patterns, high gradient amplitudes and long diffusion times (as long as 1 second) are used to probe restrictive geometries. In [8] a second order implicit time-stepping method called the generalized α method [13], which was developed for dissipating high frequencies, was used. Because of the choice of this implicit time-stepping method, the matrix solve at each time step involves the stiffness matrix, whose condition number increases as $O(h^{-2})$, where h is the spatial discretization size. There was also an early work on solving the Bloch–Torrey equation where the pulse duration is not short [14] where the computational domain is one dimensional (restricted diffusion between parallel plates).

The focus of this current paper is not porous systems. Rather, it is biological tissue dMRI, where the diffusion time is much shorter, on the order of tens of milliseconds, and the gradient amplitudes are moderate. Numerical solutions of the multiple compartment Bloch–Torrey PDE for general gradient sequences (no narrow pulse restriction) were reported in [15–18], in which the finite difference method on a Cartesian grid was coupled to the explicit Forward Euler time-stepping method, resulting in first order accuracy in space and time. We found that an *explicit* and *adaptive* second order convergent time-stepping method called the Runge–Kutta–Chebyshev (RKC) method [19] is a much better choice for our application. The RKC method was especially formulated for diffusive PDEs (of which the Bloch–Torrey PDE is an example) to allow for much larger time steps than alternative explicit time-stepping methods such as the Forward Euler method. Explicit stepping methods have an advantage over implicit methods in that the solution of linear systems with the stiffness matrix is not required. Instead, linear systems solutions involve only the mass matrix, whose condition number is $O(1)$, and hence the linear systems are easy to solve. The RKC method is adaptive, in that it allows error control on the ODE solution and adapts the time step size to satisfy the error tolerance during the course of the simulation.

Now we summarize some important requirements of a numerical code for the simulation of biological tissue dMRI:

1. Allows arbitrary diffusion-encoding pulse shapes and sequences, including, for example, pulses that are square (idealized pulsed-gradient spin echo (PGSE) [20]), trapezoid (more realistic PGSE with non-zero rise time), oscillating sine and cosine [21,22], or a yet-determined shape to be optimized [23].
2. Allows generally-shaped cell membranes that are permeable to water passage.
3. Allows the periodic extension of the computational domain so that water can enter and exit the computational domain in a physically reasonable manner, as was done in [16,18].
4. Efficient for large-scale simulations in two and three dimensions.

We tried to satisfy the above requirements by the following choices:

1. The complex-valued Bloch–Torrey PDE (not the diffusion equation) is solved for arbitrary pulse shapes and sequences.
2. Linear finite elements discretization is used to allow generally-shaped compartment interfaces (modeling cell membranes).
3. Additional degrees of freedom are added on the compartment interfaces to allow permeability conditions on generally-shaped interfaces.
4. The periodic extension of the computational domain is an allowed option. To impose this condition the Bloch–Torrey PDE is transformed so that the boundary condition becomes more computationally efficient to implement.
5. The time-stepping method is chosen to be the explicit and adaptive Runge–Kutta–Chebyshev (RKC) method [19].

The combination of linear finite elements and the RKC method makes our approach second order accurate in space and time. For an efficient implementation of finite elements we chose to base our code on the FEniCS Finite Elements platform [24]. The Bloch–Torrey PDE has several unconventional features that cause implementation issues for a standard PDE platform such as FEniCS. We describe these issues and how we resolved them. First, we allowed jumps in the finite elements solution at the compartment interfaces by implementing double-nodes at the interfaces. Second, the pseudo-periodic boundary conditions resulting from the periodic extension of the computational domain are reduced to standard periodic boundary conditions by transforming the Bloch–Torrey PDE, as in [18]. We note, however, that in [18], the discretized PDE using centered finite difference did not take into account first order terms. To obtain second order convergence in space, we had to include all the appropriate first order terms in our discretization. Third, we reformulated the Bloch–Torrey PDE so that the real and imaginary parts of the magnetization are decoupled to allow the solution of two systems of half the number of unknowns compared to a naive implementation. We show accuracy and timing results for our method and use our code to simulate and gain insight into the diffusion characteristics of some complex geometries.

In addition to the complex-valued Bloch–Torrey PDE, we also solved steady-state (real-valued) Laplace PDEs that produce the homogenized diffusion tensor that describes the diffusion in a heterogeneous medium in the long time limit. We used the formulation where the computational domain is assumed to be periodically extended to infinity in all three coordinate directions. Using the same finite elements discretization as for the time-dependent Bloch–Torrey PDE, we will show the convergence of the apparent diffusion coefficient of the time-dependent problem to the result produced by the steady-state problem.

The paper is organized as follows. In Section 2, we introduce two types of PDEs occurring in diffusion MRI: the time-dependent Bloch–Torrey PDE and the steady-state Laplace equations that lead to the homogenized diffusion tensor. In Section 3, we explain our numerical method, including the double-node formulation to allow jumps in the finite elements solution on the compartment interfaces, the transformation of the PDE to replace pseudo-periodic by standard periodic boundary conditions, the decoupling of the real and imaginary parts, as well as the coupling of the finite elements discretization to the RKC time integration method. In Section 4 we briefly describe the implementation of the proposed method on the FEniCS C++ platform as well as the use of the mesh generator Salome [25]. In Section 5, we show accuracy and timing results for this method. In Section 6 we study the diffusion characteristics of some complex domains applicable to biological imaging. Section 7 contains our conclusions.

2. PDEs of diffusion MRI

The effect of water diffusion in biological tissue on the water proton magnetization under the influence of magnetic field gradient pulses can be modeled by a multiple compartment Bloch–Torrey PDE [9,5]. In the long time limit, the diffusion in a heterogeneous domain can be simply described by the homogenized diffusion tensor. Under the assumption of the infinite periodic extension of the domain, the homogenized diffusion tensor can be obtained by solving three steady-state Laplace PDEs. In this section we describe these two types of PDEs.

2.1. Multiple compartments Bloch–Torrey PDE

First, we define a simplified geometrical model of tissue, made up of an extra-cellular space, Ω^0 , and M non-overlapping biological cells defined by open sets: $\Omega^l \in \mathbb{R}^d$, $l = 1, \dots, M$, where $\Omega^l \cap \Omega^n = \emptyset$ for $l \neq n$, and d is the space dimension (typically $d = 2$ or $d = 3$). We also allow the possibility of adding a membrane compartment around each biological cell. In this case, there would be additional M membrane compartments: Ω^l , $l = M + 2, \dots, 2M + 1$. If the membrane compartments are not included in the tissue model, then they are approximated by the appropriate interface conditions between the cell compartments and the extra-cellular space. We denote the interface between Ω^l and Ω^n by $\Gamma^{ln} = \overline{\Omega^l} \cap \overline{\Omega^n}$. When we do not need to distinguish between the individual cells, we will group all the cell interiors into one intra-cellular compartment Ω^c , and all the membranes into one membrane compartment Ω^m , and the extra-cellular space will be denoted by Ω^e .

Ideally, the total computational volume $\bigcup_{l=1}^M \overline{\Omega^l}$ would be on the scale of the diffusion MRI resolution, usually on the order of one millimeter. However, due to the fact that cell features are on the scale of microns, typically a small portion of the tissue contributing to the signal in an imaging pixel is simulated. We denote this portion of tissue by C for the computational domain. Usually, $C = [-L/2, L/2]^d$ is a box and contains a “representative volume” of the tissue in the voxel under study.

The boundary condition to impose on ∂C can be homogeneous Neumann if the support of the initial data is far enough away from ∂C . The buffer zone needed between the support of the initial data and ∂C grows with the diffusion distance, which, in a homogeneous medium, is $O(\sqrt{Dt})$, where D is the diffusion coefficient, and t is time. Thus, the needed buffer zone is larger when the simulated diffusion time is longer.

Another choice of the boundary condition comes from the assumption that C is periodically repeated in all three coordinate directions. In this case, there would be no need for a buffer zone between the initial data and ∂C . Water will be allowed to enter and leave C . This is the choice made in [16,18] and we make the same choice here when simulating tissue geometries that contain the extra-cellular space.

To simplify the notation, we will assume that any parts of biological cells that are outside of C will be removed so that $C = \bigcup \overline{\Omega^l}$. In addition, we will define $\Gamma = \bigcup \Gamma^{ln} \setminus \partial C$ to be the union of the interfaces minus the boundary of C . We note that to study the dMRI signal inside an impermeable connected domain, there is no need for a computational box. In that case, we can impose homogeneous Neumann boundary conditions on the boundary of the connected domain.

In diffusion MRI, a time-varying linear spatial magnetic field gradient is applied to the tissue sample to encode water diffusion. Denoting the time profile of the diffusion-encoding magnetic field gradient by $f(t)$, its linear spatial dependence by $\mathcal{G}(\mathbf{r}) = \mathbf{g} \cdot \mathbf{r}$, where the vector \mathbf{g} contains the amplitude and direction information of the magnetic field gradient, the water proton magnetization $M(\mathbf{r}, t)$ satisfies the Bloch–Torrey PDE:

$$\frac{\partial}{\partial t} M(\mathbf{r}, t) = -I\gamma f(t)\mathcal{G}(\mathbf{r})M(\mathbf{r}, t) + \nabla \cdot (\mathbf{D}(\mathbf{r})\nabla M(\mathbf{r}, t)), \quad \mathbf{r} \in \bigcup \Omega^l, \tag{1}$$

where $\gamma = 2.67513 \times 10^8 \text{ rad s}^{-1} \text{ T}^{-1}$ is the gyromagnetic ratio of the water proton, I is the imaginary unit, $\mathbf{D}(\mathbf{r})$ is the (possibly discontinuous) intrinsic diffusion tensor. See Fig. 1(a) for an illustration of a piecewise continuous $\mathbf{D}(\mathbf{r})$ in three

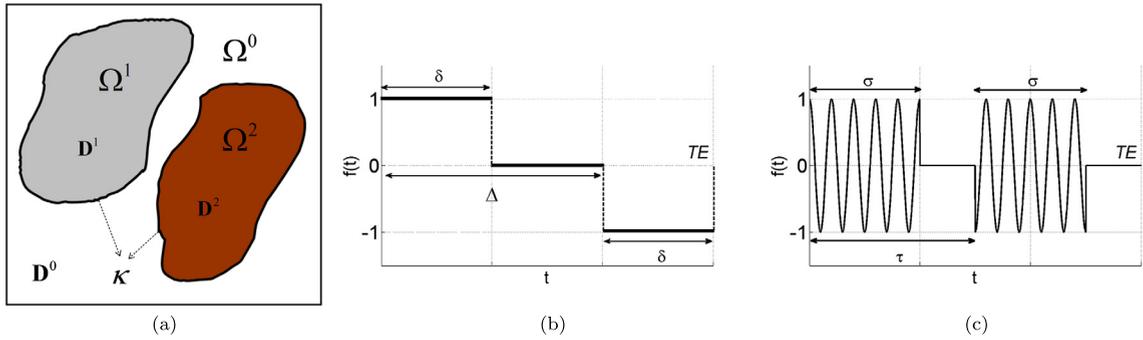


Fig. 1. (a) A multiple compartment domain for the Bloch–Torrey PDE; (b) a PGSE sequence; (c) a cos-OGSE sequence with $t_1 = 0$.

compartments. The magnetization $M(\mathbf{r}, t)$ is a function of position \mathbf{r} and time t , and depends on the diffusion gradient vector \mathbf{g} and the time profile $f(t)$.

There are two commonly used time profiles (diffusion-encoding sequences):

1. The pulsed-gradient spin echo (PGSE) [20] sequence, with two rectangular pulses of duration δ , separated by a time interval $\Delta - \delta$, for which the profile $f(t)$ is

$$f(t) = \begin{cases} 1, & t_1 \leq t \leq t_1 + \delta, \\ -1, & t_1 + \Delta < t \leq t_1 + \Delta + \delta, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where t_1 is the starting time of the first gradient pulse with $t_1 + \Delta > T_E/2$, T_E is the echo time at which the signal is measured (see Fig. 1(b)).

2. The oscillating gradient spin echo (OGSE) sequence [21,22] was introduced to reach short diffusion times. An OGSE sequence usually consists of two oscillating pulses of duration σ , each containing n periods, hence the frequency is $\omega = n\frac{2\pi}{\sigma}$, separated by a time interval $\tau - \sigma$ (see Fig. 1(c)). For a cosine OGSE, the profile $f(t)$ is

$$f(t) = \begin{cases} \cos(n\frac{2\pi}{\sigma}t), & t_1 < t \leq t_1 + \sigma, \\ -\cos(n\frac{2\pi}{\sigma}(t - \tau)), & \tau + t_1 < t \leq t_1 + \tau + \sigma, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where $\tau = T_E/2$.

The PDE in (1) needs to be supplemented by interface conditions at the interfaces Γ^{ln} , and by boundary conditions on ∂C . Let M^k and \mathbf{D}^k be restrictions of M and \mathbf{D} onto the k th compartment that take on the appropriate limiting values in case M and \mathbf{D} are discontinuous, the two interface conditions are the flux continuity:

$$\mathbf{D}^l(\mathbf{r})\nabla M^l(\mathbf{r}, t) \cdot \mathbf{n}^l = -\mathbf{D}^n(\mathbf{r})\nabla M^n(\mathbf{r}, t) \cdot \mathbf{n}^n, \quad \mathbf{r} \in \Gamma^{ln}, \forall l, n, \quad (4)$$

and a condition that incorporates a permeability coefficient κ^{ln} across Γ^{ln} :

$$\mathbf{D}^l(\mathbf{r})\nabla M^l(\mathbf{r}, t) \cdot \mathbf{n}^l = \kappa^{ln}(M^n(\mathbf{r}, t) - M^l(\mathbf{r}, t)), \quad \mathbf{r} \in \Gamma^{ln}, \forall l, n, \quad (5)$$

where \mathbf{n}^k is a normal vector pointing outward from Ω^k ($k = l, n$). If the permeability coefficient is the same at all the interfaces, then we will simply use the notation κ .

In the limit $\kappa^{ln} = \infty$, Eq. (5) reduces to the simple continuity condition on $M(\mathbf{r}, t)$:

$$M^l(\mathbf{r}, t) = M^n(\mathbf{r}, t).$$

Following [16], we extend C by periodic copies of itself to handle the diffusion of water molecules close to the boundary of C . According to [16], the two boundary conditions on ∂C are:

$$M(\mathbf{r}, t)|_{r_i=-L/2} = M(\mathbf{r}, t)|_{r_i=L/2}e^{i\theta_i(t)}, \quad i = 1, \dots, d, \quad (6)$$

$$\mathbf{D}(\mathbf{r})\frac{\partial M(\mathbf{r}, t)}{\partial r_i}\Big|_{r_i=-L/2} = \mathbf{D}(\mathbf{r})\frac{\partial M(\mathbf{r}, t)}{\partial r_i}\Big|_{r_i=L/2}e^{i\theta_i(t)}, \quad i = 1, \dots, d, \quad (7)$$

where $\mathbf{r} = (r_1, \dots, r_d)$,

$$\theta_i(t) := \gamma g_i L \int_0^t f(s) ds, \tag{8}$$

and $\mathbf{g} = (g_1, \dots, g_d)$. The PDE in Eq. (1) also needs an initial condition:

$$M^l(\mathbf{r}, 0) = \rho^l, \quad \mathbf{r} \in \Omega^l, \quad \forall l, \tag{9}$$

where ρ^l is the spin density in Ω^l .

The complete multiple compartments Bloch–Torrey PDE problem to be solved involves the PDE Eq. (1), the two interface conditions Eq. (4)–(5), the two boundary conditions Eq. (6)–(7), and the initial condition Eq. (9).

2.2. Analytical solution in a homogeneous domain for the PGSE sequence

For the PGSE sequence, there exists an analytical solution of the Bloch–Torrey PDE [26]. On the computational box $C = [-L/2, L/2]^d$, if the initial condition is the Dirac delta distribution at $\mathbf{x}_0 = (x_1^0, \dots, x_d^0) \in C$, with the gradient vector $\mathbf{g} = (g_1, \dots, g_d)$ and constant diffusion coefficient D , then

$$M(x_1, \dots, x_d, t) = \sum_{n_1=-\infty}^{\infty} \dots \sum_{n_d=-\infty}^{\infty} \prod_{i=1}^d \tilde{M}(x_i, x_i^0 + n_i L, g_i, t),$$

where,

- if $0 < t \leq \delta$:

$$\tilde{M}(x, x_0, g, t) = \frac{1}{2\sqrt{\pi t D}} \exp\left(\frac{-I}{2} t g \gamma (x_0 + x) - \frac{t^4 D^2 g^2 \gamma^2 + 3(x - x_0)^2}{12 D t}\right),$$

- if $\delta \leq t \leq \Delta$:

$$\begin{aligned} \tilde{M}(x, x_0, g, t) &= \frac{1}{2\sqrt{\pi D t}} \exp\left(-\frac{D g^2 \gamma^2 \delta^3 (-3\delta + 4t)}{12 t}\right) \\ &\times \exp\left(\frac{-I g \gamma \delta ((2t - \delta)x_0 + \delta x)}{2 t} - \frac{(-x_0 + x)^2}{4 D t}\right) \end{aligned}$$

- if $\Delta \leq t \leq \Delta + \delta$:

$$\begin{aligned} \tilde{M}(x, x_0, g, t) &= \frac{1}{2\sqrt{\pi D t}} \exp\left(\frac{I g \gamma (-\delta^2 + t^2 - \Delta^2)(x + x_0)}{2 t} - \frac{(x - x_0)^2}{4 D t}\right) \\ &\times \exp\left(\frac{-I g \gamma (-\delta^2 - \Delta^2 + t\delta + t\Delta)(x_0)}{t} + \frac{D g^2 \gamma^2 (\delta^2 + \Delta^2)^2}{4 t}\right) \\ &\times \exp\left(-\frac{1}{12} D g^2 \gamma^2 t^3 + \frac{1}{2} D g^2 \gamma^2 (\delta^2 + \Delta^2) t - \frac{D g^2 \gamma^2 (\delta^3 + 2\Delta^3 + 3\delta^2 \Delta)}{3}\right), \end{aligned}$$

- if $t > \Delta + \delta$:

$$\begin{aligned} \tilde{M}(x, x_0, g, t) &= \frac{1}{2\sqrt{\pi t D}} \exp\left(\frac{D g^2 \gamma^2 \delta^3 (-\delta + 2\Delta)}{3 t} + \frac{I g \gamma (x - x_0) \Delta \delta}{t}\right) \\ &\times \exp\left(-\frac{D g^2 \gamma^2 \delta^2 (-\delta + 3\Delta)(t - \Delta - \delta)}{3 t} - \frac{(x - x_0)^2}{4 t D}\right). \end{aligned}$$

is a solution of Eq. (1) and also satisfies the pseudo-periodic boundary conditions. We will use it in later sections as a reference solution.

2.3. DMRI signal

The dMRI signal is measured at echo time $t = T_E > \Delta + \delta$ for PGSE and $T_E > 2\sigma$ for OGSE. This signal is the integral of $M(\mathbf{r}, T_E)$:

$$S(\mathbf{g}) := \frac{\int_{\mathbf{r} \in C} M(\mathbf{r}, T_E) d\mathbf{r}}{\sum_l \rho^l \text{vol}(\Omega^l)}, \tag{10}$$

where we normalized the signal to 1 at $\mathbf{g} = 0$. We have explicitly included the dependence of the signal on the magnetic field direction and strength even though we removed it from the magnetization M to shorten the notation.

Now we describe some quantities derived from the signal that are important for MR physicists and medical researchers. In a dMRI experiment, the pulse sequence (time profile $f(t)$) is usually fixed, while \mathbf{g} is varied in amplitude (and possibly also in direction). When \mathbf{g} varies only in amplitude (while staying in the same direction), $S(\mathbf{g})$ is plotted against a quantity called the b -value. The b -value depends on \mathbf{g} and $f(t)$ and is defined as

$$b(\mathbf{g}) = \gamma^2 \|\mathbf{g}\|^2 \int_0^{T_E} du \left(\int_0^u f(s) ds \right)^2.$$

For PGSE, the b -value is [20]:

$$b(\mathbf{g}, \delta, \Delta) = \gamma^2 \|\mathbf{g}\|^2 \delta^2 (\Delta - \delta/3). \tag{11}$$

For the cosine OGSE with integer number of periods n in each of the two durations σ , the corresponding b -value is [16]:

$$b(\mathbf{g}, \sigma) = \gamma^2 \|\mathbf{g}\|^2 \frac{\sigma^3}{4n^2\pi^2} = \gamma^2 \|\mathbf{g}\|^2 \frac{\sigma}{\omega^2}. \tag{12}$$

The reason for these definitions is that in a homogeneous medium, the signal attenuation is e^{-Db} , where D is the intrinsic diffusion coefficient.

To analyze the signal attenuation as a function of the b -values for a chosen pulse sequence, we will change the independent variable \mathbf{g} of the signal to b and compute the first derivative of the logarithm of the signal attenuation curve with respect to the b -value at $b = 0$:

$$D^A := - \frac{\partial}{\partial b} \log \frac{S(b)}{S(0)} \Big|_{b=0}, \tag{13}$$

where the D^A , in the terminology used by physicists and medical researchers, is the ‘‘Apparent Diffusion Coefficient’’, and gives an indication of the root mean squared distance traveled by water molecules in the gradient direction $\mathbf{g}/\|\mathbf{g}\|$, averaged over all starting positions. Unless $\delta \ll \Delta$, the notion of diffusion time and diffusion distance, though useful, is not rigorously defined. But we follow the terminology from physics and call $\Delta - \delta/3$ the ‘‘diffusion time’’ for the PGSE sequence. The ‘‘diffusion distance’’ is then given by $\sqrt{2D^A(\Delta - \delta/3)}$. When $\delta \ll \Delta$, these definitions are rigorous. (For the OGSE sequence, the ‘‘diffusion time’’ is even harder to define and we do not use this notion for the OGSE sequence in this paper.) We numerically compute the D^A by a polynomial fit of $\log \frac{S(b)}{S(0)}$. In Section 6 we will study some properties of the D^A in complex domains, including its dependence on the gradient direction, $\mathbf{g}/\|\mathbf{g}\|$, and on Δ , the duration between pulses of the PGSE sequence.

2.4. Steady-state Laplace PDE for the homogenized diffusion tensor

In the long time limit, supposing an infinite periodic extension of the computational domain C , the diffusion characteristics can be described by an effective diffusion tensor [27]. In the presence of permeable interfaces, the homogenized diffusion tensor \mathbf{D}^{hom} can be obtained by solving the following d steady-state Laplace equations for $W_i(\mathbf{r})$, $i = 1, \dots, d$, over C [28]:

$$\nabla \cdot (\mathbf{D}(\mathbf{r}) \nabla W_i(\mathbf{r})) = 0, \quad \mathbf{r} \in \bigcup \Omega^l, \tag{14}$$

with the same interface conditions in Eqs. (4)–(5) as for the Bloch–Torrey equation, and two simpler boundary conditions on ∂C :

$$W_i(\mathbf{r})|_{r_k=-L/2} = W_i(\mathbf{r})|_{r_k=L/2} - \delta_{i,k}L, \quad k = 1, \dots, d, \tag{15}$$

$$\mathbf{D}(\mathbf{r}) \frac{\partial}{\partial r_k} W_i(\mathbf{r}) \Big|_{r_k=-L/2} = \mathbf{D}(\mathbf{r}) \frac{\partial}{\partial r_k} W_i(\mathbf{r}) \Big|_{r_k=L/2}, \quad k = 1, \dots, d, \tag{16}$$

where $\delta_{i,k} = 1$ for $k = i$, and 0 otherwise. The problem to be solved consists of Eqs. (14), (4)–(5), (15), (16), for $i = 1, \dots, d$. The entries of the homogenized tensor \mathbf{D}^{hom} are then given by:

$$\{\mathbf{D}^{\text{hom}}\}_{i,k} = \int_C \mathbf{D}(\mathbf{r}) \nabla W_i(\mathbf{r}) \cdot \mathbf{e}_k \, d\mathbf{r}, \quad i, k = 1, \dots, d,$$

where \mathbf{e}_k is the unit vector in the k th direction. We expect that

$$D^A \rightarrow \frac{\mathbf{g}^T}{\|\mathbf{g}\|} \mathbf{D}^{\text{hom}} \frac{\mathbf{g}}{\|\mathbf{g}\|}.$$

3. Method

In this section we describe our method to solve the Bloch–Torrey PDE (1), (4)–(5), (6)–(7), (9) and the steady-state Laplace PDEs (14), (4)–(5), (15), (16).

The standard Galerkin formulation for the Bloch–Torrey PDE in the weak form is

$$\frac{\partial}{\partial t} \int_{\Omega^l} Mv \, d\mathbf{r} = -I\gamma f(t) \int_{\Omega^l} \mathcal{G}Mv \, d\mathbf{r} - \int_{\Omega^l} \mathbf{D}\nabla M \cdot \nabla v \, d\mathbf{r} + \int_{\partial\Omega^l \cap \Gamma} \mathbf{D}\nabla M \cdot \mathbf{n}^l v \, ds + \int_{\partial\Omega^l \cap \partial C} \mathbf{D}\nabla M \cdot \mathbf{n}^l v \, ds, \tag{17}$$

for each compartment Ω^l , where \mathbf{n}^l is the outward pointing normal and v is a test function. We separated the two surface integrals into one involving the interface conditions and the other involving the boundary conditions and we will describe how to enforce them in the following sections. Similarly, the weak form for the steady-state Laplace PDE is

$$\int_{\Omega^l} \mathbf{D}\nabla W \cdot \nabla v \, d\mathbf{r} - \int_{\partial\Omega^l \cap \Gamma} \mathbf{D}\nabla W \cdot \mathbf{n}^l v \, ds - \int_{\partial\Omega^l \cap \partial C} \mathbf{D}\nabla W \cdot \mathbf{n}^l v \, ds = 0. \tag{18}$$

We will use linear elements and write our code using the finite elements platform FEniCS [24].

There are some numerical issues in the spatial discretization that will be addressed in this section. They concern several non-standard aspects of these two types of PDEs that require special handling when using a general finite elements platform like FEniCS.

3.1. Interface conditions: allowing jumps in the finite elements solution

Standard finite elements discretization enforces that the solution is continuous across elements. Discontinuous Galerkin discretization allows the solution to be fully discontinuous across all the elements [24]. In our case, the solution is continuous in each compartment Ω^l , and possibly discontinuous on the compartment interfaces $\Gamma = \bigcup \Gamma^{ln}$. For this reason, the discontinuous Galerkin discretization, which would have approximately double the number of nodes as a finite element discretization (for linear elements), is not efficient and we do not use it. Instead, we keep the finite elements formulation in order to use the matrix assembly routines in FEniCS. To do so, we need to find a way to incorporate jumps in the solution on the interface Γ while still keeping the solution ‘continuous’, at least formally.

To achieve this goal, we looped through the finite elements mesh and repeated nodes that lie on the interfaces Γ and created elements of zero volume there. We call these additional elements *interface elements*: an element consists of d distinct vertices, each repeated once. Standard finite elements are triangles in 2D and tetrahedrons in 3D. Interface elements are “fake” elements that are segments in 2D and triangles in 3D. In this way, the solution is formally continuous across the interface elements, but it is physically discontinuous because the interface elements have zero volume. We will then associate a local stiffness matrix to the interface elements even though this stiffness matrix represents a surface integral and not a volume integral.

We explain this discretization in detail for the steady-state Laplace PDE in Eq. (18). We write the solution as $W(\mathbf{r}) = \sum_{i=1}^N \xi_i \varphi_i(\mathbf{r})$, where $\varphi_i(\mathbf{r})$ is the linear basis function that is equal to 1 at the i th vertex, \mathbf{v}_i , zero at all other vertices, and whose support lies in the elements containing \mathbf{v}_i ; ξ_i is the unknown value of the solution at \mathbf{v}_i ; N is the total number of nodes in a discretized mesh of C . We consider a standard (non-interface) element $E_i \in \Omega^l$ containing the vertices $\{\mathbf{v}_k\}$, with

$$W(\mathbf{r}) = \sum_{\{k, \mathbf{v}_k \in \bar{E}_i\}} \xi_k \varphi_k(\mathbf{r}), \quad \mathbf{r} \in E_i.$$

We set $v = \varphi_j$ and consider the quantity

$$\int_{E_i} \mathbf{D}\nabla W \cdot \nabla \varphi_j \, d\mathbf{r} - \int_{\partial E_i \cap \Gamma} \mathbf{D}\nabla W \cdot \mathbf{n}^i \varphi_j \, ds = \sum_{\{k, \mathbf{v}_k \in \bar{E}_i\}} \xi_k \left(\int_{E_i} \mathbf{D}\nabla \varphi_k \cdot \nabla \varphi_j \, d\mathbf{r} \right) - \int_{\partial E_i \cap \Gamma} \mathbf{D}\nabla W \cdot \mathbf{n}^i \varphi_j \, ds, \tag{19}$$

for j where $\mathbf{v}_j \in \bar{E}_i$. (For other j , this quantity is zero.) The stiffness matrix associated with the element E_i is the first term on the right-hand side and that has entries

$$S_{jk} = \int_{E_i} \mathbf{D}\nabla \varphi_k \cdot \nabla \varphi_j \, d\mathbf{r}. \tag{20}$$

The second term on the right-hand side,

$$- \int_{\partial E_i \cap \Gamma} (\mathbf{D}\nabla W \cdot \mathbf{n}^i) \varphi_j \, ds, \tag{21}$$

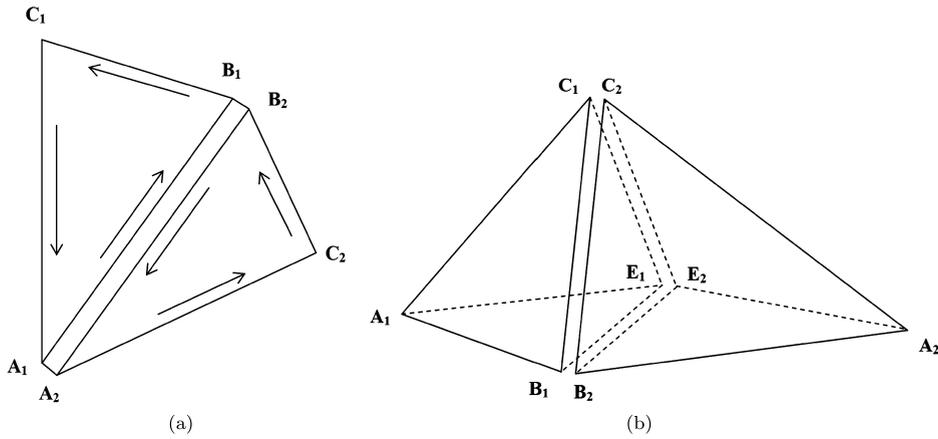


Fig. 2. (a) In two dimensions, neighboring elements $A_1B_1C_1$ and $A_2C_2B_2$, and the interface element $A_2B_2B_1A_1$; (b) in three dimensions, neighboring elements $A_1B_1C_1E_1$ and $A_2B_2C_2E_2$ and the interface element $B_2C_2E_2E_1C_1B_1$.

is zero if $\{\mathbf{v}_j\} \notin \partial E_i \cap \Gamma$. So for this term, we only need to consider vertices on a compartment interface. Because we have doubled nodes on Γ , as described earlier, we know these same vertices are repeated. See Fig. 2 for the two pairs of repeated nodes: $\{A_1, A_2\}, \{B_1, B_2\}$, in two dimensions and the three pairs of repeated nodes $\{B_1, B_2\}, \{C_1, C_2\}, \{E_1, E_2\}$ in three dimensions. For the ease of the global assembly routines, we define an interface element from these repeated nodes: in two dimensions, the interface element is $A_2B_2B_1A_1$, in three dimensions, it is $B_2C_2E_2E_1C_1B_1$. We can then define the local “stiffness” matrix, S^{loc} , associated with the interface elements, by collect the terms associated with Eq. (21):

$$\left(S^{loc} \equiv \frac{\kappa c_k}{6} \begin{bmatrix} 2 & 1 & -1 & -2 \\ 1 & 2 & -2 & -1 \\ -1 & -2 & 2 & 1 \\ -2 & -1 & 1 & 2 \end{bmatrix} \right) \begin{bmatrix} \xi_{A_2} \\ \xi_{B_2} \\ \xi_{B_1} \\ \xi_{A_1} \end{bmatrix},$$

in two dimensions, and

$$\left(S^{loc} \equiv \frac{\kappa c_k}{12} \begin{bmatrix} 2 & 1 & 1 & -1 & -1 & -2 \\ 1 & 2 & 1 & -1 & -2 & -1 \\ 1 & 1 & 2 & -2 & -1 & -1 \\ -1 & -1 & -2 & 2 & 1 & 1 \\ -1 & -2 & -1 & 1 & 2 & 1 \\ -2 & -1 & -1 & 1 & 1 & 2 \end{bmatrix} \right) \begin{bmatrix} \xi_{B_2} \\ \xi_{C_2} \\ \xi_{E_2} \\ \xi_{E_1} \\ \xi_{C_1} \\ \xi_{B_1} \end{bmatrix}, \tag{22}$$

and in three dimensions, where c_k is the length of the interface segment in 2D and the area of the interface triangle in 3D. A derivation can be found in [29].

3.2. Boundary conditions: transformation of the Bloch–Torrey PDE

The pseudo-periodic boundary conditions for the Bloch–Torrey PDE in Eqs. (6)–(7) and the steady-state Laplace PDE in Eqs. (15)–(16) differ slightly from standard periodic boundary conditions. In addition, the boundary conditions for the Bloch–Torrey PDE involve complex numbers and are time-dependent. We limit our discussion to the Bloch–Torrey PDE because it is the more complicated case.

We assume the mesh of C is generated in such a way so that the nodes are mirror reflected on the opposite faces of C . Let $\{E^k\}$ be the elements that touch ∂C . They give rise to the non-zero entry $\int_{\partial E^k \cap \partial C} \mathbf{D} \nabla M \cdot \mathbf{n} \, ds$. Since $\nabla M \cdot \mathbf{n}$ is unknown on ∂C , this quantity has to be eliminated by using the pseudo-periodic relation (7) on the normal derivative. For example, in the x -direction, this means replacing the rows of the global finite elements matrices associated to the face of C , $r_1 = b_1$, by new rows that are obtained by multiplying the original rows by $e^{i\theta_1(t)}$ and subtracting them from the rows associated to the opposite face, $r_1 = a_1$. Then one replaces the rows associated to the face, $r_1 = a_1$, by the pseudo-periodic relation on the function value in Eq. (6).

This naive way of implementing pseudo-periodic boundary conditions introduces complex arithmetic and time dependence into the global finite elements matrices. This is a very undesired characteristic for the mass matrix, since many linear systems involving the mass matrix have to be solved repeatedly.

If we could change the pseudo-periodic boundary conditions to standard periodic boundary conditions, then we can keep the mass matrix real-valued and time-independent. Thus, the same as in [18], we chose to transform the magnetization to a new unknown $m(\mathbf{r}, t)$:

$$m(\mathbf{r}, t) = M(\mathbf{r}, t)e^{I\mathcal{G}(\mathbf{r})\gamma\mathcal{F}(t)}, \quad \mathcal{F}(t) = \int_0^t f(s) ds.$$

The Bloch–Torrey PDE (1) is then transformed to

$$\frac{\partial}{\partial t}m = -I\gamma\mathcal{F}(\nabla\mathcal{G} \cdot \mathbf{D}\nabla m + \nabla m \cdot \mathbf{D}\nabla\mathcal{G}) - \nabla\mathcal{G} \cdot \mathbf{D}\nabla\mathcal{G}m(\gamma\mathcal{F})^2 + \nabla \cdot (\mathbf{D}\nabla m), \quad \mathbf{r} \in \bigcup \Omega^l, \tag{23}$$

with periodic boundary conditions

$$\begin{aligned} [m(\mathbf{x}, t)]_{x_k=-L/2} &= [m(\mathbf{x}, t)]_{x_k=L/2}, \quad k = 1, \dots, d, \\ \left[\mathbf{D}(\mathbf{r}) \frac{\partial}{\partial x_k} m(\mathbf{x}, t) \right]_{x_k=-L/2} &= \left[\mathbf{D}(\mathbf{r}) \frac{\partial}{\partial x_k} m(\mathbf{x}, t) \right]_{x_k=L/2}, \quad k = 1, \dots, d. \end{aligned} \tag{24}$$

The interface conditions (4)–(5) are changed to

$$\begin{aligned} \mathbf{D}^n \nabla m^n \cdot \mathbf{n}^n &= \kappa^{ln}(m^l - m^n) + I\gamma m^n \mathcal{F} \mathbf{D}^n \nabla\mathcal{G} \cdot \mathbf{n}^n, \quad \forall l, n, \\ \mathbf{D}^l \nabla m^l \cdot \mathbf{n}^l &= \kappa^{ln}(m^n - m^l) + I\gamma m^l \mathcal{F} \mathbf{D}^l \nabla\mathcal{G} \cdot \mathbf{n}^l, \quad \forall l, n, \end{aligned} \tag{25}$$

where m^l, m^n are the limiting values in Ω^l and Ω^n . We note that Eq. (23) and Eq. (25) are more complicated than what was used in [18] because we kept all the first order terms in order to obtain second order accuracy in space.

The weak form of Eq. (23) is then

$$\begin{aligned} \frac{\partial}{\partial t} \int_{\Omega^l} m v \, d\mathbf{r} &= -I\gamma\mathcal{F} \int_{\Omega^l} (\nabla\mathcal{G} \cdot \mathbf{D}\nabla m + \nabla m \cdot \mathbf{D}\nabla\mathcal{G}) v \, d\mathbf{r} - \nabla\mathcal{G} \cdot \mathbf{D}\nabla\mathcal{G}(\gamma\mathcal{F})^2 \int_{\Omega^l} m v \, d\mathbf{r} \\ &\quad - \int_{\Omega^l} \mathbf{D}\nabla m \cdot \nabla v \, d\mathbf{r} + \int_{\partial\Omega^l \cap \Gamma} \mathbf{D}\nabla m \cdot \mathbf{n}^l v \, ds + \int_{\partial\Omega^l \cap \partial C} \mathbf{D}\nabla m \cdot \mathbf{n}^l v \, ds. \end{aligned} \tag{26}$$

The boundary and interface conditions are to be imposed as previously. Ignoring the boundary and interface conditions, the weak form (26) can be rewritten in matrix form:

$$\mathbf{M} \frac{\partial}{\partial t} \xi = -I\mathbf{J}\xi - \mathbf{K}\xi - \mathbf{S}\xi, \tag{27}$$

where \mathbf{M} and \mathbf{S} are the mass matrix and stiffness matrix, respectively, and

$$\begin{aligned} \mathbf{J} &= \gamma\mathcal{F} \int_{\Omega} \varphi (\nabla\mathcal{G} \cdot \mathbf{D}\nabla\varphi^T + \nabla\varphi^T \cdot \mathbf{D}\nabla\mathcal{G}) \, d\mathbf{r}, \\ \mathbf{K} &= (\gamma\mathcal{F})^2 \int_{\Omega} \varphi (\nabla\mathcal{G} \cdot \mathbf{D}\nabla\mathcal{G}\varphi^T) \, d\mathbf{r}, \end{aligned}$$

where $\varphi = [\varphi_j]_{j=1, \dots, N}$ is the column vector of basis functions.

In particular, we compute the flux term which is now complex-valued and time-dependent,

$$\int_{\partial E_i \cap \Gamma} (\mathbf{D}\nabla m \cdot \mathbf{n}^i) \varphi \, ds = (\mathbf{S}^{loc} + I\mathbf{J}^{loc}) \xi_{\{k^l, k^n\}}, \tag{28}$$

where

$$\mathbf{J}^{loc} = \frac{\gamma\mathcal{F}C_k \mathbf{D}\nabla\mathcal{G} \cdot \mathbf{n}^i}{6} \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \tag{2D},$$

and

$$\mathbf{J}^{loc} = \frac{\gamma \mathcal{F} C_k \mathbf{D} \nabla \mathcal{G} \cdot \mathbf{n}^i}{12} \begin{bmatrix} 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \\ 1 & 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 1 & 1 & 2 \end{bmatrix} \quad (3D).$$

The matrices of the flux term \mathbf{J}^{loc} and \mathbf{S}^{loc} need to be assembled into the matrix \mathbf{J} and the stiffness matrix \mathbf{S} . The periodic boundary conditions will be applied to the remaining term in Eq. (26), namely $\int_{\partial\Omega^l \cap \partial C} \mathbf{D} \nabla m \cdot \mathbf{n}^l v ds$.

After applying the interface and boundary conditions to Eq. (27), we obtain

$$\tilde{\mathbf{M}} \frac{\partial}{\partial t} \xi = -\tilde{\mathbf{J}} \xi - \tilde{\mathbf{K}} \xi - \tilde{\mathbf{S}} \xi. \quad (29)$$

This matrix equation will be solved to get $\frac{\partial}{\partial t} \xi$, which will be the input of an explicit time-stepping method described in the next section. We can see that the left-hand side contains the mass matrix, while the complex-valued terms are all on the right-hand side. The fact that the mass matrix is real-valued allows one to replace one linear system of size $2N \times 2N$ by two linear systems of size $N \times N$:

$$\begin{aligned} \tilde{\mathbf{M}} \frac{\partial}{\partial t} \xi^R &= \tilde{\mathbf{J}} \xi^I - \tilde{\mathbf{K}} \xi^R - \tilde{\mathbf{S}} \xi^R, \\ \tilde{\mathbf{M}} \frac{\partial}{\partial t} \xi^I &= -\tilde{\mathbf{J}} \xi^R - \tilde{\mathbf{K}} \xi^I - \tilde{\mathbf{S}} \xi^I, \end{aligned} \quad (30)$$

where ξ^R and ξ^I are the real and imaginary parts of ξ , $\xi = \xi^R + I \xi^I$. Moreover, because the mass matrix is time-independent, it needs only to be assembled once, and not at each time step. If Eq. (30) is solved directly, $\tilde{\mathbf{M}}$ will only need to be inverted once. Besides the numerical advantage related to having a real-valued and time-independent mass matrix, this transformation also results in a less oscillatory (both in time and space) unknown function $m(\mathbf{r}, t)$ than the original magnetization $M(\mathbf{r}, t)$, and hence, coarser discretizations can be used [18]. Note that m is identical to M at the initial time ($t = 0$) and at the final time $t = T_E$ since both $\mathcal{F}(0)$ and $\mathcal{F}(T_E)$ vanish.

Because the computational domain C is extended periodically, in some cases the cell interfaces touch ∂C . In this case, the combination of the interface condition and periodic boundary condition is necessary. For details see [29].

3.3. Time stepping for the Bloch–Torrey PDE using explicit RKC

We solve the system of ODEs in Eq. (30) using the Runge–Kutta–Chebyshev (RKC) method [19]. We briefly describe this method as it applies to solving ODEs of the form:

$$\frac{d\{\xi(t)\}}{dt} = F(t, \{\xi(t)\}),$$

where $\xi(t) = \{\xi_{ijk}(t)\}$ contains the unknowns at all the spatial discretization points. Getting $\frac{\partial}{\partial t} \xi(t)$ at a given time from Eq. (30) requires solving two linear systems with the same mass matrix $\tilde{\mathbf{M}}$. To go from $\{\xi\}^n \approx \{\xi(t^n)\}$ to $\{\xi\}^{n+1} \approx \{\xi(t^{n+1})\}$, the following s stages are taken as

$$\begin{aligned} Y^0 &= \{\xi\}^n, \\ Y^1 &= Y^0 + \mu^1 \tau^{n+1} F(t^n, \{\xi\}^n), \\ Y^j &= \mu^j Y^{j-1} + \nu^j Y^{j-2} + (1 - \mu^j - \nu^j) Y^0 + \tilde{\mu} \tau^{n+1} F(t^n + c^{j-1} \tau^{n+1}, Y^{j-1}) + \tilde{\nu} \tau^{n+1} F(t^n, Y^0), \quad 2 \leq j \leq s, \\ \{\xi\}^{n+1} &= Y^s, \end{aligned}$$

where $\tau^{n+1} = (t^{n+1} - t^n)$, and the coefficients are determined by Chebyshev polynomials.

Note that the local time error, ϵ^{n+1} , of the RKC method at t^{n+1} can be estimated as

$$\epsilon^{n+1} \approx \frac{1}{15} (12(\{\xi\}^n - \{\xi\}^{n+1}) + 6\tau^{n+1} (F(t^n, \{\xi\}^n) + F(t^{n+1}, \{\xi\}^{n+1}))),$$

and the time step can be made stable no matter how large it is by increasing the number of stages s . Specifically, the stability criterion is [19]

$$\tau_{RKC}^{n+1} \leq \Delta t_{RKC}^{\max} := \frac{0.653 s^2}{\lambda(JF(t, \{\xi\}(t)))} \approx 0.653 s^2 \frac{h^2}{6D^0}, \quad (31)$$

where λ is the spectral radius and J is the Jacobian operator.

Essentially, the RKC method chooses a t^{n+1} where the error term above is smaller than the user-specified tolerance and the number of stages s is increased to ensure that τ^{n+1} is a stable step. Because the time step can be enlarged as $O(s^2)$ whereas the computational time only increases as $O(s)$, this means that the RKC method with larger time steps computes the solution faster than taking many smaller steps of the Forward Euler method. The number of stages s is typically 30–50 in our simulations. In addition, because of the existence of a three-term recurrence relation for Chebyshev polynomials, the storage requirement does not increase with s , it stays constant at 5 times the number of unknowns.

In contrast, for the Forward Euler method, the stability criterion for a spatial discretization h is

$$\tau_{FE}^{n+1} \leq \Delta t_{FE}^{\max} := \frac{h^2}{6D^0}. \quad (32)$$

For moderate accuracy requirements, it is usually much more efficient to increase s according to Eq. (31) to get a time step that is appropriate for the desired accuracy than being limited by the stability condition (32).

The RKC method has essentially second order time convergence because the error is $O(s^{-3}\tau + C\tau^2)$ [30] and usually s is large enough that $O(\tau^2)$ dominates.

4. Implementation on FEniCS Finite Elements platform and mesh generator Salome

We call the method that we described in the previous section the FEM-RKC method. This method was implemented on the C++ version of FEniCS 1.0.0. To do so, we had to take the following steps.

Because the RKC solver was only available in Fortran, we rewrote it in C++. To define the computational domain, we need two files: a mesh file and a compartment marker file. Currently, we use Salome 6.6.0 to generate meshes. It gives a flexible way to generate some complex geometries with multiple compartments and periodic boundaries. We wrote a C++ subroutine to convert a Salome format '.unv' to FEniCS format '.xml' and a subroutine to create the compartment marker file compatible with the 'CellFunction' of FEniCS.

We then wrote a subroutine to split the mesh and add double nodes outside the FEniCS library. Matrices and vectors for standard elements are automatically assembled by FEniCS supplied routines. We enforced the interface conditions by using subroutines that we wrote outside the FEniCS library.

Various linear solvers are available in FEniCS platform but we mainly use two. For problems with a few thousand unknowns, direct solve with the sparse LU decomposition of $\tilde{\mathbf{M}}$ was used because the factorization can be reused after the first iteration. For larger problems, this decomposition becomes slow and memory demanding, and we use the iterative method GMRES. Unless specified differently, the absolute tolerance 10^{-12} and relative tolerance 10^{-10} were set for GMRES.

All simulations were performed on a Lenovo workstation (Intel(R) Xeon(R) CPU X3430@2.40 GB), running the program as a serial code on Linux Ubuntu 10.04 LTS.

5. Numerical results

In this section we show numerical results on the accuracy and timing of the FEM-RKC method.

Let $\varepsilon(\mathbf{x}, t) = M_h(\mathbf{x}, t) - M(\mathbf{x}, t)$ be the difference between the computed magnetization $M_h(\mathbf{x}, t)$ obtained on a mesh with maximum elements size h and the exact solution $M(\mathbf{x}, t)$, at some fixed time t . We will measure the spatial discretization error in three standard norms:

1. The L^2 error: $\|\varepsilon(\mathbf{x}, t)\|_{L^2} = (\int_{\Omega} |\varepsilon(\mathbf{x}, t)|^2 d\mathbf{x})^{\frac{1}{2}}$;
2. The H^1 error: $\|\varepsilon(\mathbf{x}, t)\|_{H^1} = (\int_{\Omega} |\nabla \varepsilon(\mathbf{x}, t)|^2 d\mathbf{x})^{\frac{1}{2}}$;
3. The L^∞ error: $\|\varepsilon(\mathbf{x}, t)\|_{L^\infty} = \max_{\mathbf{x} \in \Omega} |\varepsilon(\mathbf{x}, t)|$.

Additionally, for the application to dMRI, since the dMRI signal is the integral of the magnetization at $t = T_E$, we define the dMRI signal error:

4. The signal error: $\|\varepsilon(\mathbf{x}, t)\|_S = |\int_{\Omega} \varepsilon(\mathbf{x}, t) d\mathbf{x}|$.

Strictly speaking, the dMRI signal is only experimentally measured at $t = T_E$, but we will show the convergence of the integral of M at $t < T_E$ as well.

The FEM-RKC method will be compared to two other methods:

1. The FEM-BE method, coupling the same finite elements discretization described in the previous section to Backward Euler time stepping.
2. The FVM-RKC method [31], coupling a uniform Cartesian finite volume spatial discretization with RKC time-stepping.

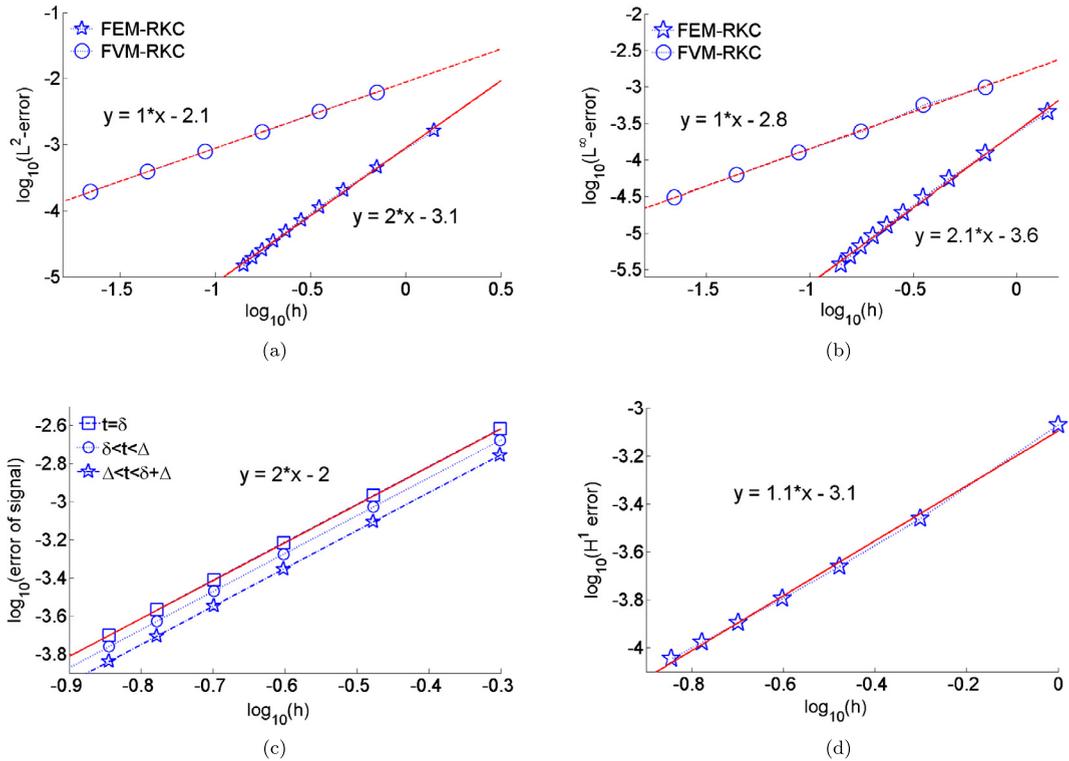


Fig. 3. Two-dimensional homogeneous problem on $C = [-5 \mu\text{m}, 5 \mu\text{m}]^2$. The L^2 -error (a), L^∞ -error (b) and signal error (c) are second order in h for FEM-RKC and of first order for FVM-RKC; FEM-RKC also gives the first order convergence in H^1 -error (d).

5.1. Spatial discretization

Similar to standard FEM, FEM-RKC with linear basis functions is expected to have the second order convergence in the L^2 -norm and the L^∞ -norm and first order in the H^1 -norm.

5.1.1. Homogeneous problem

First we consider a homogeneous domain $C = [-5 \mu\text{m}, 5 \mu\text{m}]^2$, the initial condition is a Dirac delta distribution at $\mathbf{x}_0 = (0, 0)$. The constant diffusion coefficient is $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, $|\mathbf{g}| = 373.8 \text{ mT/m}$, and the time profile is PGSE with $\delta = 4 \text{ ms}$ and $\Delta = 4 \text{ ms}$.

The mesh size was varied from 20×20 to 640×640 vertices for FVM-RKC and from 10×10 to 100×100 vertices for FEM-RKC. The results at $t = 2 \text{ ms}$ are shown in Figs. 3(a), 3(b), 3(d). One can see the second order convergence in the L^2 -norm and L^∞ -norm, and the first order convergence in the H^1 -norm, as expected for FEM-RKC. For FVM-RKC, the convergence is first order in the L^2 -norm and the L^∞ -norm. The H^1 -norm is not defined. Then we study the convergence of the integral of magnetization M . For a homogeneous domain, the mass conservation in both the FEM and FVM methods implies that the integral of M is exact up to numerical accuracy when $\mathcal{F}(t)$ is zero. So we just verify the spatial convergence of the integral of M at $t = \delta$, $\delta < t < \Delta$, and $\delta < t < \Delta + \delta$. Fig. 3(c) shows that the integral of M has second order convergence in space.

The results are similar in 3D. We solve the homogeneous problem on $[-5 \mu\text{m}; 5 \mu\text{m}]^3$ for the PGSE sequence: $\Delta = \delta = 4 \text{ ms}$, $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$ and $|\mathbf{g}| = 373.8 \text{ mT/m}$. Fig. 4 illustrates second order convergence in the L^2 -norm and the L^∞ -norm of FEM-RKC.

5.1.2. Impermeable disk and sphere

We consider the convergence of the dMRI signal (integral of the magnetization at echo time $t = T_E$) inside impermeable circular and spherical cells for the OGSE sequence. Analytical solution for comparison was obtained using the matrix formalism approach [3,10].

The dMRI signal obtained by FEM-RKC for cos-OGSE sequence with $n = 100$ periods, $T_E = 2\sigma = 400 \text{ ms}$ at b -values from 0 to 500 s/mm^2 for an impermeable disk of radius $R = 4.5 \mu\text{m}$, $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, converges to the analytical solution at the second order (Fig. 5(a)). Similarly, we also obtain second order convergence for a 3-dimensional impermeable sphere (Fig. 5(b)). The number of vertices in the various meshes ranged from 50 to 10000 (2D) and 50 to 15000 (3D).

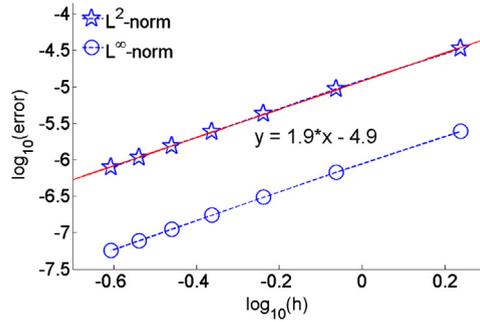


Fig. 4. Spatial convergence of L^2 -error and L^∞ -error for a three-dimensional homogeneous problem on $C = [-5 \mu\text{m}, 5 \mu\text{m}]^3$ with PGSE sequence: $\Delta = \delta = 4 \text{ ms}$, $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$ and $|\mathbf{g}| = 373.8 \text{ mT/m}$ for FEM-RKC.

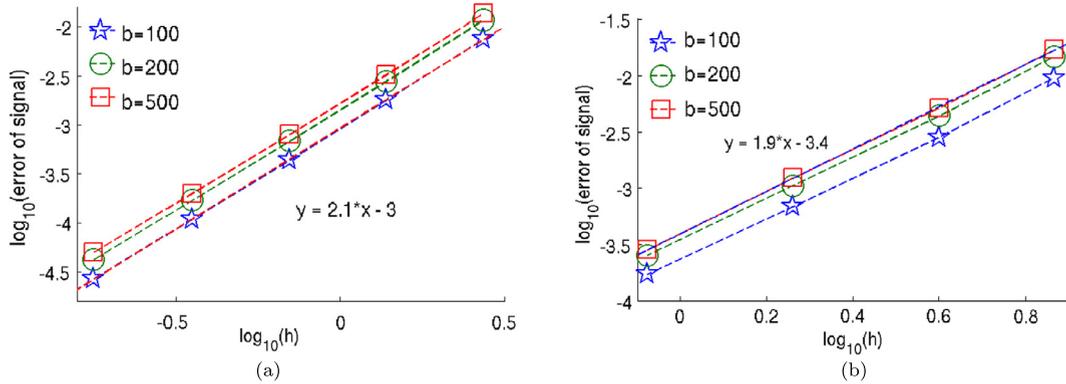


Fig. 5. The convergence of the dMRI signal versus space discretization in impermeable circular and spherical cells in 2D (a) and in 3D (b) with the same cell radius of $R = 4.5 \mu\text{m}$, $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, $T_E = 2\sigma = 400 \text{ ms}$.

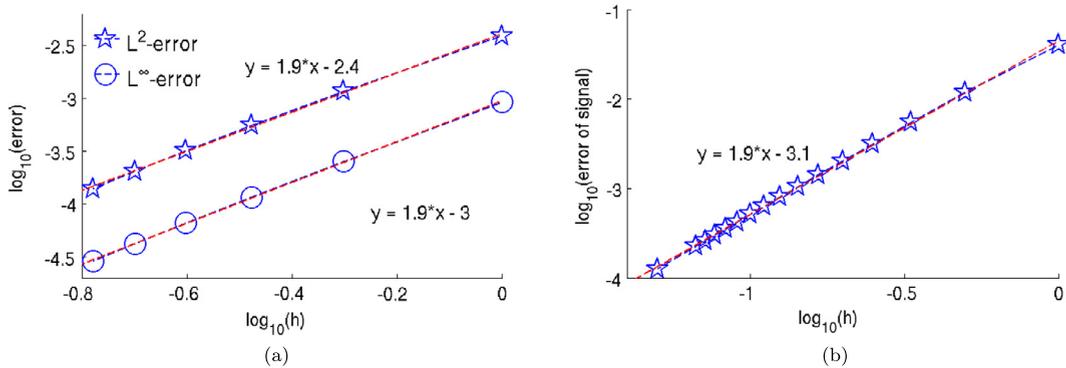


Fig. 6. The spatial convergence for permeable square cells is second order in the L_2 -norm and the L_∞ -norm (a) as well as in the dMRI signal (b).

5.1.3. Permeable square cell

For permeable square cells, there is no analytical solution. We will compare our numerical solution with the reference numerical solution obtained at the finest mesh. One square cell of side length $L = 8 \mu\text{m}$ is placed in the center of the domain $C = [-5 \mu\text{m}; 5 \mu\text{m}]^2$. The boundary conditions on ∂C imply the periodic repetition of the square cell outside of C . Both intra-cellular and extra-cellular compartments have the same intrinsic diffusion coefficient $D = 3 \times 10^{-3} \text{ mm}^2/\text{s}$. Setting the permeability $\kappa = 10^{-5} \text{ m/s}$ for the interface, we consider one case of gradient amplitude $|\mathbf{g}| = 373.8 \text{ mT/m}$. The mesh size was varied from 253 vertices to 7513 vertices and all obtained solutions are compared with the reference solution obtained at 81041 vertices. The time profile is the PGSE sequence with $\Delta = \delta = 10 \text{ ms}$. The results show that FEM-RKC gives second order convergence in the L^2 and L^∞ norms (Fig. 6(a)) and second order convergence in the dMRI signal (Fig. 6(b)), where h is the maximum element size.

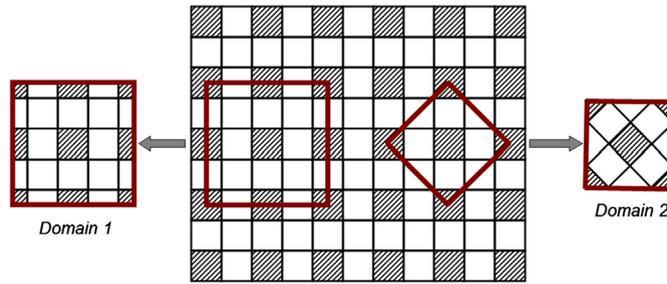


Fig. 7. Two different computational boxes derived from a periodic domain.

Table 1

The dMRI signals computed on domains 1 and 2 from Fig. 7 with $D^1 = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, $D^2 = 10^{-3} \text{ mm}^2/\text{s}$, $\kappa = 5 \cdot 10^{-5} \text{ m/s}$, and a PGSE sequence with $\delta = \Delta = 5 \text{ ms}$, RKC tolerance $tol = 10^{-6}$.

b -value (s/mm^2)	FEM-RKC		FVM-RKC	
	domain 1	domain 2	domain 1	domain 2
0.00	1.0000	1.0000	1.0000	1.0000
92.59	0.8513	0.8514	0.8517	0.8548
370.37	0.5520	0.5526	0.5533	0.5555
833.33	0.3223	0.3226	0.3232	0.3141
1481.48	0.2120	0.2120	0.2123	0.1934
2314.81	0.1680	0.1681	0.1683	0.1456
3333.33	0.1478	0.1480	0.1481	0.1248

5.1.4. Approximation of the interface

Next, we verify that the approximation of the geometry of the interface is more accurate for FEM than FVM, leading to a more accurate dMRI signal approximation. We consider a periodic domain with many striped squares and extract two periodic subdomains: domain 1 and domain 2 (Fig. 7).

In theory, the signals computed by solving the Bloch–Torrey equation with pseudo-periodic BCs on both cases should be identical. We perform two simulations with the parameters $D^1 = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, $D^2 = 10^{-3} \text{ mm}^2/\text{s}$, $\kappa = 5 \cdot 10^{-5} \text{ m/s}$, using the PGSE sequence with $\delta = \Delta = 5 \text{ ms}$. The mesh size is 64×64 for FEM-RKC and 300×300 for FVM-RKC. The results for RKC tolerance $tol = 10^{-6}$ are summarized in Table 1.

One can see that FEM-RKC gives a good approximation of slanted interfaces whereas FVM-RKC fails at high b -values. In fact, the approximation of the slanted interface on a Cartesian grid by a zigzag curve leads to significant errors in the surface area, whatever the spatial resolution of the grid.

5.2. Computational efficiency

The RKC solver for parabolic PDEs has the second order in time convergence [30] and this remains true for FEM-RKC. On the other hand, the Backward Euler method has the first order convergence in time.

We numerically show that FEM-RKC is more efficient than FVM-RKC and FEM-BE in running time. The domain $C = [-5 \mu\text{m}, 5 \mu\text{m}]^2$ is homogeneous, the initial condition is a Dirac delta distribution at $\mathbf{x}_0 = (0, 0)$, the diffusion coefficient is $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$, and $|\mathbf{g}| = 373.8 \text{ mT/m}$. The solution is computed at $t = 2 \text{ ms}$ for a constant gradient, $f(t) = 1$, $t = 0, \dots, 2 \text{ ms}$. We fix the tolerance of RKC solver at $tol = 10^{-9}$ and consider different mesh sizes. The mesh size was varied from 20×20 to 640×640 for FVM-RKC and from 10×10 to 100×100 for FEM-RKC. We keep the same time stepping $dt = 2 \mu\text{s}$ and vary the mesh size from 10×10 to 80×80 for FEM-BE. The results show that the same accuracy is obtained with FEM-RKC much faster than with FVM-RKC and FEM-BE (Fig. 8).

5.3. Timing on heterogeneous domain

We simulated both PGSE and OGSE sequences on the computational box $C = [-20 \mu\text{m}; 20 \mu\text{m}]^3$ containing curved cylindrical cells (Fig. 9) created by Salome. The curved cylinders do not overlap. They cut the faces of C in such a way that the exit of each cylinder on a face matches the entrance of a cylinder (the same or a different one) on the opposing face. This was done by creating first one-eighth of the domain, on $[0; 20 \mu\text{m}]^3$, containing a set of 43 curved cylinders. The remaining part is the extra-cellular space. The radius r of the curved cylinders is set to $1.2 \mu\text{m}$ to obtain an intra-cellular volume fraction of $v^c = 40.3\%$. Then, this one-eighth domain is mirror reflected across the three planes, $x = 0$, $y = 0$, $z = 0$, to obtain C .

For the simulations, the same diffusion coefficient $D = 10^{-3} \text{ mm}^2/\text{s}$ was set for both intra-cellular and extra-cellular compartments. A permeability condition with $\kappa = 10^{-5} \text{ m/s}$ was set between the cylinders and the extra-cellular compartment. The uniform distribution $M(\mathbf{r}, 0) = 1$ was set as the initial condition.

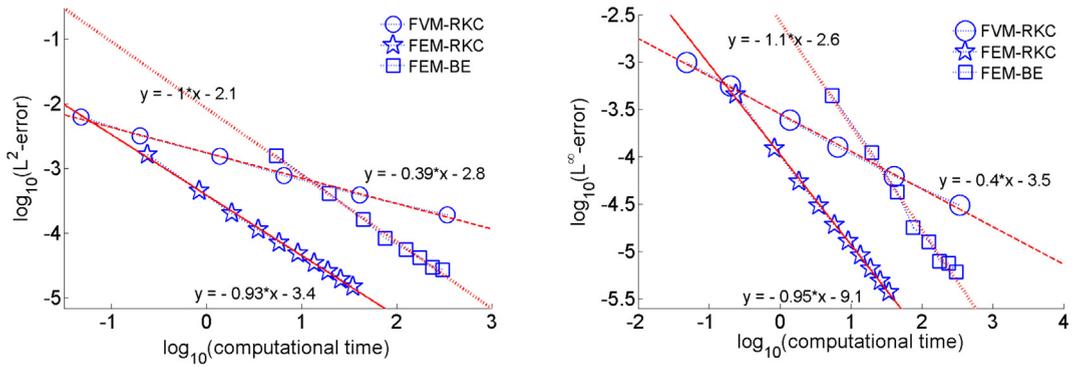


Fig. 8. The accuracy of FEM-RKC, FEM-BE and FVM-RKC versus computational time for different mesh sizes.

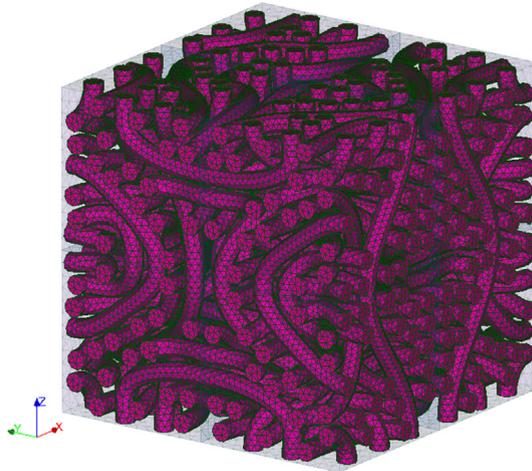


Fig. 9. Computational domain containing curved cylinders created by Salome with intra-cellular volume fraction $v^c = 40.3\%$. There are 260 363 vertices in the finite elements mesh. The curved cylinders do not overlap.

Simulations for the PGSE sequence with $\delta = 2.5$ ms and $\Delta = 10$ ms were performed on three meshes, with 109 995, 260 363, and 389 937 vertices, respectively, to estimate the simulation error. For such a large scale problem, the iterative Krylov solver, the GMRES method, was used. We set RKC tolerance to 10^{-4} , the GMRES absolute tolerance to 10^{-10} and the GMRES relative tolerance to 10^{-6} . The number of GMRES iterations required was around 10 for all the simulations. The average computational times per b -value for the three meshes were 7, 20 and 40 minutes, and the memory usage was 0.5 GB, 1.4 GB and 2 GB, respectively.

One can see in Fig. 10 that as compared to the simulated signal on the finest mesh (389 937 vertices), the simulated signals using meshes with 109 995 and 260 363 vertices have about 12% and 2% maximum relative error, respectively.

We then chose the mesh of 260 363 vertices to simulate the cos-OGSE sequence. The slowest computation took 3 hours for one b -value of the cos-OGSE sequence with $n = 4$ periods, compared to 20 minutes for one b -value ($b > 0$) for the PGSE sequence. Fig. 11(a) shows the dMRI signals for the OGSE sequence with two different frequencies and the PGSE sequence. The signal becomes smaller at higher frequency. The computational time increases monotonically with the gradient amplitude $\|\mathbf{g}\|$ (Fig. 11(b)). Because of the reduction of the oscillations by transforming the unknown to $m(\mathbf{r}, t)$, the computational time only increases slightly with $\|\mathbf{g}\|$.

6. Numerical study of diffusion

In this section we illustrate the diffusion characteristics of some complex domains.

6.1. Relating the \mathbf{D}^{hom} of the three-compartment model and the two-compartment models of cells

It has been proposed [32] that water diffuses slowly in a physical layer around the cell membranes. The thickness of this layer is supposed much larger than the actual thickness of cell membranes (which is in the order of nanometers). Here we study whether this “thick” membrane layer can be approximated by an infinitely thin interface between the intra-cellular

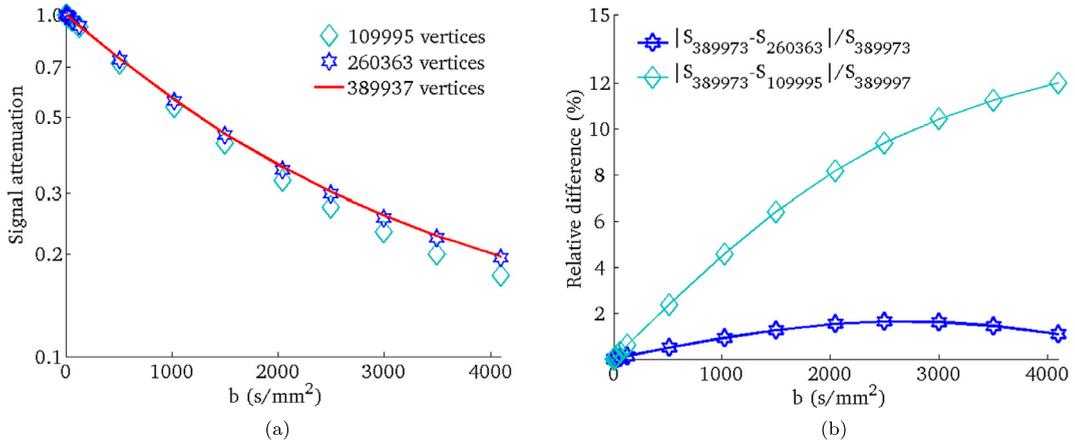


Fig. 10. (a) The simulated signals using three meshes, with 109995, 260363, and 389937 vertices; (b) relative signal difference compared to the finest mesh signal. PGSE sequence with $\delta = 2.5$ ms and $\Delta = 10$ ms.

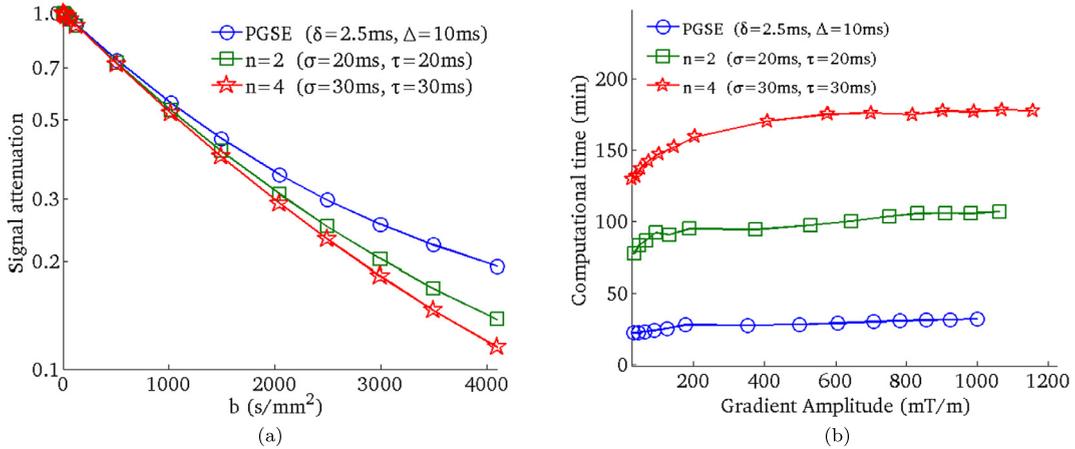


Fig. 11. Signal (a) and computational time (b) for the cos-OGSE sequence with $n = 2, 4$ periods and the PGSE sequence, on the random curved cylinders domain with intra-cellular volume fraction $v^c = 40.3\%$.

space and the extra-cellular space. The word “thick” refers to the fact that the width of this layer is much larger than the physical width of the cell membranes, but it is small compared to the cell radius.

In this three-compartment model, the physical space is the union of the cell interiors Ω^c , the space exterior to the cells Ω^e , and the membrane layer Ω^m of thickness ℓ (see Fig. 12(a)). Furthermore, we suppose that the magnetization is continuous across the cell-membrane and the membrane-extra-cellular space interfaces (infinite permeability coefficient). Each physical compartment is characterized by its own intrinsic diffusion coefficient D^c , D^e and D^m respectively. In the two-compartment model, the membrane layer is replaced by an infinitely thin interface (see Fig. 12(b)) which is characterized by a finite permeability κ , that to a first order approximation is:

$$\kappa \approx D^m / \ell. \tag{33}$$

The accuracy of this approximation depends on the thickness and the permeability of the membrane. We perform the simulations on the domain $C = [-5 \mu\text{m}, 5 \mu\text{m}]^3$ containing one spherical cell of radius $R = 4 \mu\text{m}$. The thickness ℓ of the membrane layer is decreased from $1.0 \mu\text{m}$ to $0.1 \mu\text{m}$ in order to check the closeness of the two models. Fig. 13(a) shows the $\frac{\mathbf{g}^T}{\|\mathbf{g}\|} \mathbf{D}^{\text{hom}} \frac{\mathbf{g}}{\|\mathbf{g}\|}$, for $\mathbf{g} / \|\mathbf{g}\| = [1, 0, 0]$, computed by solving the steady-state Laplace PDEs for two- and three-compartment domains with $\kappa = 10^{-5}$ m/s. As ℓ tends to 0, the value from the three-compartment model approaches that of the two-compartment model. The relative difference is less than 2% when $\ell \leq 0.1 \mu\text{m}$ (Fig. 13(b)). For $\ell = 0.1 \mu\text{m}$, the difference between the two signals is less than 2% for $\kappa = 10^{-5}$ m/s, $\delta = \Delta = 10$ ms (Fig. 13(c)).

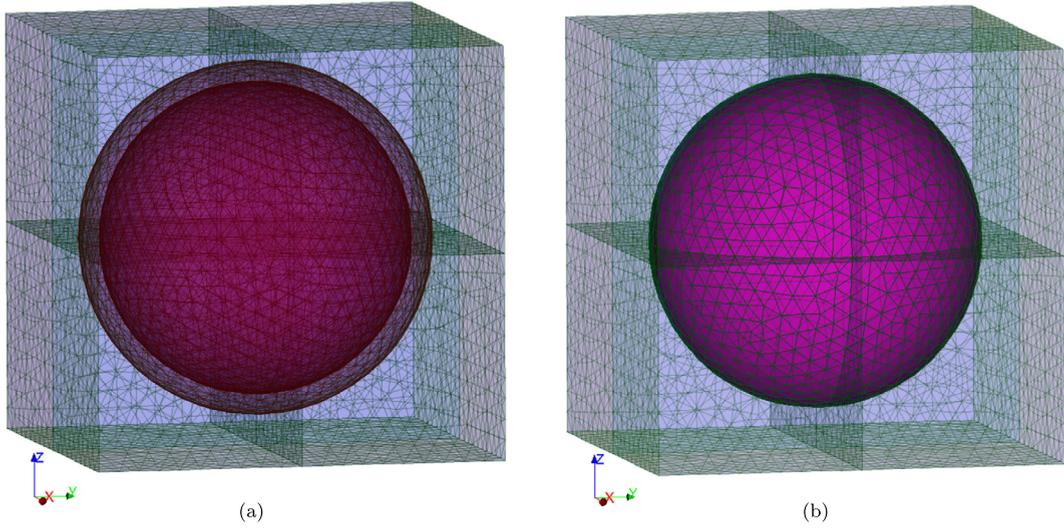


Fig. 12. In the three-compartment model (a), the physical space is the union of the cell interiors Ω^c , the space exterior to the cells Ω^e , and the membrane layer Ω^m of thickness ℓ . Each physical compartment is characterized by its own intrinsic diffusion coefficient D^c , D^e and D^m respectively. In the two-compartment model (b), the intermediate layer is replaced by an infinitely thin interface that is characterized by a finite permeability κ .

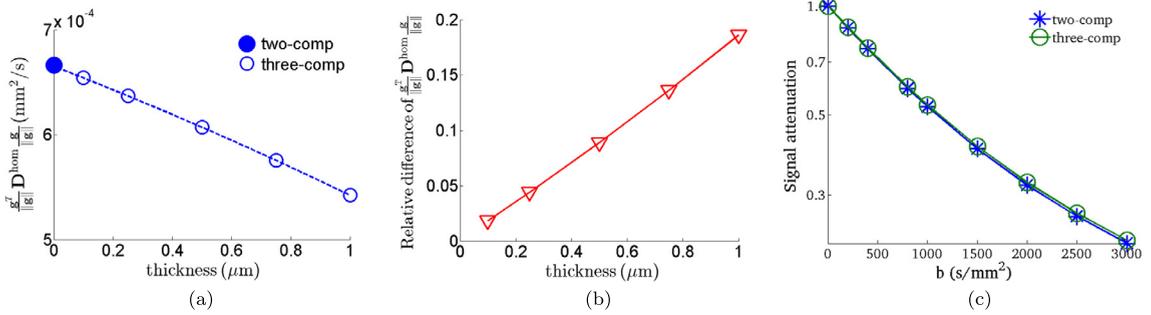


Fig. 13. Comparison between the three-compartment (Fig. 2(a)) and two-compartment (Fig. 2(b)) models, $\mathbf{g}/\|\mathbf{g}\| = [1, 0, 0]$, $\kappa = 10^{-5}$ m/s. As ℓ tends to 0, the $\frac{\mathbf{g}^T \mathbf{D}^{\text{hom}} \mathbf{g}}{\|\mathbf{g}\|}$ of the three-compartment model approaches that of the two-compartment model (a). The relative difference is shown in (b). For $\ell = 0.1 \mu\text{m}$, the difference between two signals is less than 2% for $\delta = \Delta = 10$ ms (c).

6.2. The convergence of the D^A to $\frac{\mathbf{g}^T \mathbf{D}^{\text{hom}} \mathbf{g}}{\|\mathbf{g}\|}$

Not accounting for fitting errors, the apparent diffusion coefficient (D^A) is the first derivative of the logarithm of the normalized dMRI signal with respect to b -value. In this section, we will numerically show that D^A converges to $\frac{\mathbf{g}^T \mathbf{D}^{\text{hom}} \mathbf{g}}{\|\mathbf{g}\|}$, where \mathbf{D}^{hom} is computed by solving the steady-state Laplace PDEs.

We consider a computational domain $C = [-10 \mu\text{m}, 10 \mu\text{m}]^3$ containing 100 random Voronoi cells (Fig. 14(a)) with intra-cellular volume fraction $v^c = 61.4\%$ and surface-to-volume ratio $1.03 \mu\text{m}^{-1}$. The same intrinsic diffusion coefficient $D^e = D^c = 10^{-3} \text{ mm}^2/\text{s}$ is set for both intra-cellular and extra-cellular compartments. Two permeabilities, $\kappa = 10^{-5}$ m/s and $\kappa = 10^{-4}$ m/s, were simulated. The Bloch–Torrey PDE was solved for the PGSE sequences with $\delta = 2.5$ ms and several $\Delta = 10, 20, 40, 80, 160$ ms at five b -values 0, 50, 200, 500 and 1000 s/mm^2 . From each Δ , the D^A is obtained by fitting a cubic polynomial to the curve $-\log S(b)/\log S(0)$. The steady-state Laplace PDE was also solved over C to obtain \mathbf{D}^{hom} according to Eqs. (14), (4)–(5), (15), (16).

Fig. 14(b) shows that D^A converges to $\frac{\mathbf{g}^T \mathbf{D}^{\text{hom}} \mathbf{g}}{\|\mathbf{g}\|}$ in three gradient directions: $[1, 0, 0]$, $[0, 1, 0]$ and $[0, 0, 1]$, where $\kappa = 10^{-4}$ m/s. The convergence is faster at higher permeability (Fig. 14(c)) and seems to be linear versus Δ^{-1} when the diffusion time $\Delta - \delta/3$ is long enough. This agrees with the result for 1D periodic structure in the long-time regime [33].

For a mesh size with 28 688 nodes, each D^A is computed in 10 to 20 minutes whereas the computation of one \mathbf{D}^{hom} takes less than one minute.

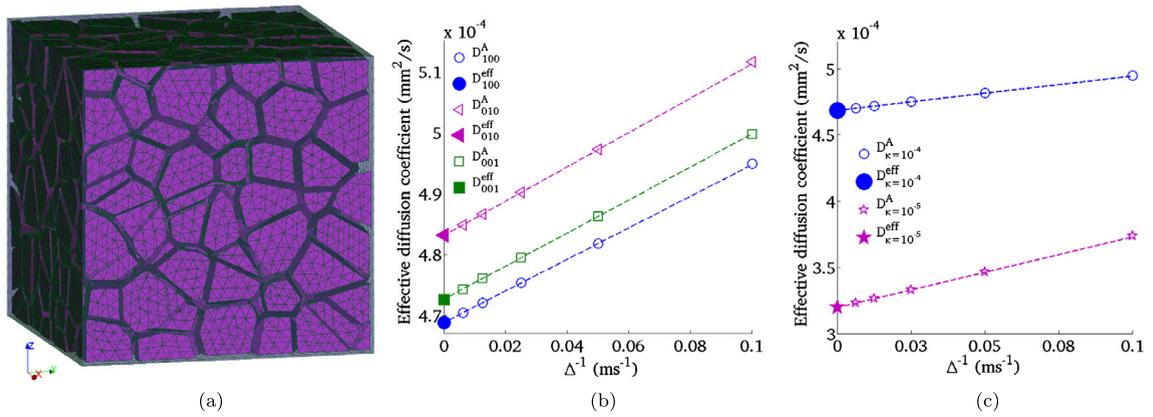


Fig. 14. (a) The computational box $[-10 \mu\text{m}, 10 \mu\text{m}]^3$ contains 100 Voronoi cells with intra-cellular volume fraction $v^c = 61.4\%$ and surface-to-volume ratio $S/V = 1.03 \mu\text{m}^{-1}$; (b) the convergence of D^A to $\frac{\kappa}{\|\mathbf{g}\|} D^{\text{hom}} \frac{\kappa}{\|\mathbf{g}\|}$ in three directions $[1, 0, 0]$, $[0, 1, 0]$ and $[0, 0, 1]$ for $\kappa = 10^{-4}$ m/s; (c) with the same gradient direction $[1, 0, 0]$, the convergence is faster at higher permeability and seems to be linear versus Δ^{-1} when the diffusion time is long enough.

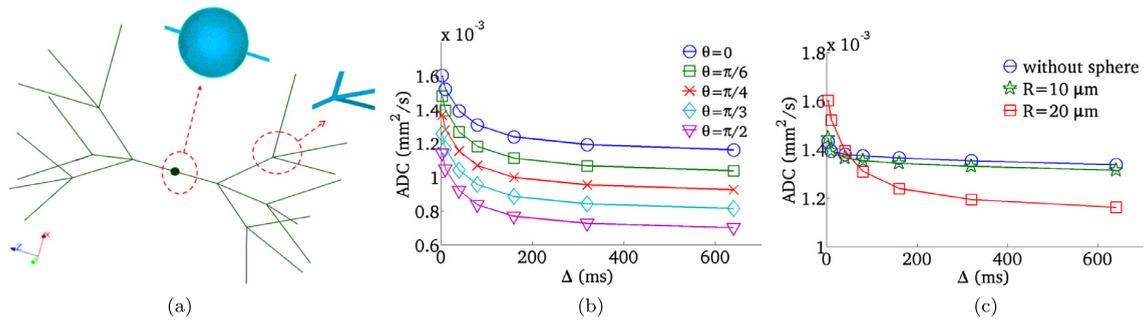


Fig. 15. (a) A neuron with a spherical neuronal body of radius $R = 20 \mu\text{m}$, the volume ratio between the dendrites (long cylindrical segments) and the sphere is 6.8. The length of the dendrite segments varies from 400 to 900 μm ; (b) the D^A of a neuron with a spherical body of radius $R = 20 \mu\text{m}$ at $\Delta = 2.5, 10, 40, 80, 160, 320, 640$ ms for various gradient directions, here θ indicates the angle between the dendrite trunk and the gradient directions. It monotonically decreases with Δ and approaches a steady-state value. The D^A is the highest when the gradient direction is parallel to the trunk of the dendrite tree; (c) the D^A approaches the steady state value faster for smaller R , shown for the gradient direction parallel to the trunk.

6.3. The apparent diffusion coefficient of neurons

Neurons are made of a solid neuronal body to which are attached long protrusions called axons and dendrites. We consider the apparent diffusion coefficient, D^A , due to water molecules diffusing inside neurons. In this simulation we do not consider the water exchange between the neurons and the extra-cellular space, thus, we make the neurons impermeable. In this case, there is no need for a computational box, the domain of simulation is the neuron itself. We construct a neuron with a spherical neuronal body to which long cylindrical segments (dendrites) are attached (Fig. 15(a)). The length of the dendrite segments varies between 400 and 900 μm . We consulted [34] for the physical dimensions. The zero Neumann boundary condition is applied on the surface of the neuron and the intrinsic diffusion coefficient inside the neuron is set to $D = 3 \cdot 10^{-3} \text{ mm}^2/\text{s}$. We observe the behavior of the D^A for a PGSE sequence with $\delta = 2.5$ ms and $\Delta = 2.5, 10, 40, 80, 160, 320, 640$ ms. Although realistic dMRI times for brain tissue requires that $\Delta \leq 100$ ms, longer Δ were simulated here to show the convergence of the D^A to a steady-state value.

Fig. 15(b) shows the results for an example where the radius of the spherical neuronal body is $R = 20 \mu\text{m}$. The volume ratio between the dendrites and the sphere is 6.8. One can see that the D^A monotonically decreases with Δ and approaches a steady-state value for different diffusion directions labeled by their angles θ measured from the trunk of the neuron. The D^A is the highest when the gradient direction is parallel to the trunk of the dendrite tree. Fig. 15(c) shows two other cases: (1) a smaller neuronal body $R = 10 \mu\text{m}$, and (2) no neuronal body, for the gradient direction parallel to the trunk. One can see that the D^A approaches the steady state value faster when the neuronal body is smaller.

7. Conclusion

We developed an efficient FEM-RKC method combining the RKC time-stepping method with a specially formulated finite elements spatial discretization to solve two types of PDEs coming from the field of diffusion MRI. Based on the double-node technique and a body-fitting mesh, FEM-RKC can give a better approximation at the cell interfaces than a Cartesian spatial

discretization. By a transformation of the Bloch–Torrey PDE, the pseudo-periodic boundary conditions were transformed to periodic ones and the oscillations in the searched-for solution were reduced. The FEM–RKC method with linear basis functions gives second order convergence in both time and space, compared to the approach in the existing literature which is first order accurate in space and time. Our method should result in improvements in both the computational time and the accuracy of dMRI signal simulations.

This efficient method can become a useful tool to investigate the diffusion of water molecules in complex biological domains and we illustrated this with three examples. One is showing that an infinitely thin membrane model can be used to approximate a thick membrane model. The second is that the apparent diffusion coefficient measured by dMRI approaches the value predicted by mathematical homogenization as diffusion time increases. The third is that when considering the dMRI signal arising from neurons, the measured D^A approaches the steady state value faster when the neuronal body is smaller. This is a precursor to more quantitative and predictive simulations of the diffusion MRI signal in biological tissue.

A version of the code described in this paper will be made publicly available in the near future.

Acknowledgements

We would like to thank the FEniCS supporters for many helpful discussions in the launchpad during the course of our code development. We are especially grateful to Johan Jansson and Johan Hoffman at the KTH, Sweden, for their welcome and help during and after our visit. Related to the use of the Salome platform, we would like to thank Christophe Bourcier, Nathalie Gore and Serge Reh binder in the Laboratoire de Genie Logiciel et de Simulation at the CEA, Saclay, France.

This work is funded in part by the ANR (Agence Nationale de la Recherche), project ‘SIMUDMRI’.

References

- [1] D. Le Bihan, H. Johansen-Berg, Diffusion MRI at 25: Exploring brain tissue structure and function, *NeuroImage* 61 (2) (2012) 324–341, <http://www.sciencedirect.com/science/article/pii/S1053811911012857>.
- [2] S.D. Stoller, W. Happer, F.J. Dyson, Transverse spin relaxation in inhomogeneous magnetic fields, *Phys. Rev. A* 44 (11) (1991) 7459–7477, <http://link.aps.org/doi/10.1103/PhysRevA.44.7459>.
- [3] P. Callaghan, A simple matrix formalism for spin echo analysis of restricted diffusion under generalized gradient waveforms, *J. Magn. Res.* 129 (1) (1997) 74–84, <http://dx.doi.org/10.1006/jmre.1997.1233>.
- [4] J. Pfeuffer, U. Fogel, W. Dreher, D. Leibfritz, Restricted diffusion and exchange of intracellular water: theoretical modelling and diffusion time dependence of 1H NMR measurements on perfused glial cells, *NMR Biomed.* 11 (1) (1998) 19–31, [http://dx.doi.org/10.1002/\(SICI\)1099-1492\(199802\)11:1<19::AID-NBM499>3.0.CO;2-O](http://dx.doi.org/10.1002/(SICI)1099-1492(199802)11:1<19::AID-NBM499>3.0.CO;2-O).
- [5] W.S. Price, A.V. Barzykin, K. Hayamizu, M. Tachiyu, A model for diffusive transport through a spherical interface probed by pulsed-field gradient NMR, *Biophys. J.* 74 (5) (1998) 2259–2271, <http://www.sciencedirect.com/science/article/B94RW-4V295RC-7/2/b7456f86e03ec9b0cc1a9ad9a1986cc2>.
- [6] A.L. Sukstanskii, D.A. Yablonskiy, Effects of restricted diffusion on MR signal formation, *J. Magn. Res.* 157 (1) (2002) 92–105, <http://www.sciencedirect.com/science/article/pii/S1090780702925826>.
- [7] D. Grebenkov, NMR survey of reflected Brownian motion, *Rev. Mod. Phys.* 79 (3) (2007) 1077–1137, <http://dx.doi.org/10.1103/RevModPhys.79.1077>.
- [8] B.F. Moroney, T. Stait-Gardner, B. Ghadirian, N.N. Yadav, W.S. Price, Numerical analysis of NMR diffusion measurements in the short gradient pulse limit, *J. Magn. Res.* 234 (0) (2013) 165–175, <http://www.sciencedirect.com/science/article/pii/S1090780713001572>.
- [9] H. Torrey, Bloch equations with diffusion terms, *Phys. Rev.* 104 (3) (1956) 563–565, <http://dx.doi.org/10.1103/PhysRev.104.563>, Online Archive (Prola).
- [10] D.S. Grebenkov, Pulsed-gradient spin-echo monitoring of restricted diffusion in multilayered structures, *J. Magn. Res.* 205 (2) (2010) 181–195, <http://www.sciencedirect.com/science/article/pii/S1090780710001199>.
- [11] H. Hagslatt, B. Jonsson, M. Nyden, O. Soderman, Predictions of pulsed field gradient NMR echo-decays for molecules diffusing in various restrictive geometries. Simulations of diffusion propagators based on a finite element method, *J. Magn. Res.* 161 (2) (2003) 138–147, <http://www.sciencedirect.com/science/article/pii/S1090780702000393>.
- [12] N. Loren, H. Hagslatt, M. Nyden, A.-M. Hermansson, Water mobility in heterogeneous emulsions determined by a new combination of confocal laser scanning microscopy, image analysis, nuclear magnetic resonance diffusometry, and finite element method simulation, *J. Chem. Phys.* 122 (2) (2005), <http://dx.doi.org/10.1063/1.1830432>.
- [13] J. Chung, G.M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method, *J. Appl. Mech.* 60 (2) (1993) 371–375, <http://dx.doi.org/10.1115/1.2900803>.
- [14] M.H. Bleses, The effect of finite duration of gradient pulses on the pulsed-field-gradient NMR method for studying restricted diffusion, *J. Magn. Reson., Ser. A* 109 (2) (1994) 203–209, <http://www.sciencedirect.com/science/article/pii/S1064185884711569>.
- [15] S.N. Hwang, C.-L. Chin, F.W. Wehrli, D.B. Hackney, An image-based finite difference model for simulating restricted diffusion, *Magn. Reson. Med.* 50 (2) (2003) 373–382, <http://dx.doi.org/10.1002/mrm.10536>.
- [16] J. Xu, M. Does, J. Gore, Numerical study of water diffusion in biological tissues using an improved finite difference method, *Phys. Med. Biol.* 52 (7) (2007) 111–126, <http://dx.doi.org/10.1088/0031-9155/52/7/N01>.
- [17] K.D. Harkins, J.-P. Galons, T.W. Secomb, T.P. Trouard, Assessment of the effects of cellular tissue properties on ADC measurements by numerical simulation of water diffusion, *Magn. Reson. Med.* 62 (6) (2009) 1414–1422, <http://dx.doi.org/10.1002/mrm.22155>.
- [18] G. Russell, K.D. Harkins, T.W. Secomb, J.-P. Galons, T.P. Trouard, A finite difference method with periodic boundary conditions for simulations of diffusion-weighted magnetic resonance experiments in tissue, *Phys. Med. Biol.* 57 (4) (2012) N35, <http://stacks.iop.org/0031-9155/57/i=4/a=N35>.
- [19] B.P. Sommeijer, L.F. Shampine, J.G. Verwer, RKC: An explicit solver for parabolic PDEs, *J. Comput. Appl. Math.* 88 (2) (1998) 315–326, <http://www.sciencedirect.com/science/article/B6TYH-3W0G0R4-12/2/96d3aca02fea5e5bcb582c862e1b3945>.
- [20] E.O. Stejskal, J.E. Tanner, Spin diffusion measurements: Spin echoes in the presence of a time-dependent field gradient, *J. Chem. Phys.* 42 (1) (1965) 288–292, <http://dx.doi.org/10.1063/1.1695690>.
- [21] P.T. Callaghan, J. Stepanik, Frequency-domain analysis of spin motion using modulated-gradient NMR, *J. Magn. Reson., Ser. A* 117 (1) (1995) 118–122, <http://www.sciencedirect.com/science/article/pii/S1064185885799597>.
- [22] M.D. Does, E.C. Parsons, J.C. Gore, Oscillating gradient measurements of water diffusion in normal and globally ischemic rat brain, *Magn. Reson. Med.* 49 (2) (2003) 206–215, <http://dx.doi.org/10.1002/mrm.10385>.
- [23] I. Drobniak, B. Siow, D.C. Alexander, Optimizing gradient waveforms for microstructure sensitivity in diffusion-weighted MR, *J. Magn. Res.* 206 (1) (2010) 41–51, <http://www.sciencedirect.com/science/article/pii/S1090780710001606>.

- [24] A. Logg, K.-A. Mardal, G.N. Wells, et al., *Automated Solution of Differential Equations by the Finite Element Method*, Springer, 2012, <http://dx.doi.org/10.1007/978-3-642-23099-8>.
- [25] SALOME, The open source integration platform for numerical simulation, <http://www.salome-platform.org/>.
- [26] V. Kenkre, Simple solutions of the Torrey–Bloch equations in the NMR study of molecular diffusion, *J. Magn. Res.* 128 (1) (1997) 62–69, <http://dx.doi.org/10.1006/jmre.1997.1216>.
- [27] A. Bensoussan, J.-L. Lions, G. Papanicolaou, *Asymptotic Analysis for Periodic Structures*, *Stud. Math. Appl.*, vol. 5, North-Holland Publishing Co., Amsterdam, 1978.
- [28] H. Cheng, S. Torquato, Effective conductivity of periodic arrays of spheres with interfacial resistance, *Proc., Math. Phys. Eng. Sci.* 453 (1956/1997) 145–161, <http://www.jstor.org/stable/52987>.
- [29] D.V. Nguyen, A finite elements method to solve the Bloch–Torrey equation applied to diffusion magnetic resonance imaging of biological tissues, Ph.D. thesis, Ecole Polytechnique, 2013–2014.
- [30] J. Verwer, W. Hundsdorfer, B. Sommeijer, Convergence properties of the Runge–Kutta–Chebyshev method, *Numer. Math.* 57 (1) (1990) 157–178, <http://dx.doi.org/10.1007/BF01386405>.
- [31] J.-R. Li, D. Calhoun, C. Poupon, D. Le Bihan, Numerical simulation of diffusion MRI signals using an adaptive time-stepping method, *Phys. Med. Biol.* 59 (2) (2013) 441–454, <http://dx.doi.org/10.1088/0031-9155/59/2/441>.
- [32] D. LeBihan, The ‘wet mind’: water and functional neuroimaging, *Phys. Med. Biol.* 52 (7) (2007), <http://dx.doi.org/10.1088/0031-9155/52/7/R02>.
- [33] D.A. Yablonskiy, A.L. Sukstanskii, Theoretical models of the diffusion weighted MR signal, *NMR Biomed.* 23 (7) (2010) 661–681, <http://dx.doi.org/10.1002/nbm.1520>.
- [34] B. Hellwig, A quantitative analysis of the local connectivity between pyramidal neurons in layers 2/3 of the rat visual cortex, *Biol. Cybern.* 82 (2) (2000) 111–121, <http://dx.doi.org/10.1007/PL00007964>.