IOP Publishing Phys. Med. Biol. 68 (2023) 175036 https://doi.org/10.1088/1361-6560/acf022

Physics in Medicine & Biology

IPEM Institute of Physics and Engineering in Medicine

RECEIVED
25 April 2023

REVISED
14 July 2023

ACCEPTED FOR PUBLICATION
14 August 2023

PUBLISHED
29 August 2023

PAPER

Incorporating interface permeability into the diffusion MRI signal representation while using impermeable Laplace eigenfunctions

Zheyi Yang, Chengran Fang and Jing-Rebecca Li

Equipe IDEFIX, INRIA Saclay, UMA, ENSTA PARIS, Palaiseau, France

E-mail: jingrebecca.li@inria.fr

Keywords: Bloch–Torrey equation, diffusion magnetic resonance imaging, matrix formalism, permeability

Abstract

Objective. The complex-valued transverse magnetization due to diffusion-encoding magnetic field gradients acting on a permeable medium can be modeled by the Bloch–Torrey partial differential equation. The diffusion magnetic resonance imaging (MRI) signal has a representation in the basis of the Laplace eigenfunctions of the medium. However, in order to estimate the permeability coefficient from diffusion MRI data, it is desirable that the forward solution can be calculated efficiently for many values of permeability. *Approach*. In this paper we propose a new formulation of the permeable diffusion MRI signal representation in the basis of the Laplace eigenfunctions of the same medium where the interfaces are made impermeable. *Main results.* We proved the theoretical equivalence between our new formulation and the original formulation in the case that the full eigendecomposition is used. We validated our method numerically and showed promising numerical results when a partial eigendecomposition is used. Two diffusion MRI sequences were used to illustrate the numerical validity of our new method. *Significance.* Our approach means that the same basis (the impermeable set) can be used for all permeability values, which reduces the computational time significantly, enabling the study of the effects of the permeability coefficient on the diffusion MRI signal in the future.

1. Introduction

Diffusion magnetic resonance imaging (diffusion MRI) is a widely-used non-invasive imaging modality to probe the micro-structural properties of biological tissue by indirectly measuring the diffusion displacement of water molecules (Stejskal and Tanner 1965, Le Bihan *et al* 1986). In the free diffusion case, the mean squared displacement of molecules is given by $\overline{x^2} = 2dD_0t$, where d is the spatial dimension, D_0 is the intrinsic diffusion coefficient and t is the diffusion time. In biological tissue, the diffusion process is usually hindered or restricted by cell membranes and the mean squared displacement will be smaller than in the case of free diffusion. This deviation can serve to infer tissue micro-structural information (Palombo *et al* 2020, Romascano *et al* 2020).

The estimation of micro-structural parameters is of research and clinical interest. Permeable membranes occur in biological tissues and ignoring permeability effects will make micro-structural estimation inaccurate. For example, ignoring the permeability may under-estimate neurite volume fraction (Jelescu *et al* 2022). Some recent works in the diffusion MRI literature on tissue micro-structural estimation have begun to take cell membrane permeability into account (Nguyen *et al* 2015, Bai *et al* 2020, Jelescu *et al* 2022, Olesen *et al* 2022).

The diffusion MRI physics is mathematically described by the Bloch–Torrey partial differential equation (PDE) (Torrey 1956), which governs the time evolution of the complex transverse water proton magnetization subject to diffusion-encoding magnetic field gradient pulses. For simple geometries such as circles, spheres, plates, analytical expressions of the diffusion MRI signal exist. These analytical expressions have been used to estimate tissue micro-structure and interface permeability (Nedjati-Gilani *et al* 2017, Moutal and Grebenkov 2019, Bai *et al* 2020, Jelescu *et al* 2022). However, for more complex and realistic cellular geometries, there are no explicit analytical expressions available and numerical simulations are needed.

If only a small number of simulations are needed, the two main groups of approaches are (1) Monte Carlo/random walk simulations (Hall and Alexander 2009, Grebenkov 2011, Waudby and Christodoulou 2011, Yeh *et al* 2013, Grebenkov 2014) and (2) solving the discretized Bloch–Torrey PDE (Russell *et al* 2012, Nguyen *et al* 2014, Beltrachini *et al* 2015, Nguyen *et al* 2019). Monte Carlo simulation uses random walkers to mimic the diffusion process during a diffusion MRI experiment. It randomly places a large number of spins inside the complex geometry, and let them move according to the diffusion dynamics. To incorporate permeable membranes, the water exchange through interfaces is modeled via a transit probability P_{trans}, which is the probability that spins will either cross or reflect when they arrive at a permeable interface (Fieremans *et al* 2010, Lee *et al* 2020, 2021, Alemany *et al* 2022). However, as the permeability increases, the time steps must become smaller (see the reasoning in the paper (Fieremans *et al* 2010, Lee *et al* 2020) to obtain the condition $P_{\text{trans}} \ll 1$) which results in a high demand of computational resources and computer memory. The discretization of the Bloch Torrey PDE can be used to directly solve for the magnetization in a geometrical configuration. The computational domain is discretized either by finite elements (Nguyen *et al* 2014, Beltrachini *et al* 2015, Nguyen *et al* 2019) or finite differences (Russell *et al* 2012). SpinDoctor (Li *et al* 2019) is a MATLAB-based diffusion MRI simulation toolbox that solves the Bloch–Torrey PDE using the finite element method (FEM) and an adaptive time stepping method, allowing for arbitrary values of permeability.

Finally, we come to another important representation of the diffusion MRI signal, derived twenty years ago, that projects the magnetization in the basis of the eigenfunctions of the Laplace operator in the imaged domain. This representation goes under the name of matrix formalism (Callaghan 1997, Barzykin 1999, Grebenkov 2007, Drobnjak *et al* 2011). There are two advantages to the matrix formalism signal representation (Li *et al* 2020). First, it makes explicit the link between the Laplace eigenvalues and eigenfunctions of the medium and its diffusion MRI signal. This clear link may help in the formulation of reduced models of the diffusion MRI signal. Second, once the Laplace eigendecomposition has been computed and saved, the diffusion MRI signal can be calculated for many experimental configurations at negligible additional cost. This makes it feasible to use the matrix formalism as the inner loop of optimization or parameter estimation procedures.

In a previous work, we presented a numerical implementation of the matrix formalism for permeable interfaces (Agdestein *et al* 2021), called the numerical matrix formalism method, where the permeability interface conditions are incorporated in the Laplace eigendecomposition step. In this paper, we present a new method, where the diffusion MRI signal of a permeable medium is computed using only impermeable Laplace eigenfunctions. This idea is inspired by how the paper (Grebenkov 2008) treats surface relaxation. We prove that the new method produces the same diffusion MRI signal as the original numerical matrix formalism method, under the condition that the full set of eigenfunctions is used. We show the numerical convergence of the new method when the number of eigenfunctions used is much smaller than the full set. We also show the improved computational efficiency of the new method if simulations using many permeability coefficients are needed.

2. Theory

In this section we introduce the Bloch–Torrey equation and the numerical matrix formalism method.

2.1. Geometrical description

Consider a connected domain $\Omega = \bigcup_{i=1}^{N_{cmpt}} \Omega_i \in \mathbb{R}^d$, made up of N_{cmpt} compartments $\{\Omega_i\}_{1 \leqslant i \leqslant N_{cmpt}}$. We denote the interface between two compartments Ω_i and Ω_j by $\Gamma_{ij} = \Omega_i \cap \Omega_j$ for $i \neq j$, $(i, j) \in \{1,\ldots,N_{cmpt}\}^2$. If two compartments do not touch each other, $\Gamma_{ij} = \varnothing$. Let $\partial\Omega$ be the outer boundary of the domain Ω, we denote the restriction of outer boundary in compartment Ω_i by $\Sigma_i = \partial\Omega \cap \Omega_i$, $i \in \{1, \ldots, N_{cmpt}\}$. If the compartment does not touch the outer boundary, $\Sigma_i = \varnothing$.

We model the brain white matter using the above geometrical description. Axons are enclosed in the extracellular space (ECS), with the water exchange between axons and the ECS described by a permeability coefficient. The geometry has an outer boundary that is impermeable to water. Thus, the number of compartments is the number of axons plus one (the ECS compartment).

2.2. Bloch–Torrey PDE

In diffusion MRI, a time-varying magnetic field gradient is applied to the tissue to encode water diffusion. Denoting the effective time profile of the diffusion-encoding magnetic field gradient by $f(t)$ and the magnetic field gradient by $\boldsymbol{g}$, the restriction of the complex-valued transverse water proton magnetization $M(\boldsymbol{x}, t)$ in the ith compartment Ω_i by $M^i(\boldsymbol{x}, t)$, the governing equation is the Bloch–Torrey equation (Torrey 1956):

$$\frac{\partial}{\partial t} M^i(\boldsymbol{x}, t) = \nabla \cdot D_i \nabla M^i(\boldsymbol{x}, t) - I\gamma f(t) \boldsymbol{g} \cdot \boldsymbol{x}\, M^i(\boldsymbol{x}, t), \qquad \boldsymbol{x} \in \Omega_i, \tag{1}$$

$$D_i \nabla M^i(\boldsymbol{x}, t) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = -D_j \nabla M^j(\boldsymbol{x}, t) \cdot \boldsymbol{n}_j(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Gamma_{ij}, \tag{2}$$

$$D_i \nabla M^i(\boldsymbol{x}, t) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = \kappa_{ij}(M^j(\boldsymbol{x}, t) - M^i(\boldsymbol{x}, t)), \qquad \boldsymbol{x} \in \Gamma_{ij}, \tag{3}$$

$$D_i \nabla M^i(\boldsymbol{x}, t) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = 0, \qquad \boldsymbol{x} \in \Sigma_i, \tag{4}$$

$$M^i(\boldsymbol{x}, 0) = \rho, \qquad \boldsymbol{x} \in \Omega_i, \tag{5}$$

where $\gamma = 2.67513 \times 10^8$ rad s^{-1}T^{-1} is the gyromagnetic ratio of the water proton, I is the imaginary unit, D_i is the intrinsic diffusion coefficient in the compartment Ω_i, $\boldsymbol{n}(\boldsymbol{x})$ is the unit outward pointing normal vector, ρ is the initial spin density and κ_{ij} is the permeability coefficient of the interface Γ_{ij}. We assume the initial spin density ρ is the same in all compartments and the interface permeability is non negative, i.e. $\kappa_{ij} = \kappa_{ji} \geqslant 0$.

The magnetization $M(\boldsymbol{x}, t)$ is a function of position $\boldsymbol{x}$ and time t, and depends on the diffusion gradient vector $\boldsymbol{g}$ and the time profile $f(t)$. For the interface between ith and jth compartments Γ_{ij}, the two interface conditions are the flux continuity and a condition that incorporates a permeability coefficient κ_{ij}. The outer boundary is homogeneous Neumann boundary condition and we assume that the initial condition is the same for all compartments.

Some commonly used time profiles (diffusion-encoding sequences) are the pulsed-gradient spin echo (PGSE) sequence (Stejskal and Tanner 1965, Callaghan and Stepišnik 1995) and the oscillating gradient spin echo (OGSE) sequence (Does *et al* 2003). For the simplicity, for most of the paper, we will only consider the PGSE sequence, with two rectangular pulses of duration δ, separated by a time interval $\Delta - \delta$, for which the profile $f(t)$ is

$$f(t) = \begin{cases} 1, & 0 \leqslant t \leqslant \delta, \\ -1, & \Delta < t \leqslant \Delta + \delta, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

where $t = 0$ is the starting time of the first gradient pulse and $T_E = \delta + \Delta$ is the echo time at which the signal is measured. The diffusion MRI signal due to spins in the domain Ω is the space integral of magnetization, measured at echo time T_E:

$$S(\boldsymbol{g}, f) = \int_{\boldsymbol{x} \in \Omega} M(\boldsymbol{x}, T_E) d\boldsymbol{x}. \tag{7}$$

In a diffusion MRI experiment, the pulse sequence (time profile $f(t)$) is usually fixed, while $\boldsymbol{g}$ is varied in amplitude (and possibly also in direction). When $\boldsymbol{g}$ varies only in amplitude (while staying in the same direction), S is plotted against a quantity called the b-value. The b-value depends on $\boldsymbol{g}$ and $f(t)$ and is defined as

$$\mathrm{b}(\boldsymbol{g}) = \gamma^2 \|\boldsymbol{g}\|^2 \int_0^{T_E} du \left(\int_0^u f(s) ds \right)^2. \tag{8}$$

For PGSE, by replacing equation (6) into equation (8), the b-value is Stejskal and Tanner (1965):

$$\mathrm{b}(\boldsymbol{g}, \delta, \Delta) = \gamma^2 \|\boldsymbol{g}\|^2 \delta^2 (\Delta - \delta/3). \tag{9}$$

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

2.3. Matrix formalism representation

The matrix formalism solution of the Bloch–Torrey equation uses the basis of Laplace eigenfunctions on Ω. Let $\phi_k(\boldsymbol{x})$ and λ_k, $k = 1, \ldots$, be the L^2-normalized eigenfunctions and eigenvalues associated to the Laplace operator on Ω satisfying interface conditions and outer boundary condition above:

$$-\nabla \cdot D_i \nabla \phi_k^i(\boldsymbol{x}) = \lambda_k \phi_k^i(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Omega_i, \tag{10}$$

$$D_i \nabla \phi_k^i(\boldsymbol{x}) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = -D_j \nabla \phi_k^j(\boldsymbol{x}) \cdot \boldsymbol{n}_j(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Gamma_{ij}, \tag{11}$$

$$D_i \nabla \phi_k^i(\boldsymbol{x}) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = \kappa_{ij}(\phi_k^j(\boldsymbol{x}) - \phi_k^i(\boldsymbol{x})), \qquad \boldsymbol{x} \in \Gamma_{ij}, \tag{12}$$

$$D_i \nabla \phi_k^i(\boldsymbol{x}) \cdot \boldsymbol{n}_i(\boldsymbol{x}) = 0, \qquad \boldsymbol{x} \in \Sigma_i, \tag{13}$$

where $\phi_k^i(\boldsymbol{x})$ denotes the restriction of $\phi_k(\boldsymbol{x})$ to compartment Ω_i for $i \in \{1, \ldots, N_{cmpt}\}$. One remark is that this set of L^2-normalized eigenfunctions is orthogonal since the permeability coefficient is the same on the both sides of interfaces. We assume the non-negative real-valued eigenvalues are ordered in non-decreasing order:

$$0 = \lambda_1 \leqslant \lambda_2 \leqslant \lambda_3 \leqslant \ldots \tag{14}$$

We will suppose that Ω is connected. If all permeability coefficients are strictly positive, then only the first eigenvalue will be zero and the corresponding eigenfunction will be the constant function on Ω. If all permeability coefficients are zero, then the first N_{cmpt} eigenvalues will be zero and there will be N_{cmpt} corresponding constant eigenfunctions supported on each compartment. For the ease of the presentation, we

will limit ourselves to these two cases, and define N_{group} as the number of constant eigenfunctions in the basis. Clearly, $N_{group} = 1$ in the former case, and $N_{group} = N_{cmpt}$ in the latter case.

Let $\boldsymbol{L}$ be the diagonal matrix containing the first N_{eig} Laplace eigenvalues:

$$\boldsymbol{L} = \mathrm{diag}(\lambda_1, \lambda_2,\ldots,\lambda_{N_{eig}}) \in \mathbb{R}^{N_{eig}\times N_{eig}}. \tag{15}$$

Denoting by

$$\Phi(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}),\ldots,\phi_{N_{eig}}(\boldsymbol{x})),$$

the vector of Laplace eigenfunctions corresponding to the first N_{eig} eigenvalues. Because $\{\phi_k(\boldsymbol{x})\}_{k\ =\ 1,2,3,\ldots}$ is a complete basis on Ω with the correct interfaces and boundaries conditions, we decompose the magnetization $M(\boldsymbol{x}, t)$ in this basis as

$$M(\boldsymbol{x}, t) \approx \sum_{k=1}^{N_{eig}} T_k(t)\phi_k(\boldsymbol{x}) = \Phi(\boldsymbol{x})\mathbf{T}(t), \tag{16}$$

with the time dependent coefficient column vector

$$\mathbf{T}(t) = (T_1(t), T_2(t),\ldots,T_{N_{eig}}(t))^T.$$

Substituing equation (16) into Bloch–Torrey equation, multiplying both sides with $\phi_l(\boldsymbol{x})$ and integrating over Ω gives

$$\frac{\partial}{\partial t}T_l(t) = -\lambda_l T_l(t) - I\gamma\sum_{k=1}^{N_{eig}} T_k(t)\int_\Omega \boldsymbol{g}\cdot\boldsymbol{x}\phi_k(\boldsymbol{x})\phi_l(\boldsymbol{x})d\boldsymbol{x},\ l = 1, 2,\ldots,N_{eig}. \tag{17}$$

Define

$$\boldsymbol{W}(\boldsymbol{g}) \coloneqq g_x\boldsymbol{A}^x + g_y\boldsymbol{A}^y + g_z\boldsymbol{A}^z, \tag{18}$$

where $\boldsymbol{g} = (g_x, g_y, g_z)^T$ is the encoding gradient vector and $\boldsymbol{A}^x$, $\boldsymbol{A}^y$ and $\boldsymbol{A}^z$ are three symmetric $N_{eig}\times N_{eig}$ matrices whose entries are the first order moments in the coordinate directions of the product of pairs of eigenfunctions:

$$\boldsymbol{A}^r_{kl} \coloneqq \int_{\boldsymbol{x}\in\Omega} r\phi_k(\boldsymbol{x})\phi_l(\boldsymbol{x})d\boldsymbol{x},\ (k, l) \in \{1,2,\ldots,N_{eig}\}^2,\ r \in \{x, y, z\}. \tag{19}$$

Then the Bloch–Torrey operator $-\nabla\cdot D_i\nabla + I\gamma f(t)\boldsymbol{g}\cdot\boldsymbol{x}$ in the Laplace eigenfunction basis is given by the complex-valued matrix

$$\boldsymbol{L} + I\gamma f(t)\boldsymbol{W}(\boldsymbol{g}), \tag{20}$$

and equation (17) can be written as a system of ordinary differential equations below

$$\frac{d}{dt}\mathbf{T}(t) = -(\boldsymbol{L} + I\gamma f(t)\boldsymbol{W}(\boldsymbol{g}))\mathbf{T}(t). \tag{21}$$

Define

$$\boldsymbol{H}(\boldsymbol{g}, f) = e^{-\delta(\boldsymbol{L}-I\gamma\boldsymbol{W}(\boldsymbol{g}))}\cdot e^{-(\Delta-\delta)\boldsymbol{L}}\cdot e^{-\delta(\boldsymbol{L}+I\gamma\boldsymbol{W}(\boldsymbol{g}))}, \tag{22}$$

and denote

$$\mathbf{T}(0) = \rho\left(\int_\Omega \phi_1(\boldsymbol{x})d\boldsymbol{x}, \int_\Omega \phi_2(\boldsymbol{x})d\boldsymbol{x},\ldots,\int_\Omega \phi_{N_{eig}}(\boldsymbol{x})d\boldsymbol{x}\right)^T \in \mathbb{R}^{N_{eig},1},$$

the vector of coefficients of the initial condition projected onto the eigenfunctions of the Laplace operator. It is easy to show that the integral of the eigenfunctions over Ω will be zero except for the constant functions,

$$\int_\Omega \phi_i(\boldsymbol{x})d\boldsymbol{x} = \begin{cases}\sqrt{\Omega_i}, & \text{if } \phi_i(\boldsymbol{x}) \text{ is a constant function,}\\ 0, & \text{otherwise.}\end{cases}$$

The magnetization measured at the echo time is

$$M(\boldsymbol{x}, T_E) = \Phi(\boldsymbol{x})\boldsymbol{H}(\boldsymbol{g}, f)T(0). \tag{23}$$

The signal is computed by integrating the magnetization over Ω:

$$S(\boldsymbol{g}, f; N_{eig}) = \int_\Omega \Phi(\boldsymbol{x})\boldsymbol{H}(\boldsymbol{g}, f)\mathbf{T}(0)d\boldsymbol{x}. \tag{24}$$

2.4. The numerical matrix formalism method

In order to numerically implement the matrix formalism method, we discretize the Laplace operator with permeable interface conditions using $P1$ finite elements. The method is described in a previous publication

(Li *et al* 2020). We discretize Ω into a finite element mesh and use $P1$ basis functions $\{\varphi_p(\boldsymbol{x})\}_{p\in\{1,\ldots,N_{node}\}}$, where N_{node} is the number of nodes, to construct the following finite element matrices: $\boldsymbol{M}$, $\boldsymbol{K}$, $\boldsymbol{Q} \in \mathbb{R}^{N_{node}\times N_{node}}$, known as the mass, stiffness and flux matrices, respectively,

$$\boldsymbol{M}_{pq} = \int_\Omega \varphi_p(\boldsymbol{x})\varphi_q(\boldsymbol{x})d\boldsymbol{x}, \tag{25}$$

$$\boldsymbol{K}_{pq} = \begin{cases} \int_\Omega D_i \nabla\varphi_p(\boldsymbol{x})\nabla\varphi_q(\boldsymbol{x})d\boldsymbol{x},\ (p,\,q) \in \mathcal{I}_i^2,\ i \in \{1,\ldots,N_{cmpt}\}, \\ 0, \quad \text{otherwise}, \end{cases} \tag{26}$$

$$\boldsymbol{Q}_{pq} = \sum_{i=1}^{N_{cmpt}}\sum_{j=1}^{N_{cmpt}} \boldsymbol{Q}_{pq}^{ij},$$

$$\boldsymbol{Q}_{pq}^{ij} = \begin{cases} \kappa_{ij}\int_{\Gamma_{ij}} \varphi_p(\mathbf{x})\varphi_q(\mathbf{x})d\Gamma(\mathbf{x}), & (p,\,q) \in \mathcal{I}_i^2,\ i \in \{1,\ldots,N_{cmpt}\}, \\ -\kappa_{ij}\int_{\Gamma_{ij}} \varphi_p(\mathbf{x})\varphi_q(\mathbf{x})d\Gamma(\mathbf{x}), & (p,\,q) \in \mathcal{I}_i \times \mathcal{I}_j,\ (i,\,j) \in \{1,\ldots,N_{cmpt}\}^2,\ i \neq j, \\ 0, & \text{otherwise}, \end{cases} \tag{27}$$

where $\mathcal{I}_i$ is the set of nodes index belonging to ith compartment. In order to implement the flux matrix, double nodes are placed at the interfaces (Li *et al* 2019, 2020).

We discretize the eigenfunctions of the Laplace operator with permeable interface condition in the P1 basis functions,

$$\phi_k(\boldsymbol{x}) = \sum_{n=1}^{N_{node}} \boldsymbol{P}_{n,k}\varphi_n(\boldsymbol{x}),\ n \in \{1,\,2,\ldots,N_{node}\},\ k \in \{1,\ldots,N_{eig}\}, \tag{28}$$

where $\boldsymbol{P} \in \mathbb{R}^{N_{node}\times N_{eig}}$, and the entry $\boldsymbol{P}_{n,k}$ is the coefficient of eigenfunction ϕ_k in the basis function φ_n.

The finite elements discretization described above changes the continuous Laplace operator eigenvalue problem in equation (10)–(13) to a discrete, generalized matrix eigenvalues problem: find the first N_{eig} eigenvalues $\boldsymbol{L}$ and corresponding eigenfunctions $\boldsymbol{P}$, such that

$$(\boldsymbol{K} + \boldsymbol{Q})\boldsymbol{P} = \boldsymbol{M}\boldsymbol{P}\boldsymbol{L}, \tag{29}$$

where $\boldsymbol{L} \in \mathbb{R}^{N_{eig}\times N_{eig}}$ is a diagonal matrix whose diagonal terms are eigenvalues of Laplace operator with permeable interface conditions.

The integrals of the finite element discretized eigenfunctions are given by

$$\int_\Omega \Phi(\boldsymbol{x})d\boldsymbol{x} = \boldsymbol{P}^T\boldsymbol{M}\mathbf{1}_{N_{node},1}, \tag{30}$$

where $\mathbf{1}_{N_{node},1}$ is column vector of all ones with size N_{node} and

$$\begin{aligned} T(0) &= \rho\int_\Omega \Phi(\boldsymbol{x})d\boldsymbol{x} \\ &= \rho\boldsymbol{P}^T\boldsymbol{M}\mathbf{1}_{N_{node},1} \\ &= [\sqrt{\Omega_1},\ldots\sqrt{\Omega_{N_{group}}},\,0,\ldots,0]^T \in \mathbb{R}^{N_{eig},1}, \end{aligned} \tag{31}$$

where N_{group} is the number of constant eigenfunctions.

Similarly, with this discretization, the matrices $\boldsymbol{A}^x$,$\boldsymbol{A}^y$ and $\boldsymbol{A}^z$ in equation (19) can be rewritten as

$$\boldsymbol{A}^r = \boldsymbol{P}^T\boldsymbol{J}^r\boldsymbol{P},\ r \in \{x,\,y,\,z\}, \tag{32}$$

where the entries of $\boldsymbol{J}^r$ are

$$\boldsymbol{J}_{kl}^r = \int_\Omega r\varphi_k\varphi_l d\boldsymbol{x},\ \ r \in \{x,\,y,\,z\}. \tag{33}$$

We define

$$\boldsymbol{J}(\boldsymbol{g}) \equiv g_x\boldsymbol{J}^x + g_y\boldsymbol{J}^y + g_z\boldsymbol{J}^z \in \mathbb{R}^{N_{node},N_{node}}, \tag{34}$$

$\boldsymbol{W}(\boldsymbol{g})$ can be rewritten as

$$\boldsymbol{W}(\boldsymbol{g}) = \sum_{r=\{x,y,z\}} g_r\boldsymbol{A}^r = \sum_{r=\{x,y,z\}} g_r\boldsymbol{P}^T\boldsymbol{J}^r\boldsymbol{P} = \boldsymbol{P}^T\boldsymbol{J}(\boldsymbol{g})\boldsymbol{P}. \tag{35}$$

Substituting $\boldsymbol{L}$, $\boldsymbol{P}$ and $\boldsymbol{J}(\boldsymbol{g})$ into $\boldsymbol{H}(\boldsymbol{g}, f)$:

$$\boldsymbol{H}(\boldsymbol{g}, f) = e^{-\delta(\boldsymbol{L}-I\gamma\boldsymbol{W}(\boldsymbol{g}))} \cdot e^{-(\Delta-\delta)\boldsymbol{L}} \cdot e^{-\delta(\boldsymbol{L}+I\gamma\boldsymbol{W}(\boldsymbol{g}))}. \tag{36}$$

The numerical matrix formalism diffusion MRI signal at echo time is the following:

$$S^{\mathrm{NMF}}(\boldsymbol{g}, f; N_{eig}) = \rho(\mathbf{1}^T_{N_{node},1}\boldsymbol{MP})\boldsymbol{H}(\boldsymbol{g}, f)(\boldsymbol{P}^T\boldsymbol{M}\mathbf{1}_{N_{node},1}) = \rho\ \mathbf{T}(0)^T\boldsymbol{H}(\boldsymbol{g}, f)\mathbf{T}(0). \tag{37}$$

2.4.1. Choice of N_{eig}

When $N_{eig} = N_{node}$, the full set of discretized eigenfunctions will be used to compute the diffusion MRI signal. In practice, the large eigenvalues have little contribution to the physics of diffusion. Using the eigenfunctions whose spatial scales are on the order of the cell structure and the diffusion distance can yield a good approximation, and result in a $N_{eig} \ll N_{node}$.

When computing the diffusion MRI signal from the numerical matrix formalism method, we do not choose N_{eig} arbitrarily. Rather, we impose a length scale cut-off L_s that is appropriate for the geometry and for the MRI experiment. For realistic MRI experimental parameters and brain cell geometries, the smallest length scales that can influence the diffusion MRI signal is around 1 μm. We refer to the works (Li *et al* 2020, Agdestein *et al* 2021) for details about the numerical matrix formalism method and how to choose the length scale cut-off. To translate the eigenvalues into length scales, we use the expression

$$l(\lambda) = \begin{cases} \infty, & \lambda = 0, \\ \pi\sqrt{D/\lambda}, & \lambda > 0. \end{cases} \tag{38}$$

Typically, $l(\lambda)$ is related to the wavelength of the oscillations in the corresponding eigenfunction. The number N_{eig} associated with this length scale choice is determined by the relation

$$\infty \geqslant l(\lambda_1) \geqslant \cdots \geqslant l(\lambda_{N_{eig}}) \geqslant L_s > l(\lambda_{N_{eig}+1}) \geqslant \cdots > 0. \tag{39}$$

Thus, we use interchangeably N_{eig} or L_s to indicate the truncation of the eigenfunctions.

Remark 1. It is to be noted that for the same geometry and the same L_s, when using the numerical matrix formalism method, the resulting N_{eig} is smaller at higher permeability.

Remark 2. Even though eigendecomposition routines can accept an eigenvalue range (length scale cut-off) as input, it is more computationally efficient to input the number of desired eigenvalues and then keep those eigenvalues within the length scale cut-off. This is what we do in practice to compute a subset of eigenfunctions.

3. New formulation using the impermeable Laplace eigenfunctions

The main aim of our paper is to derive a new formulation of the numerical matrix formalism method for permeable interfaces, using the eigenvalues and eigenfunctions of Laplace operator from the impermeable case.

Suppose that all the interfaces are impermeable, thenequation (29)becomes:

$$\boldsymbol{KP}_{imp} = \boldsymbol{MP}_{imp}\boldsymbol{L}_{imp}. \tag{40}$$

The subscript *imp* indicates these matrices are from the impermeable case. We now want to use these two matrix $\boldsymbol{L}_{imp}$ and $\boldsymbol{P}_{imp}$ as well as

$$\boldsymbol{W}_{imp}(\boldsymbol{g}) \equiv g_x\boldsymbol{P}^T_{imp}\boldsymbol{J}^x\boldsymbol{P}_{imp} + g_y\boldsymbol{P}^T_{imp}\boldsymbol{J}^y\boldsymbol{P}_{imp} + g_z\boldsymbol{P}^T_{imp}\boldsymbol{J}^z\boldsymbol{P}_{imp} \tag{41}$$

$$= \boldsymbol{P}^T_{imp}\boldsymbol{J}(\boldsymbol{g})\boldsymbol{P}_{imp}, \tag{42}$$

to obtain the diffusion MRI signal in the presence of permeable interfaces.

Assume that for $i \neq j$, $(i, j) \in \{1,\ldots,N_{cmpt}\}^2$, the interfaces are permeable: $\kappa_{ij} > 0$, and denoting the corresponding flux matrix by $\boldsymbol{Q}$. We recall that $\boldsymbol{Q}$ for the permeable case is defined by equation (27). We define a new matrix,

$$\boldsymbol{Q}_{proj} \equiv \boldsymbol{P}^T_{imp}\boldsymbol{Q}\boldsymbol{P}_{imp} \in \mathbb{R}^{N_{eig},N_{eig}}, \tag{43}$$

the projection of the flux matrix onto the eigenfunctions of the Laplace operator with impermable interfaces conditions. Adding $\boldsymbol{Q}_{proj}$ to the diagonal matrix $\boldsymbol{L}_{imp}$, we define a new matrix (in general not diagonal)

$$\boldsymbol{L}_{proj} \equiv \boldsymbol{L}_{imp} + \boldsymbol{Q}_{proj},$$

as well as

$$\boldsymbol{H}_{proj}(\boldsymbol{g}, f) \equiv e^{-\delta(\boldsymbol{L}_{proj} - I\gamma\boldsymbol{W}_{imp}(\boldsymbol{g}))} \cdot e^{-(\Delta-\delta)\boldsymbol{L}_{proj}} \cdot e^{-\delta(\boldsymbol{L}_{proj} + I\gamma\boldsymbol{W}_{imp}(\boldsymbol{g}))}. \tag{44}$$

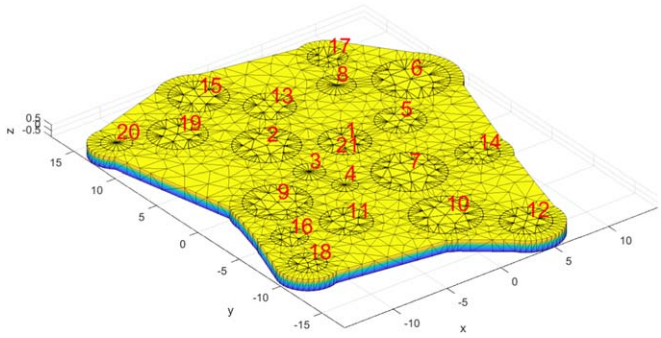

Figure 1. Finite element mesh of the geometry Ω^I. It contains 20 randomly placed cylindrical axons, wrapped in the ECS. The radii of all axons are between 1 μm and 3 μm and the height of all compartments is 1 μm. Dimension of the whole geometry is 34 μm $\times$ 29 μm $\times$ 1 μm. The geometry is generated by SpinDoctor and the finite elements mesh is created by Tetgen (Si 2015). The mesh contains 3455 nodes and 6673 elements.

Theorem 1. *In the presence of permeable interfaces, the expression*

$$\begin{aligned} S^{\text{NEW}}(\boldsymbol{g}, f; N_{eig}) &\equiv \rho(\mathbf{1}_{N_{node},1}^T \boldsymbol{M}\boldsymbol{P}_{imp})\boldsymbol{H}_{proj}(\boldsymbol{g}, f)(\boldsymbol{P}_{imp}^T \boldsymbol{M}\mathbf{1}_{N_{node},1}) \\ &= \rho \mathbf{T}_{imp}(0)^T \boldsymbol{H}_{proj}(\boldsymbol{g}, f)\mathbf{T}_{imp}(0), \end{aligned} \tag{45}$$

where

$$\mathbf{T}_{imp}(0) = [\sqrt{|\Omega_1|}, \sqrt{|\Omega_2|}, \ldots, \sqrt{|\Omega_{N_{cmpt}}|}, 0, \ldots, 0]^T \in \mathbb{R}^{N_{eig},1}, \tag{46}$$

is exactly equal to the diffusion MRI signal expression from the numerical matrix formalism method, if the full set of the eigenvalues and eigenfunctions is used for both methods. See Appendix for the proof of theorem 1.

3.1. Choice of N_{eig}

As with the original numerical matrix formalism method, the new method will not, in practice, require the use of the full set of eigenfunctions, and we will again have $N_{eig} \ll N_{node}$, with the choice of N_{eig} determined by the length scale cut-off L_s:

$$\infty \geqslant l(\lambda_1) \geqslant \cdots \geqslant l(\lambda_{N_{eig}}) \geqslant L_s > l(\lambda_{N_{eig}+1}) \geqslant \cdots > 0. \tag{47}$$

Remark 3. It is to be noted that for the same geometry and the same L_s, when using the new method, the resulting N_{eig} is the same no matter what the interface permeability, and it is usually somewhat larger than the N_{eig} of the numerical matrix formalism method (with the same value of L_s).

Remark 4. Any basis set, when it is complete, can represent any discretized solution in the finite elements basis. We used $P1$ finite elements, so any basis set is complete that has N_{node} elements. This means, the permeable Laplace eigenfunctions set and the impermeable Laplace eigenfunctions set are both sufficient to represent any PDE solution if $N_{eig} = N_{node}$.

The discretized solution of the Bloch–Torrey equation, permeable or not, is usually piecewise smooth (on each compartment) and so should be able to be represented by the smooth eigenfunctions in the impermeable basis. The discretized solution should not need to be represented by very oscillatory eigenfunctions, this means the vast majority of the oscillatory eigenfunctions in the impermeable basis are not needed. Thus, one can just keep the relatively smooth eigenfunctions in the impermeable basis and they are enough to represent any reasonable solution for the permeable problem. In short, for a discretized finite element solution of the Bloch–Torrey equation, one never needs to take N_{eig} to be anywhere close to N_{node} in any basis.

4. Numerical results

In this section we conduct a numerical validation of the new method. The generation of the computational geometries, the discretization into finite elements, and the numerical computation of the Laplace eigenfunctions in the finite element space were implemented using the SpinDoctor toolbox (Li *et al* 2019).

The simulations will be performed on a quasi-two dimensional multi-compartment geometry, denoted by Ω^I, shown in figure 1, containing 20 axons. The axons are randomly placed and then wrapped by an ECS. The ECS is not a rectangle in order to keep the axons closely packed. The axon radii vary between 1 μm and 3 μm.

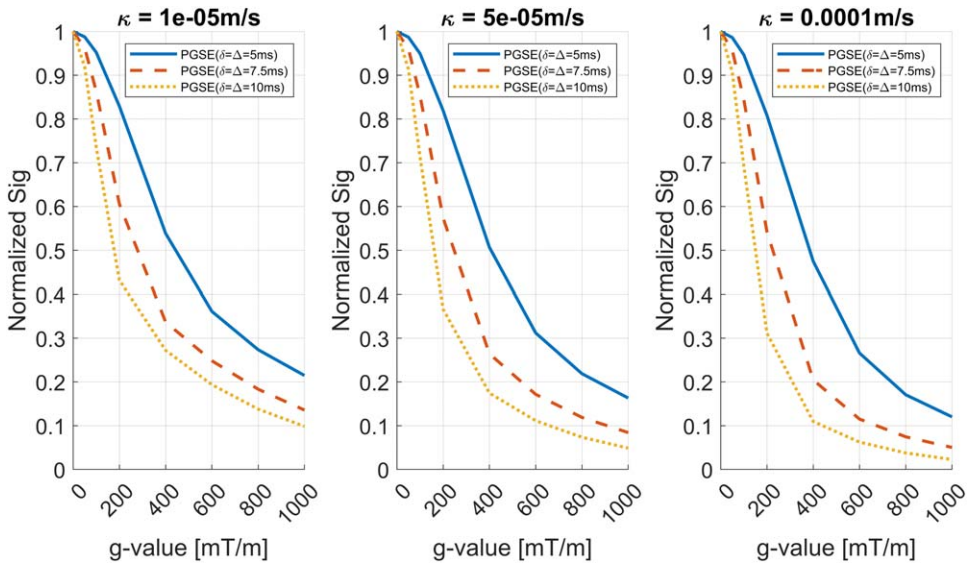

Figure 2. Normalized direction-averaged diffusion MRI signals as a function of the g-value $\|\boldsymbol{g}\|$. These are the reference solutions computed using the numerical matrix formalism method with the full set of permeable Laplce eigenfunctions, i.e. $N_{eig} = N_{node} = 3455$ eigenfunctions for Ω^I.

The dimension of the whole geometry is 34 μm $\times$ 29 μm $\times$ 1 μm. The diffusion coefficients are set to be the same for all compartments: $D_i = 2 \times 10^{-3}$ mm^2 s^{-1} for $i \in \{1, N_{cmpt}\}$. The initial spin density is set to $\rho = 1.0$. SpinDoctor creates the geometrical configuration and the surface triangulation, then pass the surface triangulation to TetGen (Si 2015) to create a volume mesh. The finite elements mesh contains $N_{node} = 3455$ nodes and 6673 elements.

In the literature, the experimentally measured permeability coefficient κ in biological cells ranges from 10^{-6} to 10^{-4} m s^{-1} (Grebenkov *et al* 2014), in particular, $\kappa = 10^{-5}$ m s^{-1} for axonal membranes (Chin *et al* 2002). Therefore, our simulations are performed using permeability coefficients up to $\kappa = 10^{-4}$ m s^{-1}. The average displacement in free diffusion is $\sqrt{2dDT_E}$, where d is the dimension. In order to reduce the amount of spins hitting the outer boundary, we want to keep the displacement to less than half of the geometry diameter, obtaining that $T_E \leqslant 20$ ms. We make this choice because the focus of this paper is on the permeable interfaces, so we want to reduce the effects from the interaction of spins with the impermeable outer boundary. Thus, we limit $\delta + \Delta \leqslant 20$ ms in the simulations. The gradient strength in *in-vivo* experiments does not exceed 1000 mT m^{-1} (Huang *et al* 2021), so we set the highest simulated g-value to 1000 mT m^{-1}.

4.1. Computing the reference solution

For the geometry Ω^I, we do not have the analytical solution of the diffusion MRI signal. We propose using the numerical formalism method with the full set of permeable Laplace eigenfunctions as the reference solution. We have compared the reference solution to the finite elements solution of the discretized Bloch–Torrey PDE and verified that for the simulations we performed, the relative differences between the two are less 0.002%. Thus, we estimate that the signals computed using the numerical matrix formalism method with the full set of permeable eigenfunctions to be accurate to 0.002% from the true signal.

The reference solution is set to be the numerical matrix formalism solution using the full set of permeable Laplce eigenfunctions i.e. $N_{eig} = N_{node} = 3455$ eigenfunctions for Ω^I,

$$S^{\mathrm{REF}}(\boldsymbol{g}, f) = S^{\mathrm{NMF}}(\boldsymbol{g}, f; N_{node}).$$

To avoid the dependence of the results on the gradient direction $\boldsymbol{g}$, we average the diffusion MRI signal over 18 gradient directions, uniformly distributed on a unit semicircle in the $x - y$ plane(equivalent to 36 gradient directions on the unit circle):

$$\overline{S}(\|\boldsymbol{g}\|, f; N_{eig}) = \frac{1}{|\Omega|}\frac{1}{18}\sum_{d=1}^{18} S(\boldsymbol{g}_d, f; N_{eig}), \quad \boldsymbol{g}_d = \|\boldsymbol{g}\|\left[\cos\left(\pi\frac{d}{18}\right), \sin\left(\pi\frac{d}{18}\right), 0\right]^T, \tag{48}$$

normalized by the total volume. Figure 2 depicts the reference signals as a function of the g-value $\|\boldsymbol{g}\|$. The simulations are performed for g-value from 0 mT m^{-1} to 1000 mT m^{-1}. The signals $\overline{S}$ have been normalized by the total volume so their maximum value is 1. We observe that diffusion MRI signal decays faster in presence of more permeable membranes.

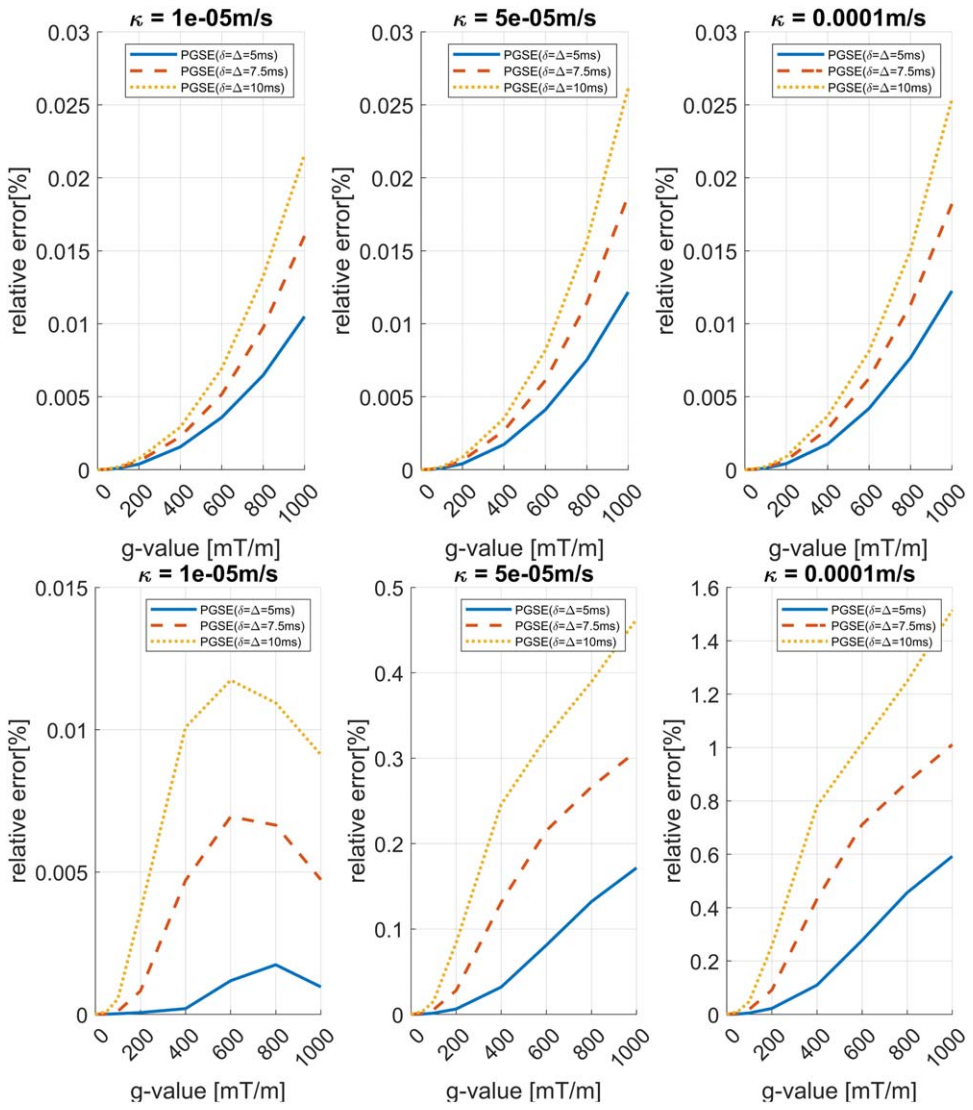

Figure 3. First row: relative errors of direction averaged signals of the numerical matrix formalism with the length scale cut-off $L_s = 1\ \mu$m ($\kappa = 10^{-4}$ m s^{-1}, $N_{eig} = 532$; $\kappa = 5 \times 10^{-5}$ m s^{-1}, $N_{eig} = 535$; $\kappa = 10^{-5}$ m s^{-1}, $N_{eig} = 538$). Second row: relative errors of direction averaged signals of the new method with the length scale cut-off $L_s = 1\ \mu$m ($N_{eig} = 538$). The relative errors are in percent. Simulations are performed on Ω^I, by SpinDoctor. Left: $\kappa = 10^{-5}$ m s^{-1}; middle: $\kappa = 5 \times 10^{-5}$ m s^{-1}; right: $\kappa = 10^{-4}$ m s^{-1}.

4.2. Validation of the new method

We have shown in theorem 1 that the new method yields the same signal as the numerical matrix formalism method if the full set of basis functions is used. However, since in practice, $N_{eig} \ll N_{node}$, we will now show the accuracy of the two methods for fixed values of the length scale cut-off L_s.

In figure 3, we show the relative errors produced by the two methods compared to the reference solution. The relative error is defined as:

$$\epsilon_{rel}(\%) = 100 \times \frac{|\bar{S}(\|\boldsymbol{g}\|, f; N_{eig}) - \bar{S}^{\mathrm{REF}}(\|\boldsymbol{g}\|, f)|}{\bar{S}^{\mathrm{REF}}(\|\boldsymbol{g}\|, f)}. \tag{49}$$

First, we see that the relative errors of the original numerical matrix formalism method where the length scale cut-off is $L_s = 1\ \mu$m are under 0.03% for all the simulated sequences. The number of the eigenfunctions differs with permeability: when $\kappa = 10^{-5}$ m s^{-1}, 5×10^{-5} m s^{-1}, 10^{-4} m s^{-1}, $N_{eig} = 538, 535, 532$, respectively. Second, for the new method, with a length scale cut-off of $L_s = 1\ \mu$m, resulting in $N_{eig} = 538$, the relative error is under 1.5% for all the sequences. The relative error increases as permeability increases, as the g-value $\|\boldsymbol{g}\|$ increases, and as the diffusion time increases.

Now we study the convergence behavior of the new method as L_s decreases (N_{eig} increases) and compare it to the original numerical matrix formalism method. The simulated gradient direction is fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$ and we use the longest sequence PSGSE (10 ms, 10 ms), which yields the biggest errors. We

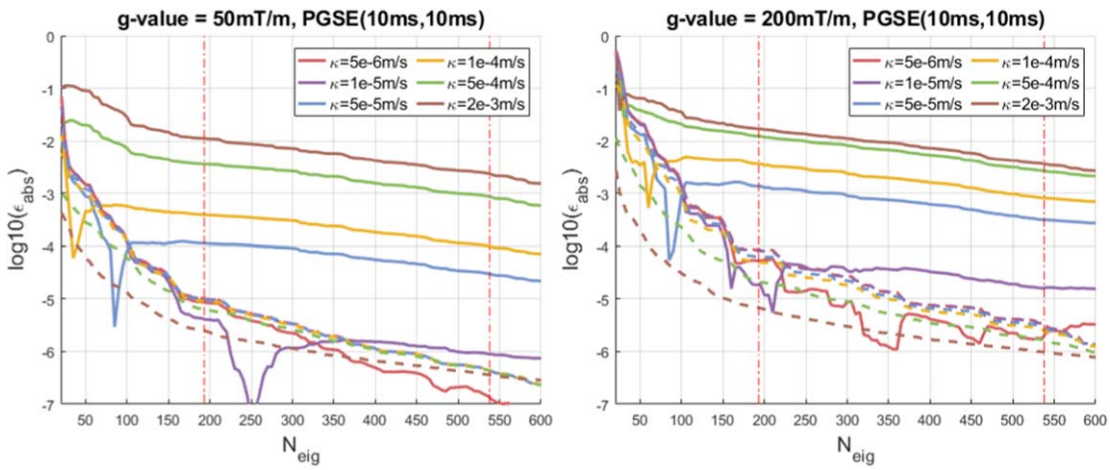

Figure 4. Normalized signal errors of the new method (in solid line) and the numerical matrix formalism method (in dashed line), compared to the reference solution, as a function of N_{eig}. The two vertical lines indicate where the truncations occur for $L_s = 2$ μm ($N_{eig} = 193$) and $L_s = 1$ μm ($N_{eig} = 538$). The simulations are performed on Ω^I, with gradient direction fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$ and the sequence PGSE (10 ms, 10 ms). Left: g-value = 50 mT m^{-1}; right: g-value = 200 mT m^{-1}.

define the normalized error between the computed signal and reference signal to be

$$\epsilon_{abs} = \frac{|S(\boldsymbol{g}, f; N_{eig}) - S^{\mathrm{REF}}(\boldsymbol{g}, f)|}{|\Omega|}. \tag{50}$$

It is a normalized error because $S/|\Omega|$ is always bounded by 1. Note this is not a relative error, we do not divide by S^{REF}, because we do not want the error to increase due to the decrease in the signal itself, we simply want to show the convergence of the signals.

Figure 4 shows the normalized errors of the diffusion MRI signals computed by the new method and by the original numerical matrix formalism method, compared to the reference solution. The x-axis gives N_{eig}. The two vertical lines indicate where the truncations occur for $L_s = 2$ μm and $L_s = 1$ μm in the impermeable case:

$$L_s = 2\ \mu\mathrm{m},\ N_{eig} = 193,$$
$$L_s = 1\ \mu\mathrm{m},\ N_{eig} = 538.$$

As more eigenfunctions are used, the errors are reduced for both the new method and the original numerical matrix formalism method. When $\kappa \leqslant 10^{-5}$ m s^{-1}, the new method converges at a similar rate as the original numerical matrix formalism. At thehigher permeabilities, the new method converges more slowly than numerical matrix formalism, but it is clear that if we are interested 2 or 3 digits of accuracy, which is reasonable given that the diffusion MRI signal noise is at least of order 0.01, using the length scale cut-off of $L_s = 2$ μm is sufficient. As far as we know, there is not an analytical way to relate the truncation size and the signal error. The truncation is defined on the impermeable Laplace eigenfunctions, whereas the signal is related to the operator including the term $i\boldsymbol{g} \cdot \boldsymbol{x}$ as well as the permeability. We observe that though the errors of the numerical matrix formalism method increases with lower permeability due to the large variations of the permeable eigenfunctions around the interfaces, the errors of the new method increases with higher permeability due to the fact that the new method uses impermeable eigenfunctions for all permeability values.

4.3. Computational time

An advantage of the new method to compute the diffusion MRI signal is the savings in computational time. To show the efficiency of the new method, we compare the computational times of the numerical matrix formalism method and our new method on a bigger geometry $\Omega^{\mathrm{axons200}}$ that contains 200 cylindrical axons enclosed in the ECS, as shown in figure 5. Its dimensions are 98 μm $\times$ 118 μm $\times$ 1 μm and there are 62 145 elements and $N_{node} = 32\,023$ nodes in total, of which 16 924 nodes in the ECS. The gradient direction of simulations is fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$. Two PGSE sequences PGSE (5 ms, 5 ms) and PGSE (10 ms, 10 ms) and four $g-$ value = [50, 200, 500, 1000] mT m^{-1} are used. The simulations are performed with 3 different values of $N_{eig} = 2000$, 4000, 5000 (we fixed N_{eig} rather than L_s to make easier comparisons of computational time). All the simulations are performed on a computing server with 20 cores of frequency 2.4 GHz, and RAM of 256 GB. The operating system is Rocky Linux 8 and the Matlab version is R2021a.

numerical matrix formalism with the full set of eigenfunctions is set to be the reference solution. Both matrix formalism and the new method compute the diffusion MRI signal in two steps: Laplace eigendecomposition and matrix exponential computations. The first step is independent of the encoding sequence settings, involving only

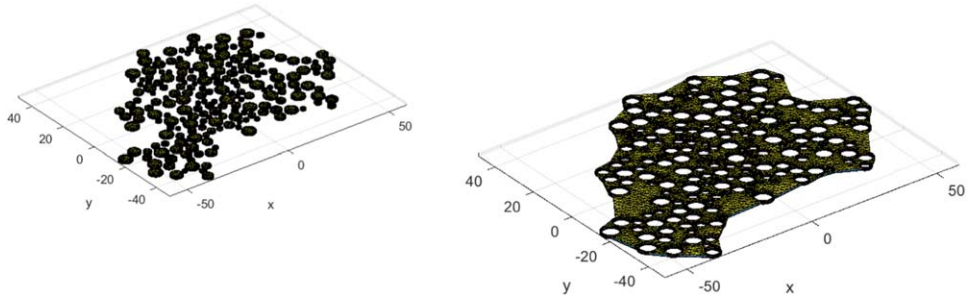

Figure 5. Finite element meshes of the geometry Ω^{axons200} for the computational times comparison. The geometry contains 200 randomly place cylindrical axons, whose radii vary between 1 μm and 3 μm, and one tightly wrapped ECS. All compartments are 1 μm in height. The dimensions are 98 μm $\times$ 118 μm $\times$ 1 μm. This mesh has 32 023 nodes and 62 145 elements in total. Left: 200 axons compartments; right: ECS compartment, which contains 16 924 nodes and 34 258 elements.

Table 1. Computational times of Laplace eigen-decomposition at different permeabilities for Ω^{axons200}, given in seconds. The full set contains $N_{node} = 32\,023$ nodes.

	Computational time (seconds)			
	$N_{eig} = 2000$	$N_{eig} = 4000$	$N_{eig} = 5000$	Full set ($N_{eig} = 32\,023$)
	New method			
Total	41	171	278	141
$\kappa(m/s)$	numerical matrix formalism method			
10^{-5}	99	419	646	723
5×10^{-5}	100	389	518	727
10^{-4}	102	353	625	734
Total	301	1161	1789	2184

sparse matrices of size $N_{node} \times N_{node}$. The second step involves dense matrices of size $N_{eig} \times N_{eig}$. In practice, we have $N_{eig} \ll N_{node}$.

Table 1 shows the computational times of the Laplace eigendecompositions by the new method and the numerical matrix formalism method. The eigenmodes are computed by the Matlab built-in function 'eigs', which computes the first smallest N_{eig} eigenmodes by Lanczos iteration. To obtain the full set of the eigenmodes, the Matlab built-in function 'eig' can be used to conduct a complete eigen-decomposition. The computational complexity of 'eig' for the generalized eigenvalue problem, $Ay = By\lambda$, is $O(N_{node}^3)$ in theory, and $O(N_{node}^{2.376})$ in practice using the Coppersmith and Winograd algorithm (Coppersmith and Winograd 1990). The computational complexity of 'eigs' is $O(N_{eig}N_{node}^2 + N_{node}N_{eig}^2)$, the first term is due to the computation of $B\backslash Ay$ at each Lanczos iteration and the second term is due to the orthogonalization of the new Krylov vectors at each Lanczos iteration (Lee *et al* 2009). In fact, because A and B are sparse matrices, the computation of $B\backslash Ay$ is $O(N_{eig}N_{node})$ rather than $O(N_{eig}N_{node}^2)$, so the dominant term of the computational complexity for 'eigs' is $O(N_{node}N_{eig}^2)$. We can see in the table that going from $N_{eig} = 2000$ to $N_{eig} = 4000$, the computational times increase by 4 in all the rows.

The original numerical matrix formalism method needs to recalculate permeable Laplace eigenfunctions when the permeability changes. On the contrary, the new method only computes the impermeable Laplace eigenfunctions once. In addition, the impermeable Laplace eigendecomposition can be achieved compartment by compartment. The computational complexity of the permeable eigendecomposition is $\mathcal{O}(N_{node}^{2.376})$ ('eig') or $\mathcal{O}(N_{node}N_{eig}^2)$ ('eigs'), compared to the impermeable case, where it is $\mathcal{O}(\sum_i^{N_{cmpt}} N_{i,node}^{2.376})$ ('eig') or $\mathcal{O}(\sum_i^{N_{cmpt}}(N_{i,node}N_{eig}^2))$ ('eigs'), $N_{i,node}$ being the number of finite elements nodes in compartment i. In table 1, we can see that for the same N_{eig}, the impermeable eigendecomposition istwo times faster than the permeable eigendecomposition.If we consider the simulation of three permeability values, using the new method, the full set eigendecomposition can be done in 141 s, whereas the numerical matrix formalism method takes 301 s at $N_{eig} = 2000$.

On the question of whether to call 'eigs' or 'eig' to compute the eigenmodes, we remind the reader that the theoretical complexities are $O(N_{node}N_{eig}^2)$ and $O(N_{node}^{2.376})$, respectively. It is clear that, at some point, as N_{eig} increases, it would be more computationally efficient to compute the full eigendecomposition instead of a partial eigendecomposition. Some further considerations are that (1) the 'eig' implementation in Matlab is well optimized for parallel computing using all the computer's cores, unlike the 'eigs', (2), the 'eig' function in MATLAB only accepts dense matrices whereas the 'eigs' function allows the designation of sparse matrices so the matrix-vector multiplications

Table 2. Computational times and normalized signal errors of the numerical matrix formalism (NMF) method and the new method in Ω^{axons200}, given in seconds. The number of eigenfunctions is $N_{eig} = 2000$. The encoding gradient direction is fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$. The units are κ: m s^{-1}, δ: ms, Δ: ms and $\|\boldsymbol{g}\|$: mT m^{-1}. We also include the cost of solving the Bloch–Torrey PDE directly using finite elements rather than computing eigenfunctions, labelled 'FE', and the value of the reference signal, labelled 'S^{REF}'.

				NMF		New method		FE	
κ	δ	Δ	$\|\boldsymbol{g}\|$	Time	ϵ_{abs}	Time	ϵ_{abs}	Time	S^{REF}
10^{-5}	5	5	50	0.9	0.000004	0.8	0.000001	16.2	0.99
			200	1.2	0.00006	1.1	0.00002	34.0	0.82
			500	1.4	0.0002	1.6	0.00008	68.9	0.43
			1000	1.8	0.0006	2.3	0.0001	117.3	0.22
	10	10	50	1.1	0.000008	1.3	0.000008	22.1	0.91
			200	1.2	0.00007	1.5	0.00009	53.1	0.43
			500	1.8	0.0003	2.3	0.0002	106.9	0.24
			1000	3.5	0.0008	3.8	0.0001	199.0	0.10
5×10^{-5}	5	5	50	0.7	0.000004	0.8	0.00002	16.6	0.99
			200	0.9	0.00005	0.9	0.0003	34.0	0.81
			500	1.4	0.0002	1.4	0.001	67.2	0.38
			1000	2.1	0.0004	1.8	0.001	106.6	0.16
	10	10	50	1.2	0.000007	1.1	0.0002	24.6	0.90
			200	1.4	0.00005	1.1	0.001	59.2	0.35
			500	2.3	0.0001	2.2	0.002	103.7	0.14
			1000	3.7	0.0003	2.6	0.0009	184.8	0.05
10^{-4}	5	5	50	0.9	0.000004	0.8	0.00007	16.0	0.99
			200	1.2	0.00005	1.0	0.0009	37.7	0.80
			500	1.7	0.0002	1.3	0.003	70.0	0.33
			1000	2.7	0.0002	1.7	0.0003	94.6	0.12
	10	10	50	1.4	0.000007	1.0	0.0005	24.9	0.90
			200	1.7	0.00004	1.2	0.004	56.6	0.28
			500	2.6	0.00006	1.7	0.0003	103.9	0.08
			1000	4.7	0.00008	3.1	0.0013	179.9	0.02

are faster and take less memory. In summary, 'eigs' is useful when: (1) only a small number of eigenmodes (for example, less than 15%) are needed, such as for simple geometries, longer diffusion times, lower gradient amplitudes; (2) when the computer RAM is limited. For 'eig', the input and output matrices are full, when $N_{node} = 32\,023$, 'eig' requires 22.9 GB of RAM. In contrast, the inputs of 'eigs' are sparse matrices, when $N_{node} = 32\,023$, 'eigs' requires 1.2 GB of RAM. For other cases, using 'eig' and selecting a subset of eigenmodes is preferred.

Table 2 shows the computational times of the matrix exponential computations in one gradient direction with different settings. In order to accelerate the computation, instead of computing the matrix exponential explicitly, we use the algorithm 'expmv' (Al-Mohy and Higham 2011), which computes the action of matrix exponential on a vector, without explicitly forming the matrix exponential. The number of eigenfunctions is set to $N_{eig} = 2000$. With this choice, the errors of the normalized signals of the original numerical matrix formalism method is less than 0.0008, and the error of the new method is less than 0.0013. We can see from the table that for the original numerical matrix formalism method, the computational time is between 0.7 to 4.7 s. For the new method, the computational time is between 0.8 and 3.8 s. Thus, the two methods are similar in the signal computational step, however, the new method offers substantial computational time advantage over the original numerical matrix formalism method due to the savings in the eigendecomposition step. In the table, we are also include the cost of solving the Bloch–Torrey PDE directly using finite elements rather than computing eigenfunctions, labelled 'FE', clearly, this approach is much more costly than either of the two eigenfunction based methods.

4.4. Numerical study of permeability effects on signal

To illustrate a way that the new method we developed in this paper can be used to study permeability, we test a hypothesis about the behavior of the signal as a function of permeability. Suppose we want to test the hypothesis that the dependence of the signal on the permeability can be approximated by the following expression for a range of values of permeability found in biological tissues:

$$S^{\text{APPROX}}(\boldsymbol{g}, f;\, N_{eig}) = e^{-\beta(\boldsymbol{g},f)\cdot\kappa} \cdot (S_{imp}(\boldsymbol{g}, f;\, N_{eig}) - S_{free}(\boldsymbol{g}, f;\, N_{eig})) + S_{free}(\boldsymbol{g}, f;\, N_{eig}), \tag{51}$$

where $\beta(\boldsymbol{g}, f)$ is a positive fitted coefficient depending on the encoding gradient and the geometry, $S_{imp}(\boldsymbol{g}, f; N_{eig})$ is the signal in the impermeable case and $S_{free}(\boldsymbol{g}, f; N_{eig})$ is the signal in absence of all interior interfaces. Both

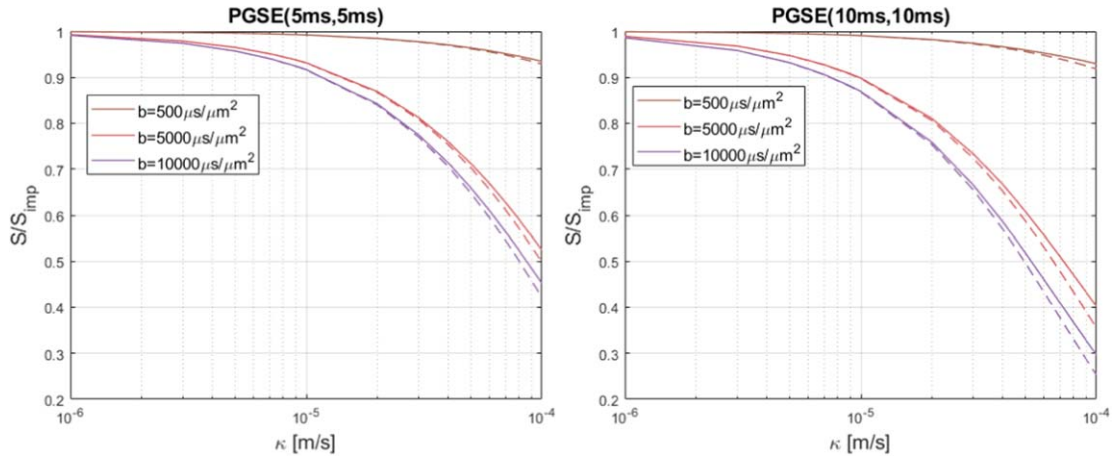

Figure 6. The simulations are performed on Ω^I, with the gradient direction fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$, using the new method with length scale cut-off $L_s = 1\ \mu$m ($N_{eig} = 538$). The solid and dashed lines represent S^{NEW}/S_{imp} and $S^{\text{APPROX}}/S_{imp}$, respectively. Left: short diffusion time case, PGSE (5 ms, 5 ms). The fitted coefficients are $\beta = 2424(\text{m s}^{-1})^{-1}$ ($b = 500$ s mm^{-2}), $\beta = 7497(\text{m s}^{-1})^{-1}$ ($b = 5000$ s mm^{-2}), $\beta = 8984(\text{ m s}^{-1})^{-1}$ ($b = 10000$ s mm^{-2}). The normalized impermeable signals are $S_{imp} = 0.7$ ($b = 500$ s mm^{-2}), $S_{imp} = 0.256$ ($b = 5000$ s mm^{-2}), $S_{imp} = 0.162$ ($b = 1000$ s mm^{-2}); right: long diffusion time case, PGSE(10 ms, 10 ms). The fitted coefficient $\beta = 3185(\text{ m s}^{-1})^{-1}$ ($b = 500$ s mm^{-2}), $\beta = 11536(\text{ m s}^{-1})^{-1}$ ($b = 5000$ s mm^{-2}), $\beta = 14535(\text{ m s}^{-1})^{-1}$ ($b = 10000$ s mm^{-2}). The normalized impermeable signals are $S_{imp} = 0.734$ ($b = 500$ s mm^{-2}), $S_{imp} = 0.341$ ($b = 5000$ s mm^{-2}), $S_{imp} = 0.281$ ($b = 1000$ s mm^{-2}).

$S_{imp}(\boldsymbol{g}, f; N_{eig})$ and $S_{free}(\boldsymbol{g}, f; N_{eig})$ are independent of permeability. By construction, (1) $S = S_{imp}$ when $\kappa = 0$ m s^{-1}; (2) $\lim_{\kappa\to+\infty} S = S_{free}$; (3) the signal is subject to exponential decay in κ;

We computed the permeable signals using the new method and in figure 6 we show the computed S^{NEW}/S_{imp} and $S^{\text{APPROX}}/S_{imp}$ with the fitted values of $\beta(\boldsymbol{g}, f)$. The good fit of the exponential dependence on κ is evident for the range of κ tested. At low gradient strength ($b = 500$ s mm^{-2}), diffusion MRI signal depends very little on κ. As the gradient strength increases, the signal is more sensitive to κ. This result is consistent with the signal behavior at high gradients for one dimensional problems in presence of multiple semi-permeable barriers discussed in Grebenkov (2014).

The evolution of the fitted values of $\beta(\boldsymbol{g}, f)$ as functions of b and $\sqrt{\delta}$ is plotted in figure 7. At low gradient strength, $\beta \propto \sqrt{\delta}$.

4.5. Application to other diffusion MRI sequences

Our methodology can be applied to other sequences, such as double PGSE (Khrapitchev and Callaghan 2001), OGSE (Does *et al* 2003), flow compensation sequence (Haacke and Lenz 1987) and long-narrow pore imaging sequence (Laun *et al* 2011). One should represent or approximate the sequence profile $f(t)$ as a piece-wise constant function defined on n intervals:

$$f(t) = \sum_{i=0}^{n-1} f(t_i)\mathbb{1}_{[t_i, t_{i+1}]}, \tag{52}$$

where $\{t_0, \cdots, t_n\}$ is a strictly increasing sequence between 0 and T_E and $\mathbb{1}_{[t_i, t_{i+1}]}$ is the indicator function on the interval $[t_i,\ t_{i+1}]$.

We illustrate the application of our method to the long-narrow pore imaging sequence (Laun *et al* 2011), which consists of two rectangular pulses of durations $T_E\delta_1$ and $T_E\delta_2$, of gradient strengths $-\|g\|$ and $\|g\|\delta_1/\delta_2$, respectively, separated by a time interval $T_E(1-\delta_1-\delta_2)$, for which the temporal profile $f_{pore}(t)$ is

$$f_{pore}(t) = \begin{cases} -1, & 0 \leqslant t \leqslant T_E\delta_1, \\ \delta_1/\delta_2, & T_E(1-\delta_2) < t \leqslant T_E, \\ 0, & \text{otherwise}, \end{cases} \tag{53}$$

where $\delta_1 > 0$ and $\delta_2 > 0$ are two dimensionless positive time coefficients, with $\delta_1 + \delta_2 \leqslant 1$.

We perform the simulations on Ω^I, with long-narrow pore imaging parameters below:

- $T_E = 20$ ms, $\delta_1 = 1 - \delta_2$ and $\delta_2 = [0.5, 0.2, 0.1, 0.05]$;
- g-value from 0 to 200 mT m^{-1};
- 18 gradient directions uniformly distributed on a unit semicircle.

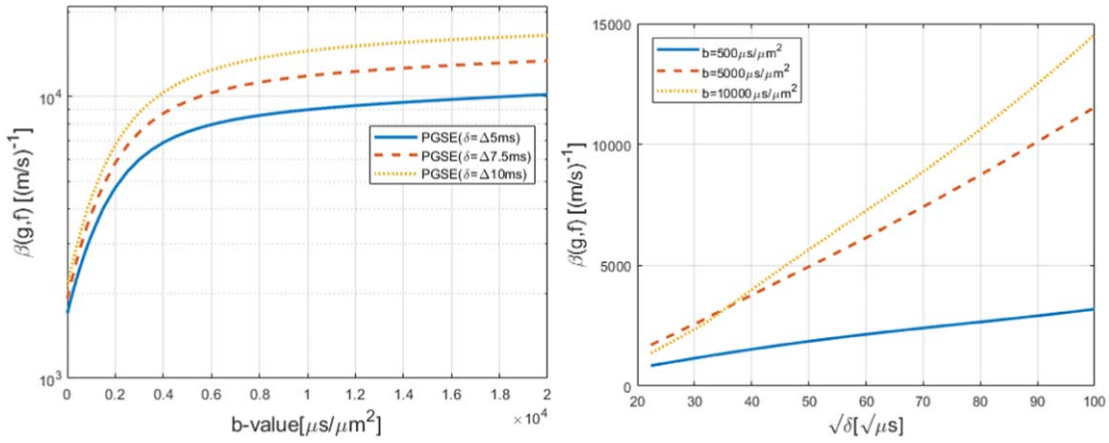

Figure 7. The evolution of $\beta(\boldsymbol{g}, f)$ as a function of b-value (left) for three tested sequences, PGSE (5 ms, 5 ms), PGSE (7.5 ms, 7.5 ms) and PGSE (10 ms, 10 ms), and as a function of $\sqrt{\delta}$ (right) for three tested gradient strengths, $b = 500$ s mm^{-2}, $b = 5000$ s mm^{-2} and $b = 10000$ s mm^{-2}. The simulations are performed on Ω^I, with the gradient direction fixed in $[\sqrt{2}/2,\ \sqrt{2}/2,\ 0]^T$, using the new method with length scale cut-off $L_s = 1$ μm ($N_{eig} = 538$).

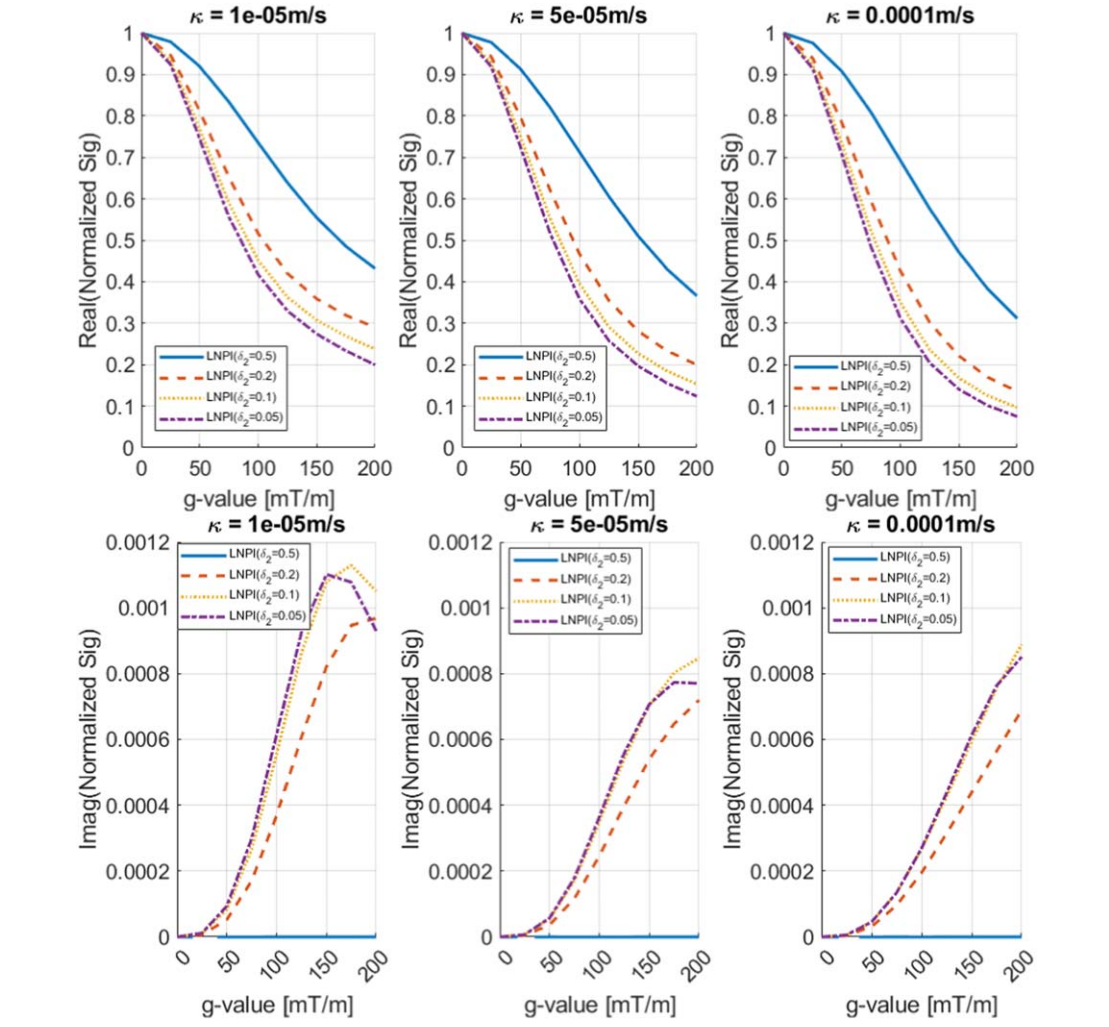

Figure 8. Real part (first row) and imaginary part (second row) of normalized direction-averaged diffusion MRI signals using the long-narrow pore imaging sequence, with ($\delta_1 = 1 - \delta_2$ and $T_E = 20$ ms). Simulations are performed on Ω^I, by SpinDoctor. Left: $\kappa = 10^{-5}$ m s^{-1}; middle: $\kappa = 5 \times 10^{-5}$ m s^{-1}; right: $\kappa = 10^{-4}$ m s^{-1}.

We show in figure 8 the simulated reference signals. We note that when $\delta_2 = 0.5$, we are in the PGSE case. When $\delta_2 \neq 0.5$, we see that the signals have a non-zero imaginary part.

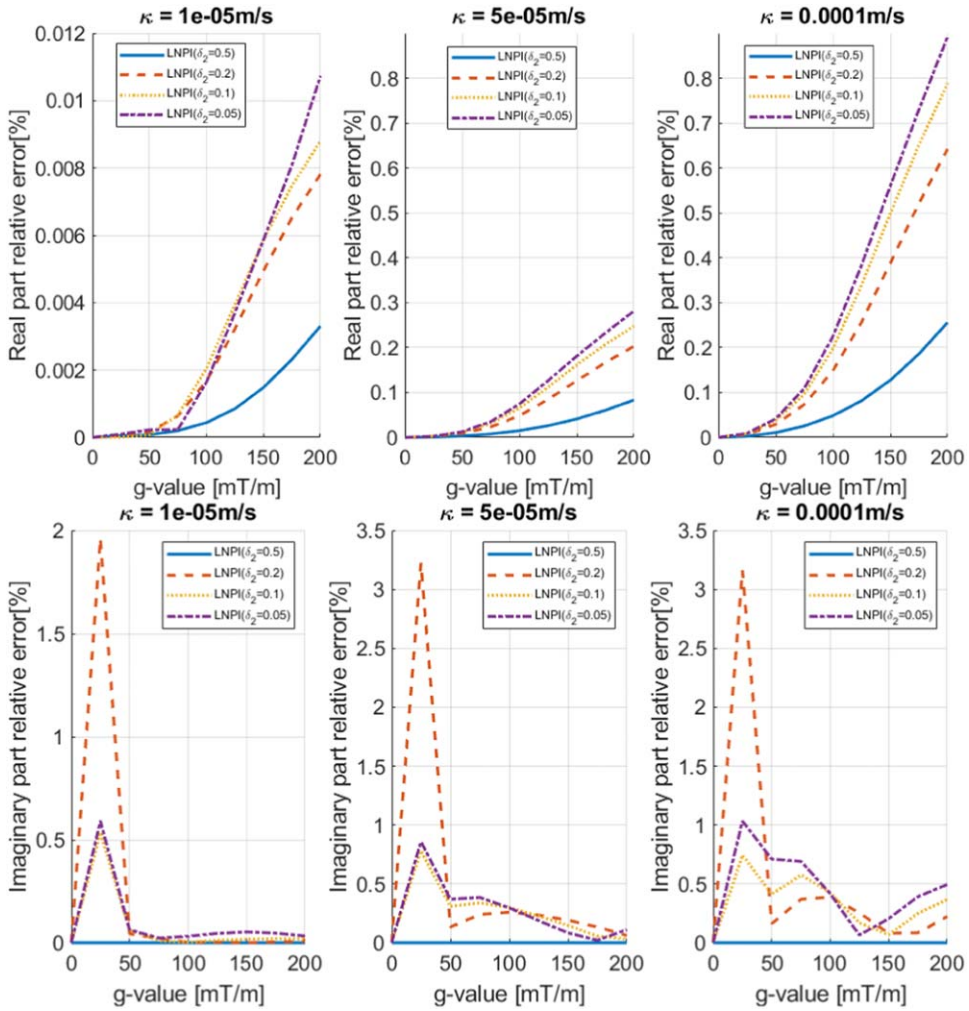

Figure 9. Relative errors of the real part (first row) and the imaginary part (second row) of the direction averaged signals of the new method with the length scale cut-off $L_s = 1$ μm ($N_{eig} = 538$). The relative errors are in percent. Simulations are performed on Ω^I, by SpinDoctor. Left: $\kappa = 10^{-5}$ m s^{-1}; middle: $\kappa = 5 \times 10^{-5}$ m s^{-1}; right: $\kappa = 10^{-4}$ m s^{-1}.

We show in figure 9 the relative errors between the new method with the length scale cut-off $L_s = 1$ μm ($N_{eig} = 538$) and the reference signals. The errors in the real part of the signal are between 0.01% and 1%, the errors in the imaginary part of the signal are between 2% and 3%.

5. Conclusion

The numerical matrix formalism method produces a diffusion MRI signal representation using the Laplace eigenfunctions basis computed on a domain with permeable interfaces. In this paper, we formulated a new representation of the diffusion MRI signal using the Laplace eigenfunctions in the same domain while making the interfaces impermeable. This means our new method can use the same set of eigenfunctions for many different values of permeability, thus saving computational time in the eigendecomposition step. While the new method requires more eigenfunctions than the original numerical matrix formalism method to achieve the same accuracy, we have shown that if the permeability is not too high (while still staying in the realistic range for biological cell membranes), the total computational time is still significantly lower than the original numerical matrix formalism method.

Acknowledgments

Zheyi Yang is financially supported by a PhD scholarship from the China Scholarship Council (CSC).

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary information files). Data will be available from 1 January 2024.

Appendix. Proof of theorem 1

Proof. For a permeability matrix $\boldsymbol{Q} \in \mathbb{R}^{N_{node},N_{node}}$, let $\boldsymbol{L}_{per} \in \mathbb{R}^{N_{eig},N_{eig}}$ and $\boldsymbol{P}_{per} \in \mathbb{R}^{N_{node},N_{eig}}$ be the eigenvalues matrix and the eigenfunctions matrix, respectively,

$$(\boldsymbol{K} + \boldsymbol{Q})\boldsymbol{P}_{per} = \boldsymbol{M}\boldsymbol{P}_{per}\boldsymbol{L}_{per}. \tag{A1}$$

The subscript *per* indicates these matrices are from permeable case. Then the H matrix for the permeable case is

$$\boldsymbol{H}_{per}(\boldsymbol{g}, f) \equiv e^{-\delta(\boldsymbol{L}_{per} - I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))}e^{-(\Delta-\delta)\boldsymbol{L}_{per}}e^{-\delta(\boldsymbol{L}_{per} + I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))}, \tag{A2}$$

where

$$\boldsymbol{W}_{per}(\boldsymbol{g}) = \boldsymbol{P}_{per}^{T}\boldsymbol{J}(\boldsymbol{g})\boldsymbol{P}_{per},$$

and the signal is

$$S^{\mathrm{NMF}}(\boldsymbol{g}, f; N_{eig}) = (\mathbf{1}_{N_{node},1}^{T}\boldsymbol{M}\boldsymbol{P}_{per})\boldsymbol{H}_{per}(\boldsymbol{g}, f)(\boldsymbol{P}_{per}^{T}\boldsymbol{M}\mathbf{1}_{N_{node},1})\rho. \tag{A3}$$

We recall that all the eigenfunctions are L^2-normalized and orthogonal, so that

$$\boldsymbol{P}_{imp}^{T}\boldsymbol{M}\boldsymbol{P}_{imp} = I_{N_{eig}}, \tag{A4}$$

$$\boldsymbol{P}_{per}^{T}\boldsymbol{M}\boldsymbol{P}_{per} = I_{N_{eig}}, \tag{A5}$$

where $I_{N_{eig}}$ is identity matrix, thus, multiplying $\boldsymbol{P}_{imp}^{T}$ on the both sides of equation (40) and $\boldsymbol{P}_{per}^{T}$ on the both sides of equation (A1) gives

$$\boldsymbol{L}_{imp} = \boldsymbol{P}_{imp}^{T}\boldsymbol{K}\boldsymbol{P}_{imp}, \tag{A6}$$

$$\boldsymbol{L}_{per} = \boldsymbol{P}_{per}^{T}(\boldsymbol{K} + \boldsymbol{Q}_{per})\boldsymbol{P}_{per}. \tag{A7}$$

We define a new matrix $\boldsymbol{C} \in \mathbb{R}^{N_{eig},N_{eig}}$, projecting the permeable Laplace eigenfunctions onto the impermeable Laplace eigenfunctions:

$$\boldsymbol{C} \equiv \boldsymbol{P}_{imp}^{T}\boldsymbol{M}\boldsymbol{P}_{per}. \tag{A8}$$

Knowing that the mass matrix $\boldsymbol{M}$ is real, symmetric and positive-definite, we apply the Cholesky factorization, $\boldsymbol{M} = \boldsymbol{R}^{T}\boldsymbol{R}$. Under the condition that $N_{eig} = N_{node}$ so that $\boldsymbol{P}_{imp}$ and $\boldsymbol{P}_{per}$ are full rank square matrices, we have

$$\begin{aligned}
&\boldsymbol{P}_{imp}^{T}(\boldsymbol{R}^{T}\boldsymbol{R})\boldsymbol{P}_{imp} = I_{N_{eig}} \leftrightarrow (\boldsymbol{R}\boldsymbol{P}_{imp})(\boldsymbol{P}_{imp}^{T}\boldsymbol{R}^{T}) = I_{N_{node}},\\
&\boldsymbol{P}_{per}^{T}(\boldsymbol{R}^{T}\boldsymbol{R})\boldsymbol{P}_{per} = I_{N_{eig}} \leftrightarrow (\boldsymbol{R}\boldsymbol{P}_{per})(\boldsymbol{P}_{per}^{T}\boldsymbol{R}^{T}) = I_{N_{node}}.
\end{aligned} \tag{A9}$$

Then, we can derive that $\boldsymbol{C}$ is a unitary matrix:

$$\begin{aligned}
\boldsymbol{C}^{T}\boldsymbol{C} &= \boldsymbol{P}_{per}^{T}\boldsymbol{M}\boldsymbol{P}_{imp}\boldsymbol{P}_{imp}^{T}\boldsymbol{M}\boldsymbol{P}_{per}\\
&= \boldsymbol{P}_{per}^{T}(\boldsymbol{R}^{T}\boldsymbol{R})\boldsymbol{P}_{imp}\boldsymbol{P}_{imp}^{T}(\boldsymbol{R}^{T}\boldsymbol{R})\boldsymbol{P}_{per}\\
&= \boldsymbol{P}_{per}^{T}\boldsymbol{R}^{T}(\boldsymbol{R}\boldsymbol{P}_{imp}\boldsymbol{P}_{imp}^{T}\boldsymbol{R}^{T})\boldsymbol{R}\boldsymbol{P}_{per}\\
&= I_{N_{eig}}.
\end{aligned} \tag{A10}$$

In addition, we can derive that

$$\boldsymbol{P}_{per} = \boldsymbol{P}_{imp}\boldsymbol{C}, \tag{A11}$$

$$\boldsymbol{P}_{imp} = \boldsymbol{P}_{per}\boldsymbol{C}^{T}, \tag{A12}$$

because

$$\begin{aligned}
&\boldsymbol{P}_{imp}\boldsymbol{C} = \boldsymbol{P}_{imp}\boldsymbol{P}_{imp}^{T}\boldsymbol{M}\boldsymbol{P}_{per}\\
\leftrightarrow\ &\boldsymbol{R}\boldsymbol{P}_{imp}\boldsymbol{C} = \boldsymbol{R}\boldsymbol{P}_{imp}\boldsymbol{P}_{imp}^{T}\boldsymbol{R}^{T}\boldsymbol{R}\boldsymbol{P}_{per}\\
\leftrightarrow\ &\boldsymbol{R}\boldsymbol{P}_{imp}\boldsymbol{C} = \boldsymbol{R}\boldsymbol{P}_{per}\\
\leftrightarrow\ &\boldsymbol{P}_{imp}\boldsymbol{C} = \boldsymbol{P}_{per}
\end{aligned}$$

since $\boldsymbol{R}$ is invertible.

Combining equations (A6), (A7), (A11), (A12) we have

$$\boldsymbol{L}_{proj} \equiv \boldsymbol{L}_{imp} + \boldsymbol{Q}_{proj} = \boldsymbol{P}_{imp}^{T}(K + Q_{per})\boldsymbol{P}_{imp} = \boldsymbol{C}\boldsymbol{P}_{per}^{T}(K + Q_{per})\boldsymbol{P}_{per}\boldsymbol{C}^{T} = \boldsymbol{C}\boldsymbol{L}_{per}\boldsymbol{C}^{T}, \tag{A13}$$

and similarly,

$$\boldsymbol{W}_{imp} \equiv \boldsymbol{P}_{imp}^{T}\boldsymbol{J}(\boldsymbol{g})\boldsymbol{P}_{imp} = \boldsymbol{C}\boldsymbol{P}_{per}^{T}\boldsymbol{J}(\boldsymbol{g})\boldsymbol{P}_{per}\boldsymbol{C}^{T} = \boldsymbol{C}\boldsymbol{W}_{per}\boldsymbol{C}^{T}. \tag{A14}$$

Then the matrix exponentials satisfy

$$e^{-(\Delta-\delta)\boldsymbol{L}_{proj}} = e^{-(\Delta-\delta)\boldsymbol{C}\boldsymbol{L}_{per}\boldsymbol{C}^{T}} = \boldsymbol{C}e^{-(\Delta-\delta)\boldsymbol{L}_{per}}\boldsymbol{C}^{T},$$
$$e^{-\delta(\boldsymbol{L}_{proj}+I\gamma\boldsymbol{W}_{imp}(\boldsymbol{g}))} = e^{-\delta\boldsymbol{C}(\boldsymbol{L}_{per}+I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))\boldsymbol{C}^{T}} = \boldsymbol{C}e^{-\delta(\boldsymbol{L}_{per}+I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))}\boldsymbol{C}^{T},$$

because $\boldsymbol{C}\boldsymbol{C}^{T} = I$. Thus,

$$\begin{aligned}\boldsymbol{H}_{proj}(\boldsymbol{g}, f) &= e^{-\delta(\boldsymbol{L}_{proj}-I\gamma\boldsymbol{W}_{imp}(\boldsymbol{g}))} \cdot e^{-(\Delta-\delta)\boldsymbol{L}_{proj}} \cdot e^{-\delta(\boldsymbol{L}_{proj}+I\gamma\boldsymbol{W}_{imp}(\boldsymbol{g}))} \\ &= \boldsymbol{C}e^{-\delta(\boldsymbol{L}_{per}-I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))}\boldsymbol{C}^{T}\boldsymbol{C}e^{-(\Delta-\delta)\boldsymbol{L}_{per}}\boldsymbol{C}^{T}\boldsymbol{C}e^{-\delta(\boldsymbol{L}_{per}+I\gamma\boldsymbol{W}_{per}(\boldsymbol{g}))}\boldsymbol{C}^{T} \\ &= \boldsymbol{C}\boldsymbol{H}_{per}(\boldsymbol{g}, f)\boldsymbol{C}^{T}.\end{aligned} \tag{A15}$$

Substituting equation (A11) A15 into 45, we obtain the equivalence:

$$\begin{aligned}S^{\mathrm{NEW}}(\boldsymbol{g}, f; N_{node}) &= \rho(\mathbf{1}_{N_{node},1}^{T}\boldsymbol{M}\boldsymbol{P}_{imp})\boldsymbol{H}_{proj}(\boldsymbol{g}, f)(\boldsymbol{P}_{imp}^{T}\boldsymbol{M}\mathbf{1}_{N_{node},1}) \\ &= \rho\mathbf{1}_{N_{node},1}^{T} \cdot \boldsymbol{M} \cdot (\boldsymbol{P}_{imp} \cdot \boldsymbol{C}) \cdot \boldsymbol{H}_{per}(\boldsymbol{g}, f) \cdot (\boldsymbol{C}^{T} \cdot \boldsymbol{P}_{imp}^{T}) \cdot \boldsymbol{M} \cdot \mathbf{1}_{N_{node},1} \\ &= \rho(\mathbf{1}_{N_{node},1}^{T}\boldsymbol{M}\boldsymbol{P}_{per})\boldsymbol{H}_{per}(\boldsymbol{g}, f)(\boldsymbol{P}_{per}^{T}\boldsymbol{M}\mathbf{1}_{N_{node},1}) \\ &= S^{\mathrm{NMF}}(\boldsymbol{g}, f; N_{node}).\end{aligned} \tag{A16}$$

Thus, when the full set of the discretized eigenfunctions is used, the new method using the impermeable eigenfunctions gives the same signal as the original numerical matrix formalism method that uses permeable eigenfunctions.

ORCID iDs

Zheyi Yang https://orcid.org/0000-0001-9750-0172
Chengran Fang https://orcid.org/0000-0003-4933-7328
Jing-Rebecca Li https://orcid.org/0000-0001-6075-5526

References

Agdestein S D, Tran T N and Li J R 2021 *NMR Biomed.* **35** e4646
Alemany I, Rose J N, Garnier-Brun J, Scott A D and Doorly D J 2022 *Sci. Rep.* **12** 10759
Al-Mohy A H and Higham N J 2011 *SIAM J. Sci. Comput.* **33** 488–511
Bai R, Li Z, Sun C, Hsu Y C, Liang H and Basser P 2020 *NeuroImage* **219** 117039
Barzykin A 1999 *J. Magn. Reson.* **139** 342–53
Beltrachini L, Taylor Z A and Frangi A F 2015 *J. Magn. Reson.* **259** 126–34
Callaghan P T 1997 *J. Magn. Reson.* **129** 74–84
Callaghan P T and Stepišnik J 1995 *J. Magn. Reson.* A **117** 118–22
Chin C L, Wehrli F W, Hwang S N, Takahashi M and Hackney D B 2002 *Magn. Reson. Med.* **47** 455–60
Coppersmith D and Winograd S 1990 *J. Symb. Comput.* **9** 251–80
Does M D, Parsons E C and Gore J C 2003 *Magn. Reson. Med.: Official J. Int. Soc. Magn. Resona. Med.* **49** 206–15
Drobnjak I, Zhang H, Hall M G and Alexander D C 2011 *J. Magn. Reson.* **210** 151–7
Fieremans E, Novikov D S, Jensen J H and Helpern J A 2010 *NMR Biomed.* **23** 711–24
Grebenkov D 2014 Efficient monte carlo methods for simulating diffusion-reaction processes in complex systems *First-Passage Phenomena and Their Applications* (World Scientific) pp 571–95
Grebenkov D S 2007 *Rev. Mod. Phys.* **79** 1077–137
Grebenkov D S 2008 *Concepts Magn. Reson.* A **32A** 277–301
Grebenkov D S 2011 *J. Magn. Reson.* **208** 243–55
Grebenkov D S 2014 *J. Magn. Reson.* **248** 164–76
Grebenkov D S, Van Nguyen D and Li J R 2014 *J. Magn. Reson.* **248** 153–63
Haacke E M and Lenz G W 1987 *Am. J. Roentgenol.* **148** 1251–8
Hall M and Alexander D 2009 *IEEE Trans. Med. Imaging* **28** 1354–64
Huang S Y *et al* 2021 *NeuroImage* **243** 118530
Jelescu I O, de Skowronski A, Geffroy F, Palombo M and Novikov D S 2022 *NeuroImage* **256** 119277
Khrapitchev A A and Callaghan P T 2001 *J. Magn. Reson.* **152** 259–68
Laun F B, Kuder T A, Semmler W and Stieltjes B 2011 *Phys. Rev. Lett.* **107** 048102
Le Bihan D, Breton E, Lallemand D, Grenier P, Cabanis E and Laval-Jeantet M 1986 *Radiology* **161** 401–7
Lee H H, Fieremans E and Novikov D S 2021 *J. Neurosci. Methods* **350** 109018
Lee H H, Papaioannou A, Novikov D S and Fieremans E 2020 *NeuroImage* **222** 117054
Lee J, Balakrishnan V, Koh C K and Jiao D 2009 From o(k2n) to o(n): a fast complex-valued eigenvalue solver for large-scale on-chip interconnect analysis *2009 IEEE MTT-S Int. Microwave Symp. Digest* pp 181–4

Li J R, Nguyen V D, Tran T N, Valdman J, Trang C B, Nguyen K V, Vu D T S, Tran H A, Tran H T A and Nguyen T M P 2019 *NeuroImage* **202** 116120
Li J R, Tran T N and Nguyen V D 2020 *NMR Biomed.* 33 (arXiv:1911.07165)
Moutal N and Grebenkov D 2019 *J. Sci. Comput.* **81** 1630–54
Nedjati-Gilani G L, Schneider T, Hall M G, Cawley N, Hill I, Ciccarelli O, Drobnjak I, Wheeler-Kingshott C A G and Alexander D C 2017 *NeuroImage* **150** 119–35
Nguyen D V, Li J R, Grebenkov D and Le Bihan D 2014 *J. Comput. Phys.* **263** 283–302
Nguyen H T, Grebenkov D, Van Nguyen D, Poupon C, Le Bihan D and Li J R 2015 *Phys. Med. Biol.* **60** 3389–413
Nguyen V D, Leoni M, Dancheva T, Jansson J, Hoffman J, Wassermann D and Li J R 2019 *J. Magn. Reson.* **309** 106611
Olesen J L, Østergaard L, Shemesh N and Jespersen S N 2022 *NeuroImage* **251** 118976
Palombo M, Ianus A, Guerreri M, Nunes D, Alexander D C, Shemesh N and Zhang H 2020 *NeuroImage* **215** 116835
Romascano D, Barakovic M, Rafael-Patino J, Dyrby T B, Thiran J P and Daducci A 2020 *Magn. Reson. Med.* **83** 2322–30
Russell G, Harkins K D, Secomb T W, Galons J P and Trouard T P 2012 *Phys. Med. Biol.* **57** N35–46
Si H 2015 *ACM Trans. Math. Softw.* **41** 11:1–11:36
Stejskal E O and Tanner J E 1965 *J. Chem. Phys.* **42** 288–92
Torrey H C 1956 *Phys. Rev.* **104** 563–5
Waudby C A and Christodoulou J 2011 *J. Magn. Reson.* **211** 67–73
Yeh C H, Schmitt B, Le Bihan D, Li-Schlittgen J R, Lin C P and Poupon C 2013 *PLoS One* **8** e76626