Master Thesis

Fitting and sampling of multi-compartment models to Diffusion Weighted MRI data

R.L. Harms

Master Thesis DKE 12-07

Thesis submitted in partial fulfillment
of the requirements for the degree of Master of Science
of Operations Research at the Department of Knowledge
Engineering of the Maastricht University

Thesis Committee:

Dr. F. Thuijsman
Dr. A. Roebroeck

Maastricht University
Faculty of Humanities and Sciences
Department of Knowledge Engineering
Master Operations Research

July 5, 2012

Preface

This thesis is the final requirement for the Master of Science degree at the Department of Knowledge Engineering. One of my interests has always been the human body, at one point I actually considered studying Medicine instead of Knowledge Engineering. When I was about to choose a subject for my thesis, I read Dick Swaab's ”We are our brains”. This book inspired me to focus on the human brain as my subject of study. Through Frank Thuijsman a meeting was arranged with Alard Roebroeck of Cognitive Neuroscience and soon a thesis subject was chosen. What I like about this thesis is that it combines my study of Mathematics and Computer Science with my interest in the human brain.

The thesis is written at the Department of Knowledge Engineering, Maastricht University in collaboration with the Department of Cognitive Neuroscience, Maastricht University. I would like to thank my supervisors, Dr. Frank Thuijsman, Dr. Alard Roebroeck and Matteo Bastiani, MSc. Alard's and Matteo's enthusiasm and knowledge on the subject helped me to advance quickly in the field of Neuroscience. I would like to give special thanks to Frank Thuijsman for his aid and support, not only during this thesis but during most of my years at DKE.

Summary

Diffusion Weighted Magnetic Resonance Imaging (DW-MRI) can be used to probe the microstructure of biological tissue. In DW-MRI water molecules are sensitized to the microscopic thermal motion (diffusion) of water molecules. In the brain, diffusion is hindred by various tissue processes such as intracellular and extracellular volume and the direction of oriented structures. As such it can be used to characterize the white matter in the human brain. This can be used for the diagnosis of stroke, tumor and degenerative white matter diseases to delineating axonal fiber bundles that connect remote regions in the brain.

The most frequently approach to tractography – deriving the complete trajectories of fibre-bundles through the brain – is two-staged. In the first, local models are fitted to each voxel in the dataset to obtain the local fiber-bundle direction. In the second stage, a tractography algorithm connects these local directions step-by-step through the whole 3D data set to form estimated fiber bundles. In global tractography both the parameters of local models and the trajectories of the fiber-bundles are estimated at the same time.

In both local and global tractography it is necessary to model DW-MRI data per voxel. This thesis presents a toolkit aimed at the fitting and sampling of multi-compartment models to DW-MRI data. Using this toolbox it is possible to easily fit a multitude of local models to each voxel of a 3D DW-MRI dataset. Next to fitting models the toolbox also contains sampling methods with Automatic Relevance Detection (ARD) which allow sampling of the posterior distribution of the model parameters. This last feature can be used as a fitting tool (by taking the mean of the sampled parameter distributions) but it is also the first step in the direction of global tractography.

Contents

Chapter 1

Introduction

Diffusion-weighted Magnetic Resonance Imaging (DW-MRI) allows us to depict the organization of the white matter of a brain *in vivo*, using the random, thermally driven motion of water molecules as a contrast mechanism [11]. In the brain the movement of water molecules is restricted and hindered by various tissue processes and can therefore be used to reconstruct neuronal pathways in white matter. This process of reconstruction is called tractography, which can be divided into two groups: local and global tractography [18]. In local tractography, fibres are constructed one by one and do not influence each other. Long pathways are constructed step-by-step using a local voxel-wise definition of the brain. This can be done either deterministically, or probabilistically using an Orientation Distribution Function (ODF). In global tractography the whole projectome (the complete set of white matter fascicles, see [16]) of the brain is computed at once by finding the configuration of fibres which best describes the data measured. This approach implicitly requires calculating the fit between the proposed fibres and the measured DW-MRI signal.

Both tractography methods require the prediction of the DW-MRI signal measured using a given neuronal direction. This can be done by using either non-parametric or parametric models. The focus of this thesis is on parametric models and the main product of this thesis is a toolkit which can be used to pre-process DW-MRI data, fit parametric models and, using a sampler, to approximate the posterior distribution of the model parameters. This posterior distribution can then be used to create an ODF. The additional advantage of this toolkit over other toolkits such as Camino [7] is that this toolkit is written in Matlab (used by many researchers), is easily extendable to new parametric models and includes sampling methods with Automatic Relevance Detection (ARD).

In order to be able to interpret and appreciate the toolkit and the experimental results some background information is given about the brain, MRI and tractography. Section 1.1 introduces the brain and white matter fibres, section 1.2 introduces MRI and DW-MRI and section 1.3 introduces tractography. Section 1.4 lists the different research questions and finally section 1.5 provides an outline of the remainder of this thesis.

1.1 The human brain

The human brain is a highly complex organ containing between 80 to 120 billion neurons [12]. The outermost layer of the brain is called the cortex; here, all information processing takes place. One layer deeper, contained by the cortex, the white matter of the brain is situated. White matter forms the bridge between various parts of the cortex, acting as a relay and coordinating communication between different brain regions [10]. Understanding the structure of white matter may help to understand the functioning of white matter. As S. Jbabdi puts it [13]:

> For example, a hierarchical network architecture allows for the possibility of bottom-up and top-down message passing, and ladder-like hierarchies can allow shared computations between parallel hierarchies [...] Understanding these topographies can potentially give us indicators, or even predictors, for the activity of target cortical regions.

There are two ways by which these white matter bundles can be located. One way is *ex vivo* by cutting and slicing the brain to locate the bundles. This however is inappropriate in a clinical setting. To visualize white matter bundles *in vivo* and non-invasively there is currently only one approach which is by means of diffusion weighted MRI and tractography.

1.2 Diffusion weighted MRI

Diffusion weighted Magnetic Resonance Imaging (DW-MRI) is an MRI technique in which the apparent diffusion of water molecules in white matter is measured along a number of different gradient directions. Molecular diffusion, or Brownian motion as first formally described by A. Einstein in 1905 [9], refers to the notion that any kind of molecule is randomly displaced when agitated by thermal energy (Fig. 1.1).This movement is completely random and is only hindered by the boundaries of its container. In statistical terms this displacement can be captured in a displacement distribution which can give the proportion of molecules undergoing displacement in a given direction over a given distance. For example, suppose we have a one-dimensional system of water molecules in which, for each molecule, we track the one-dimensional displacement over a specific time interval. In general this distribution will follow a Gaussian distribution in which most of the molecules travel only a short distance and only a few travel a long distance (Fig. 1.2).

Since the brain is a three-dimensional object, in MRI imaging of the brain the water molecules are displaced in three directions. To get a high resolution MR image of the brain, the brain is partitioned into small three dimensional cubes called voxels (currently in the order of mm^3). For each of these voxels we then measure the diffusion of water molecules. To visualise the displacement distribution in such a voxel we can use colors to indicate a low probability (red, for example) and a high probability (blue) of displacement. See for example fig. 1.3a for an example of isotropic diffusion. Neuronal tissue consists of tightly packed and coherently aligned axons (nerve fibres) surrounded by glial cells

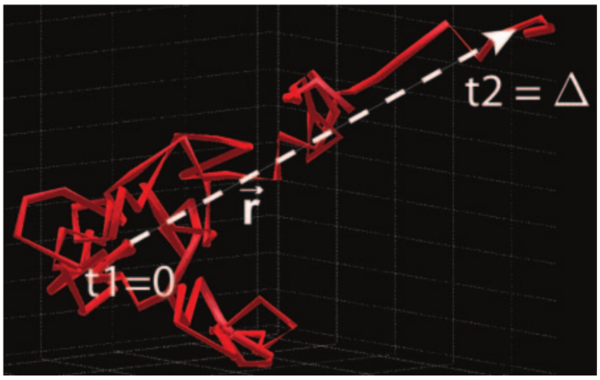


Figure 1.1: Figure showing the random movement of a single molecule during diffusion in the time interval t1=0 to t2=Δ. The white line (vector **r**) shows the net displacement of the molecule. Image taken from [11].

(provide support and protection for neurons in the brain [14]) and are often organized in bundles [11]. ”Often”, since in some areas of the brain, for example near the cortex, the axons may not be bundled. The diffusion of water molecules is mostly hindered by these axon bundles which results in a lower diffusion perpendicular to the axons and a higher diffusion parallel to the axons. In the case of a single fibre bundle of aligned axons the resulting displacement distribution is cigar shaped (Fig. 1.3b). In the case of, for example, a crossing of two fibres we get a cross shaped displacement distribution (Fig. 1.3c). Other distributions are of course equally possible.

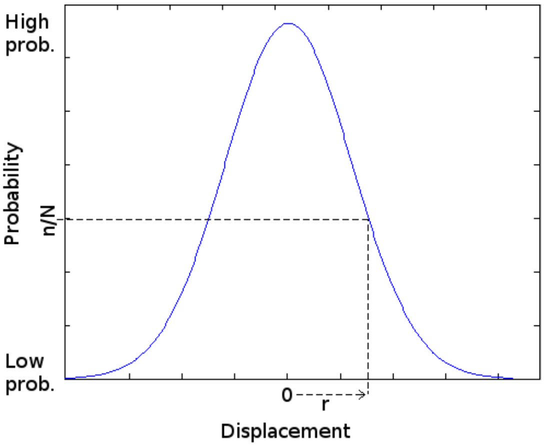


Figure 1.2: In a one-dimensional setting we can plot the diffusion distribution in a two-dimensional graph with the displacement on the x-axis and the proportion of molecules that are displaced over the given distance on the y-axis. For each displacement distance **r** the corresponding probability is the proportion of molecules n/N displaced over that distance.

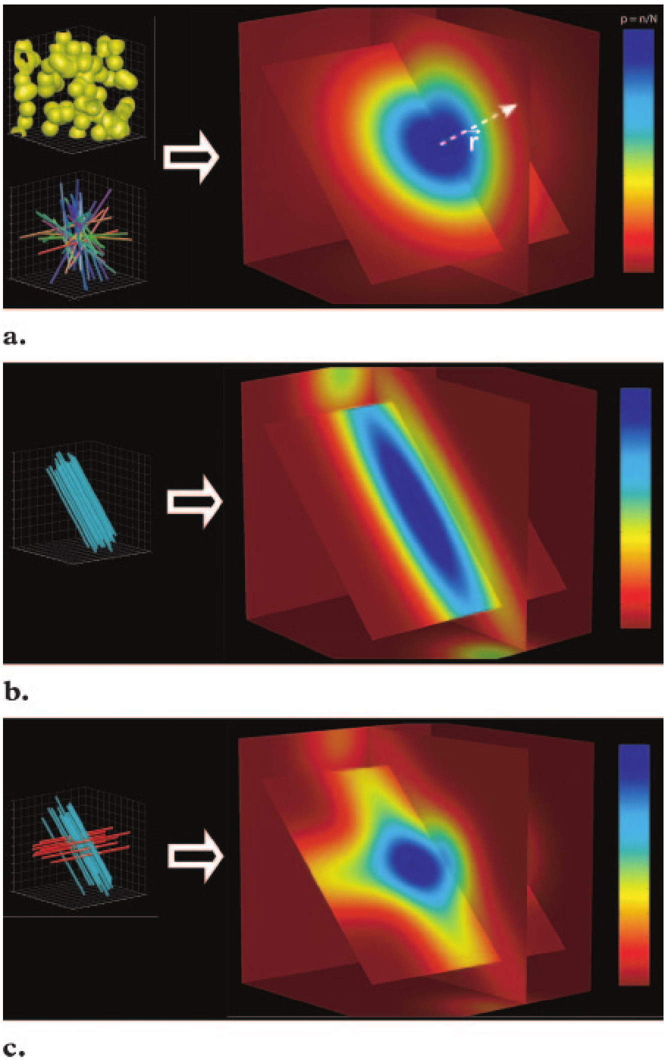


Figure 1.3: Diffusion within a single voxel. On the left of each case a,b and c the structure of the voxel is shown, and in the center the resulting 3D displacement distribution. The bar on the right shows the color coding representing the probabilities of displacement, from low (red) to high (blue). (a) Diagram shows the displacement distribution for a voxel which contains spherical cells (top) or randomly oriented cylindrical structures which intersect, such as axons (bottom). This results in a symmetric image since there is no preferential direction of diffusion. (b) Diagram showing the distribution function following from a voxel in which all the axons are aligned in a uniform direction. This results in a cigar shaped distribution. (c) Diagram showing the resulting displacement distribution for a voxel in which there are two aligned sets of axons perpendicular to each other. The resulting distribution is cross shaped. Figure taken from [11].

1.2.1 Diffusion weighting

In normal MR imaging the images are gray scale images depicting the signal intensity for each voxel, which results in a 3D image of the brain. In mathematical terms this image is the graph of a three-dimensional function $f(\mathbf{p}) : \mathbb{N}^3 \mapsto \mathbb{R}$, where for each voxel $\mathbf{p}$ in the brain $f(\mathbf{p})$ gives the signal intensity in a black and white range. Since this function only gives the nett molecule displacement for each voxel this 3D function is not helpful in locating white matter bundles. To characterize the molecular displacement for each voxel better we would have to approximate the displacement distribution for each voxel. This displacement distribution is again a 3D function. If we combined these two three-dimensional functions we would obtain a six-dimensional (6D) function $f(\mathbf{p}, \mathbf{r})$, which gives for each voxel $\mathbf{p}$ the proportion of water molecules that have been displaced over direction $\mathbf{r}$.

To construct this function $f(\mathbf{p}, \mathbf{r})$ for each location $\mathbf{p}$ the displacement in direction $\mathbf{r}$ must be measured. How this is done exactly is explained by P. Hagmann et al in [11]. For this thesis a less detailed explanation will suffice. To capture the displacement in direction $\mathbf{r}$ in each voxel, during the MRI sequencing the water molecules in the brain are 'activated' in the direction of a gradient $\mathbf{q}$. If, after activating the water molecules an MR image is taken we will observe a low intensity signal in regions where diffusion along the applied gradient direction $\mathbf{q}$ is high and *vice versa*. By sampling over this $\mathbf{q}$-space we can calculate for each voxel the displacement distribution and hence construct the function $f(\mathbf{p}, \mathbf{r})$. The gradient strength (proportional to the length of the vector $\mathbf{q}$), often named diffusion weighting, is sometimes expressed in terms of a b value. This b value is proportional to the product of the square of the gradient strength $\mathbf{q}$ and the diffusion time interval Δ, so: $b \sim q^2 \cdot \Delta$ [11].

1.2.2 Data representation

In general, the acquisition settings and the measured signal are stored independently of each other in two different datasets. The acquisition settings are often stored in a so called *scheme* file. The basic scheme file is a two-dimensional dataset with, in its first three columns columns, gradient direction $g = \{g_x, g_y, g_z\}$ and in the last column the b value (the diffusion weighting) (Table 1.1) where $\mathbf{q} = b \cdot g$ for each line in the scheme file. The measured DW-MRI signal is stored in a four-dimensional matrix in which dimension one to three are the voxels and dimension four represents for each of the rows in the scheme file the measured signal intensity. Together these two matrices hold the same information as represented by the function $f(\mathbf{p}, \mathbf{r})$ mentioned before.

	g_x	g_y	g_z	b
1	...	...	...	...
2	...	...	...	...
3	...	...	...	...
...	...	...	...	...

Table 1.1: Basic scheme file listing the acquisition gradients

1.3 Tractography

Tractography is about deriving the complete trajectories of fibre-bundles through the brain. In general there are two approaches, local and global tractography. Local tractography is two-staged. In the first, voxel-wise estimates of local models are obtained which define the local fibre bundle direction. In the second, a tractography algorithm follows local directions through the whole 3D data set to form estimated fibre bundles. The global approach to tractography estimates both the parameters of local models and trajectories of fibre-bundles at the same time. In both approaches a good approximation of the local (voxel-wise) models is necessary to locate the trajectory of the fibres. Approximating the fibre orientation from DW-MRI data is a difficult problem since different fibre orientations can give the same displacement distribution. For example a kissing (touching of two bended fibre bundles) and a crossing (two fibre bundles perpendicular to each other) can give the same displacement distribution (Fig. 1.4). Predicting the correct orientation is important since it directly affects the quality of the tractography. If a crossing is supposed where a kissing is present the calculated fibre model can be erroneous.

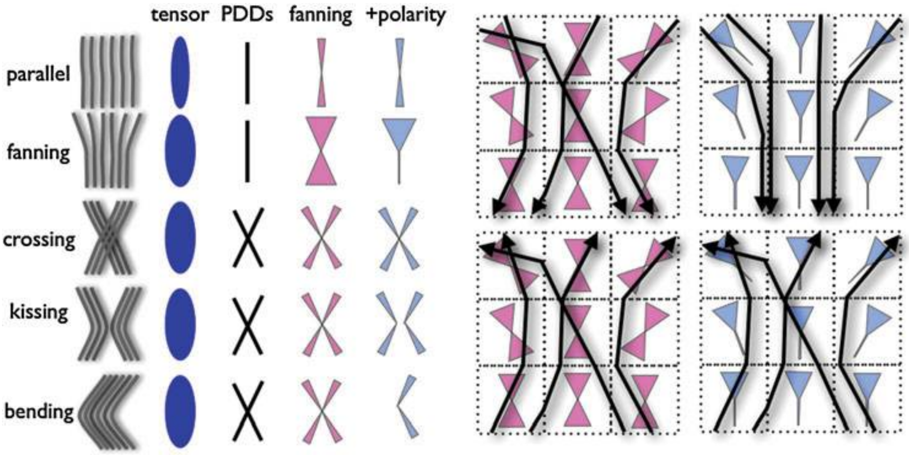


Figure 1.4: The table on the left shows various fibre orientations (left column) and their representation by a few different models. The tensor model is unable to differentiate between different orientations. Adding the Principal Diffusion Direction (PDD) already gives more information about the fibre orientation. Further modelling of the fanning and polarity can help to distinguish all the different cases. The image on the right shows the consequences of the ambiguity of the axonial orientation. The same (local) structure can lead to different fibre orientations. Image taken from [13].

1.3.1 Local models

The first stage of local tractography is to get a voxel-wise estimate of local models which define the local fibre bundle direction. In general there are two different methods which can be used to model these fibre directions, namely non-parametric and parametric models. Non-parametric models are for example multi-dimensional polynomial functions. These models can be fitted to the data to get a representation of the underlying fiber orientation, but the optimized parameters have no biological representation and lack specificity as they are affected by many features of the micro structure [17]. To overcome these limitations, the recent trend in the diffusion MRI field is towards more descriptive models of tissue; the parametric models. In the parametric models the parameters of the model represent specific properties of the tissue, such as cell size and cell density. The focus of this thesis is on the parametric models.

1.4 Research questions

The majority of approaches to tractography are local tractography approaches in which the fibres are tracked one-by-one by first fitting a local model and then linking the local fibre parts. These methods are rather fast, but are problematic in an important way. Small imperfections in the determination of local steps can accumulate and significantly affect the tracked fibre. Global tractography tries to overcome these problems by trying to reconstruct the fibres directly from the DW-MRI data. The goal of this thesis is two-fold, the first is to build a DW-MRI toolkit with parametric models and the second is a 'global estimation algorithm' which estimates both the parameters of local models and trajectories of fiber-bundles at the same time. The latter is a high-dimensional optimization problem which will require an efficient Markov Chain Monte Carlo (MCMC) algorithm. The accompanying research questions are:

- What are the likelihood models which tie parameter values to observed data, which can be estimated from current-day DW-MRI data?

- Which heuristics in the form of proposal distributions tune the performance of the global MCMC algorithm?

- Does global MCMC estimation make local model parameter estimates better?

- Does global MCMC estimation make fiber trajectory estimates better?

1.5 Outline

Chapter 2 introduces the DW-MRI toolkit we developed for fitting and sampling parametric models to DW-MRI data. Next, chapter 3 lists the experiments performed and the results which illustrate the applicability of the toolkit. Finally, chapter 4 concludes this thesis and gives suggestions for further research.

Chapter 2

A DW-MRI Toolkit

The primary contribution of this thesis to the field of DW-MRI is a Matlab toolkit aimed at pre-processing and processing DW-MRI data. Pre-processing includes loading DW-MRI data in various formats, creating a brain mask based on the unweighted MRI signal and normalizing the data using the unweighted MRI signal. Processing includes fitting a multitude of parametric models to the MRI data using a point-optimal algorithm and, using a sampler, to generate samples from the posterior parameter distributions. The main advantage of this toolkit over other toolkits such as Camino [7] is that it is easily extendable to new parametric models, features sampling with Automatic Relevance Detection (ARD), and that is it written in Matlab. The latter item is an advantage, since Matlab is a widely recognized and accepted mathematical software platform and as such our toolkit is available to a large group of researchers.

The structure of this chapter follows the steps from raw DW-MRI data to optimized local tissue models. Section 2.1 explains the pre-processing the MRI data, section 2.2 lists the various tissue models implemented, section 2.3 explains how the model is fitted to the data and finally section 2.4 explains the sampling of the posterior parameter distributions.

2.1 Data pre-processing

Loading MRI data When loading DW-MRI using a combination of multiple third-party libraries it is possible to load DW-MRI data from most of the popular formats into the right format for this toolkit, which is a four-dimensional matrix of type double (64 bits of storage per entry). The first three dimensions of this matrix are the locations of the voxels and the last dimension is the signal intensity for each gradient direction of the scheme file. The scheme file (a two-dimensional matrix ,also of type double), should be loaded seperately. This can either be in a basic format $\{g,b\}$ with as columns the gradient direction $g = \{g_x, g_y, g_z\}$ and the diffusion weighting b or in a more advanced format $\{g,b,G,\Delta,\delta,TE\}$, with G the gradient strength, Δ the pulse separation, δ the pulse duration, and TE the echo time for the measurement. Some of the parametric methods discussed in section 2.2 require this more advanced format.

Creating a brain mask After having loaded the dataset it is now possible to create a brain mask. This is a three-dimensional matrix of type boolean (1 bits of storage per entry), masking the voxels which are outside of the brain and irrelevant for tracking white matter. Locating these voxels is done by thresholding on the unweighted MRI signal. An unweighted MRI signal is an MRI measurement with b-value zero, i.e. an MRI image without diffusion weighting. In general, during acquisition multiple unweighted signals are measured and the average of these is taken to represent the unweighted signal. Since air contains very few water molecules it has a very low MRI intensity. This results in a higher unweighted signal inside the brain than outside the brain. This makes it possible to create a brain mask by masking all voxels in which the unweighted signal is lower than a certain threshold.

Pre-processing MRI data The next and last pre-processing step is to normalize the data in such a way that the signal values are between 0 and 1. Again this can be done by using the unweighted MRI signal. In brain tissue the unweighted signals should always be higher than the diffusion weighted signals (if not, the weighted and/or the unweighted signal will contain noise). This means that for each voxel we can divide the diffusion weighted MRI signals by the mean of the unweighted MRI signals. Next, the unweighted signals are removed from the four-dimensional signal matrix and from the scheme file.

2.2 Tissue models

Brain tissue consists of multiple compartments which each have their own characteristic diffusion pattern. When modeling DW-MRI data the output of multiple models is combined to form the output of the total model. In this thesis the available models are partitioned into three compartment types. These compartment types are a) water inside the axons (fig 2.1a), b) water from outside the axons (fig. 2.1b) and c) water from other cellular structures, such as glial cells, trapped water on membranes or non-parallel fibres. We refer to models of these types as a) intra-axonal, which are restricted models of diffusion, b) extra-axonal, which include isotropic and anisotropic non-restricted models and c) compartments of isotropic restriction. Isotropic models are meant to model unrestricted water molecules, water molecules which can diffuse in most of the directions. Anisotropic models are meant to model restricted water diffusion in which diffusion in one direction is larger than in another.

When using multiple models to model DW-MRI data, we use the name multi-compartment models for the total model (the combination of the sub-models) even if only models of one compartment are used. The total diffusion MRI signal for a multi-compartment model is

$$S = \sum_{i=0}^{n} f_i S_i$$

where n is the number of models, S_i is the normalized MRI signal of each of the models and f_i is the weighting of the models with

$$\sum_{i=0}^{n} f_i = 1, \;\; 0 \leq f_i \leq 1 \; \forall i$$

In his paper Panagiotaki [17] lists a number of known tissue models. To make results comparable and reproducable, all his methods are implemented in this toolkit. Each of the models listed below generate a signal S. This signal is the signal for one gradient direction g with diffusivity b. To get the total signal for a voxel these signals must be summed for each g and b in the scheme file.

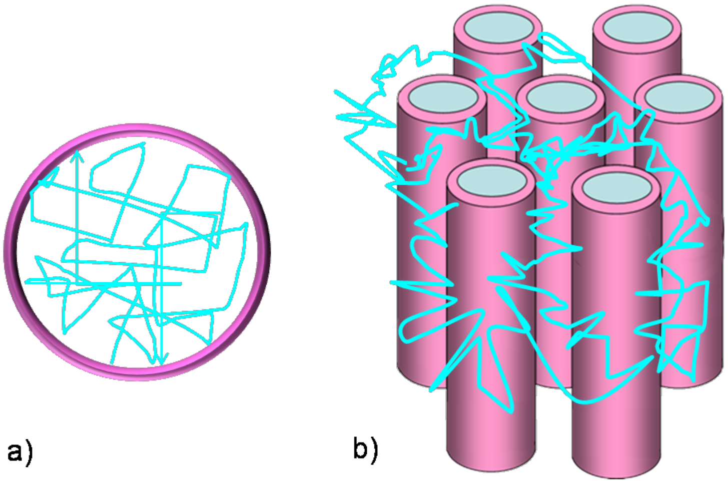


Figure 2.1: Figure showing how diffusion is limited by intra- and extra-axonal biological tissue. a) inside each cell the motion of water molecules is restricted by the cellular walls. b) biological cells hinder the motion of extra-cellular water molecules.

Intra-axonal models

1. The first is the "Stick" models of Behrens et al. 2003 [4]. This anisotropic model is meant to model fibres with direction n and diffusivity d along gradient direction g with diffusivity b. The signal for this model is

$$S = e^{-bd(g \cdot n)^2}$$

2. The second model is the "Cylinder" model of Alexander 2008 [1], using the Gaussian Phase Distribution (GPD) approximation given by Douglass and McCall [8]. This model extends the "Stick" model by adding a parameter R for a single axion radius.

3. The third model, the "GDRCylinders" extends the "Cylinder" model by having multiple cylinders with gamma-distributed radii as used in Assaf and Basser 2005 [3]. The radii are drawn from a gamma distribution

$$P(R; k, v) = \frac{R^{k-1} e^{\frac{-R}{v}}}{\Gamma(k) v^k}$$

Intra-axonal models

Model	Signal function	Degrees of freedom
	$S = e^{-bd(g \cdot n)^2}$	d, θ, ϕ
	GPD approximation, see [1] for implementation details	d, θ, ϕ, R
	$P(R; k, v) = \frac{R^{k-1} e^{\frac{-R}{v}}}{\Gamma(k) v^k}$	d, θ, ϕ, k, v

Table 2.1: Table listing the intra-axonal compartment models. The vector **n** is defined by the angles θ and ϕ. Illustrations taken from [17].

where Γ is the gamma function. The shape parameter k and scale parameter v define the cylinders' radii.

Extra-axonal models

1. The first of the extra-axonal is the "Ball" model as described by Behrens et al. 2003 [4]. This model only has the diffusivity d as parameter and is meant to model isotropic signals. The signal function is given by:

$$S = e^{-db}$$

2. The second model is the "Zeppelin" model as in Alexander 2008 [1]. This model has as parameters n, the principal direction of diffusion, $d_{\|}$ the diffusivity parallel to the principal direction and $d_{\perp}$ the diffusivity perpendicular to the principal direction. The resulting signal is:

$$S = e^{-b((d_{\|} - d_{\perp}) \cdot (g \cdot n)^2 + d_{\perp})}$$

3. The last model is a full tensor model referred to as "Tensor". This model extends the "Zeppelin" model by having a third diffusivity parameter $d_{\perp_2}$ and a parameter α rotating the tensor around its principal axis. By having this extra parameters it has three principal directions, n, $n_{\perp}$ and $n_{\perp_2}$. The signal function is:

$$S = e^{-b(d_{\|}(g \cdot n)^2 + d_{\perp}(g \cdot n_{\perp})^2 + d_{\perp_2}(g \cdot n_{\perp_2})^2)}$$

Extra-axonal models

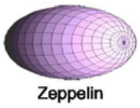
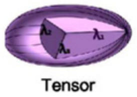
Model	Signal function	Degrees of freedom
	$S = e^{-db}$	d
	$S = e^{-b((d_{\parallel} - d_{\perp}) \cdot (g \cdot n)^2 + d_{\perp})}$	$d_{\parallel}$, $d_{\perp}$, θ, ϕ
	$S = e^{-b(d_{\parallel}(g \cdot n)^2 + d_{\perp}(g \cdot n_{\perp})^2 + d_{\perp_2}(g \cdot n_{\perp_2})^2)}$	d, θ, ϕ, k, v

Table 2.2: Table listing the extra-axonal compartment models.

Isotropic restriction models The last set of models are intended to capture all other cellular structures not captured by the previous methods. All of these models assume isotropic restriction. The "Astrosticks" and the "Astrocylinders" assume restrictions from isotropically oriented cylinders and the "Sphere" and "Dot" assume spherically bounded restrictions. For implementation details see [17]. The models are as follows:

1. The "Astrosticks" model has cylinders with uniformly distributed orientation and zero radius. It is used to represent populations of arbitrarily oriented axons.

2. The next model "Astrocylinders" extends the "Astrosticks" by adding a non-zero radius R.

3. The "Sphere" model is a spherical model modelling signal coming from water molecules inside spherical glial cells [20].

4. The "Dot" model is a special case of the "Sphere" model with a radius of 0. Alexander et al. in [2] suggests that the "Dot" model is used to model signals coming from molecules trapped in glial cells or stuck into cellular membranes in fixed tissue. The generated signal is $S = 1$.

Note on implementation Some of the methods have as parameter n, a three dimensional vector, for the principal direction of diffusion. The "Tensor" model also has the parameters $n_{\perp}$ and $n_{\perp_2}$. Optimizing n would require optimizing three parameters, but we can do better than that, by optimizing two angles, θ and ϕ. The vector n can then be calculated using:

$$n = [cos(\phi)sin(\theta), sin(\phi)sin(\theta), cos(\theta)]^T$$

Isotropic restriction models

Model	Signal function	Degrees of freedom
Astrosticks	$S_G = \int S_r p(\mathbf{n}) d\mathbf{n},\ R = 0$	d
Astrocylinders	$S_G = \int S_r p(\mathbf{n}) d\mathbf{n},\ R > 0$	$d,\ R$
	GPD approx. $R_s > 0$	$d,\ R_s$
	$R = 0,\ S = 1$	-

Table 2.3: Table listing the isotropic restriction compartment models.

For the tensor we use an additional angle α to calculate $n_\perp$ and $n_{\perp_2}$. All these vectors are normalized between 0 and 1, this is because the parameter d is used to scale the vector.

Naming convention In the remainder of this thesis combined multi-compartment models are referred to by the combined names of the sub-models. For example a "BallStickStick" model uses two "Stick" models to model intra-axonal fibres and a "Ball" model to model the isotropic extra-axonal compartments. No additional isotropic restriction model is used. Other combinations are also possible. The signal resulting from such a combined model is the weighted sum of each of the single models. Note that the order of the methods does not make a difference, i.e. the "BallStick" model is equal to the "StickBall" model.

2.3 Model fitting

In model fitting a chosen set of models are fitted to the measured MRI signal by means of an iterative procedure. That is, for a given set of models we try to find the optimal set of parameters in such a way that the models best describe the measured signal. Since the model parameters do not relate linearly to the signal a non-linear optimization routine is required. For this we use the trust-region-reflective algorithm by which we fit the models using the least squares objective function:

$$L = \sum_{n=1}^{M} (\tilde{S}_n - S_n)^2$$

where M is the number of measurements (the number of rows in the scheme file), $\tilde{S}_n$ is the model predicted signal for measurement n and S_n is the measured MRI signal for measurement n. The use of this local model fitting is for (deterministic) local tractography in which the first stage was to calculate the local main direction of diffusion.

Multiple runs During optimization it is possible to specify a perturbation function $p(\mathbf{x})$ and a scalar s which enables the optimization to run the optimization routine s times each with a different starting position. After each run of the optimization routine the optimal point is perturbated using the function $p(\mathbf{x})$ and the result is used as starting point for the next optimization run. In the end the result of the run with the lowest error value is returned. The default perturbation function uses a Gaussian distribution with $\sigma = 0.1$.

Model cascade It is also possible to fit a cascade of models to the data. For example, if we wish to fit the "BallStickStick" model it is possible to specify that first a "BallStick" should be fitted and the result should be used to initialize the "BallStickStick" model. In general, the starting positions of the more detailed model can be initialized by fitting a smaller, less detailed model. This initialization can be specified in two ways, model based and parameter based. In a model based cascade the starting point for the next run are the optimized parameters of the same model(s) of the previous run. The "BallStickStick" example above is such a model based cascade. In a parameter based cascade the parameters of the previous fit are used to initialize the same parameters in the next optimization. This allows, for example, initializing some of the parameters of the "Tensor" model with the parameters of the "Zeppelin" model.

2.3.1 Model selection

The toolkit also supports model selection based on the Bayesian Information Criterion (BIC) [19]. BIC can be used to choose the analytically most optimal model by rewarding those which minimize the objective function while penalizing increasing numbers of model parameters:

$$BIC = -2 \ln L + k \ln n$$

where L is the likelihood of the estimated model, k is the number of free parameters and n is the number of measurements. The model with the lowest BIC is the one to be preferred. One use of this functionality is for example in fitting one, two and three stick models to the same data and then voxel-wise selecting the optimal model based on BIC voxel-wise.

2.3.2 Parameter spaces

During optimization the model parameters are transformed to another parameter space. In the toolkit the two parameter spaces are referred to as "model space" and "optimization space". The models use the parameters when in model space and the optimization routine uses the parameters when in optimization space. Before optimization the starting position $\mathbf{x}_0$ is transformed from model space to optimization space using a function $f(\mathbf{x})$. The optimization routine then optimizes the parameters in optimization space. To calculate the fit between the models and the DW-MRI data, during optimization, the proposed optimization space parameters are transformed back to model space using a function $g(\mathbf{x})$.

One of the advantages of using this intermediate step during optimization is that the parameters can be constrained to biologically plausible ranges. The complete set of transformations is given in appendix A. For the experiments in chapter 3 the transformations for the "Ball" and "Stick" method are needed, so we list them here. Both the "Ball" and "Stick" method have as parameter d the diffusivity of the molecules. To restrict these parameters to positivity the square roots of these parameters are optimized. Furthermore, for numerical precision the parameter is scaled by 10^9. The transformation functions are

$$f(x) = \sqrt{10^9 x}$$

to go from model space to optimization space and

$$g(x) = \frac{x^2}{10^9}$$

to go from optimization space to model space. The "Stick" model additionally requires the direction of diffusion. As shown in the previous section this direction vector is encoded using two angles, θ and ϕ. For both these parameters $f(x) = x$ and $g(x) = x \bmod 2\pi$. The weights of a composed model are also transformed. Here the functions are $f(x) = cos^{-1}(\sqrt{x})$ and $g(x) = cos^2(x)$.

Parameter locking Another useful aspect of having this intermediate parameter transformation step is that parameters can be locked together. By locking common parameters together it is possible to decrease the number of parameters which are actually optimized. For example, the d parameter, which is the diffusivity, is a common parameter for most of the models. If assumed that the diffusivity in a voxel is equal for all compartments, the model complexity can be reduced. Also, when optimizing n models, $n-1$ model weightings are sufficient to restore the complete set of weightings. For example, the "BallStick" model requires the parameters:

$$\mathbf{x}_{model} = [f_1, f_2, d_{ball}, d_{stick}, \theta, \phi]$$

in model space. When locking the model fractions (weightings) and the d parameter, in optimization space this vector looks like:

$$x_{opt} = [cos^{-1}(\sqrt{f_1}), \sqrt{\frac{d_{ball}}{1e^9}}, \theta, \phi]$$

Note on implementation In Matlab these transformation functions are recursively constructed anonymous functions (functions without a specific source file). Recursively so that they may automatically be constructed for a given set of models and anonymous so that they may be passed around as variables. There is a small performance penalty for repeatedly transforming the parameters but this is needed to guarantee biologically plausible ranges. Also, the extra time consumption is small compared to the time in the optimization routine and finally locking some of the parameters may give a speed gain which can compensate for the performance penalty.

2.4 Sampling of the posterior distributions

There are two types of local tractography, deterministic and probabilistic. In deterministic tractography for each voxel a model is fitted to look for the set of parameters (ω) which best describe the measured data. This point estimate is then used for tracking the white matter bundles. Probabilistic models extend this information by adding uncertainty about the direction of diffusion. This information can be in the form of a standard deviation of the directions, or in the form of the complete Orientation Distribution Function (ODF). This ODF can be calculated by associating a probability distribution function (pdf) to the parameters. In a Bayesian framework the ODF is given by the *posterior distribution* on the parameters given the data:

$$P(\omega|Y, M) = \frac{P(Y|\omega, M)P(\omega|M)}{P(Y|M)} \tag{2.1}$$

where Y is the data and M is the model. This posterior density allows us to pose questions like "Given a hypervolume $\mathcal{V} \in \Omega$ and the measured data, to what extend can we assume that the true value of ω is in $\mathcal{V}$?". Such questions and their answers represent the uncertainty we have with respect to the values of the parameters ω. Unfortunately, calculating the denominator of the posterior density function in eq. 2.1 involves an integral over the parameter space Ω:

$$P(Y|M) = \int_{\Omega} P(Y|\omega, M)P(\omega|M)d\omega \tag{2.2}$$

which often is not tractable analytically. This *joint* posterior *pdf* is often not the disribution we are interested in. Most often we are interested in the posterior pdf of a single parameter or a subset of parameters. Obtaining these marginal distributions again involves performing large integrals:

$$P(\omega_I|Y,M) = \int_{\Omega - I} P(\omega|Y,M) d\omega_{-I} \tag{2.3}$$

Where ω_I are the parameters of interest and ω_{-I} are all other parameters. Again, these integrals are often not tractable analytically. One solution to approximating the joint posterior distribution is by using a sampler to draw samples in parameter space from the joint posterior distribution. In this way we implicitly calculate the integrals numerically. The sampler used in this thesis is the Metropolis Hastings sampler (MH sampler), which uses the Markov Chain Monte Carlo (MCMC) algorithm [6]. The MH sampler is able to simulate multivariate distributions by proposing samples preferentially in areas of high probability. A downside of MH is that drawn samples are not independent of each other. An upside is that the high probability of acceptance allows for many samples to be drawn in a relative short period of time.

Optimization with sampling The sampling can also be used for optimization purposes by taking the mean of the sampled posterior distribution. This is especially useful in combination with Automatic Relevance Detection (see subsection 2.4.2).

2.4.1 The posterior distribution function

The posterior distribution function is a product of two parts, the likelihood distribution of the parameters and the prior distribution on those parameters. The likelihood function for a single measurement y_n is the probability of observing the data under the given models and parameters: $P(Y|\omega, M)$. Assuming Rician noise [1] this likelihood is given as:

$$P(y_n|\omega, M) = R(S_n|\tilde{S}_n, \sigma)$$

where S_n is the measured signal for measurement n, $\tilde{S}_n$ is the model predicted signal and the function

$$R(x|\mu, \sigma) = \frac{x}{s^2} \cdot e^{\frac{-(x^2+v^2)}{2s^2}} \cdot B_0(\frac{xv}{s^2})$$

is the probability density function for the Rician noise model with $B_0(z)$ the zeroth order modified Bessel function of the first kind. See figure 2.2 for a graph of the pdf of the Rice distribution. The total likelihood function (for all the measurements) is given by:

$$P(Y|\omega, M) = \prod_{n=1}^{M} P(y_n|\omega, M)$$

The second part of the posterior distribution function is the prior over the parameters. The complete list of prior functions is given in Appendix B, here it suffices to give the priors for the d, θ and ϕ parameters of the "Ball" and "Stick" models and the prior for model weightings. The prior for the d parameter is $p(x) = I_{(x>0)}$ which constrains d to positive values. Here the function $I_{(z)}$ is an indicator function returning a 1 if the expression z evaluates to true and a 0 if z evaluate to false. For θ the prior is $p(x) = I_{(x>0)} \cdot |\sin x| \cdot I_{(x\leq 2\pi)}$ which assures values between 0 and (inclusive) 2π, and the absolute sinus ensures we cover the entire angle space. For ϕ the prior is $p(x) = U(x|0, 2\pi)$ with $U(x|a, b)$ the uniform distribution between a and b. All the weightings f_i are restricted to values between 0 and 1, and the sum of the weightings should be lower than 1 and higher than 0.98, so: $0.98 < \sum_{i=0}^{n} f_i < 1$.

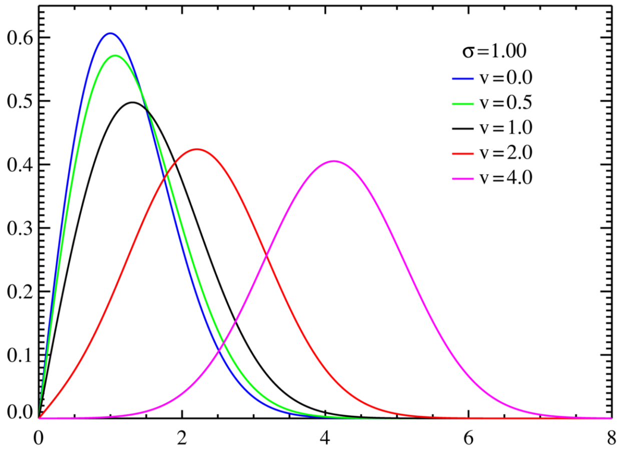


Figure 2.2: Figure showing the Rice probability density function used for Rician noise.

Parameter spaces For the optimization we used two transformation functions to convert the parameters between parameter spaces. The same is done during sampling except that here most of the transformations are identity functions. Only the the d parameter is scaled again for precision and the angles θ and ϕ are transformed in the same manner. Parameter locking is supported in the same way as for the optimization transformations.

2.4.2 Automatic Relevance Detection

Automatic Relevance Detection (ARD) is a model selection technique used in a Bayesian setting. ARD, or shrinkage priors, were originally devised for use in Neural Networks by McKay 1995 [15]. It has since been used in Neuroimaging [5] as a technique for inferring multiple fibre orientations if, and only if, there is evidence in the data to support them. If we tried to fit multiple fibre directions to a voxel which truly only has one, that fit may lead to a bad estimate of the local fibre orientation. Such minor errors in local fibre fitting may accumulate and provide a significant error in tracked fibres.

Most model selection techniques require fitting multiple models of varying complexity from which the optimal model will be selected based on a metric on the goodness of fit and the model complexity. ARD differs from these models in that it will fit the more complex model but also ensures that parameters which are not supported by the data, do not contribute to the likelihood [5]. ARD can be applied to a number of parameters and will act independently on each of those parameters.

Applying ARD The first step in using ARD is to select one or more parameters on which ARD will function. The next step is that a prior distribution is placed on each of the selected parameters, which will force that parameter to zero if, and only if, there is no evidence in the data for its existence. The most common prior is a Gaussian distribution with mean zero, but unknown variance. This variance is added as a hyperparameter to the sampling routine. If there is no evidence in the data for the existence of one of the selected parameters the hyperparameter will be estimated as very small, forcing the original parameter to zero. However, if there is evidence in the data for the existence of the parameter, the hyperparameter (the variance of the prior) will be very large, allowing the original parameter to take any value. In this setup the posterior distribution we try to sample is multiplied by the prior distribution(s) of the ARD (of each parameter) and by the prior of the hyperparameter(s).

ARD on model weightings In this thesis we will use ARD on the weightings of one or more of the submodels of a combined multi-compartment model. This allows ARD to select which and how many models are supported by the data. For example, if we fitted a "BallStickStickStick" with ARD we could place a prior on the weightings of one or more of the "Stick" models so that we only infer multiple "Sticks" (and thus multiple fibres) if there is support for this in the data. Since the model weightings are restrained between 0 and 1 we will use the Beta distribution for the priors instead of the Gaussian distribution. This means that for each weighting we apply ARD on, we need to multiply the posterior distribution with the shrinkage prior:

$$P(f_i|\eta) = \beta(f_i|1, \eta)$$

and with the prior of the hyperparameter:

$$P(\eta) = \frac{1}{\eta}$$

in which η is the hyperparameter of the ARD and $\beta(x|a, b)$ is the Beta *pdf* with parameters a and b.

Chapter 3

Experiments and results

In this chapter we will present some experiments performed, and results obtained by applying the toolkit to various DW-MR images of human brain tissue. In his paper, E. Panagiotaki [17] provides a taxonomy of various compartment models; we will not repeat those experiments here. Instead we will focus on one model, the ”BallStick” model of Behrens 2003 [4] extended to three sticks, i.e. ”BallStickStickStick”. This model uses the ”Ball” model for isotropic compartments and three ”Stick” models for modelling anisoptric compartments. The direction of the fitted ”Stick” models represents the supposed direction of the underlying fibres (see Chapter 2). Both the ”Ball” as the ”Stick” model are relatively simple models with only a few parameters, which makes them relatively easy to fit to the data. Furthermore it is easy to optimize the models in a cascade by first fitting ”BallStick”, then ”BallStickStick” and finally ”BallStickStickStick”.

The experiments in this chapter use two datasets, the Skyra and the Juelich dataset, named after the name of the MRI scanner which scanned the tissue. The Skyra dataset is the result of a whole brain measured *in vivo* with a whole-body 3T MRI machine scanned at b-value $3e^9 s/m^2$ using 137 gradient directions of which 9 are unweighted. The Juelich dataset is measured *ex vivo* on a small sample of *post-mortem* human tissue (motor cortex) on a 9.4T small bore MRI system scanned at b-value $2.7250e^9 s/m^2$ using 66 gradient directions of which 6 are unweighted.

This chapter is divided into two sections. Section 3.1 discusses model fitting using the optimization routine and section 3.2 discusses parameter sampling using the Metropolis Hastings sampler and model fitting using Metropolis Hastings sampling (MH sampling) with Automatic Relevance Detection (ARD).

3.1 Model fitting

This section shows the results of fitting the ”BallStickStickStick” model to the two datasets mentioned. The Skyra set was optimized using a cascade of one stick, two sticks and finally three sticks of which each intermediate step is run five times. The Juelich set was optimized with three sticks in only one run, due to time constraints. The results of the fitting are visualized using Trackmark, a program written by Alard Roebroeck. In this visualization the gray-scale map is the sum of all the "Stick" model fractions, with white a high value (close to 1) and black a low value (close to 0). The directions of the "Sticks" are visualized using cylinders which are color coded by their direction in the brain. Red is from left to right (ear to ear), blue is from superior to inferior (top of skull to spinal cord) and green from posterior to anterior (back to front). In the visualization only those directions are shown in voxels of which the summed stick fraction is higher than 0.2.

Skyra dataset The first visualization is that of the model fitting on the Skyra dataset. Figure 3.1 shows a central axial slice of the brain containing the Corpus Callosum. The second figure, figure 3.2 shows a coronal slice with a region marked in red of which a close-up is found in figure 3.3. This region is of interest since in this region three major fibre bundles cross each other. At least two of these fibres are clearly visible. In some places there appear to be gaps in places where it seems quite evident that at least one "Stick" should be present. It could be that in these places multiple anisotropic signals are present in such a way that fitting them all at once with a large isotropic "Ball" model gives a better fit.

Juelich dataset The other dataset is the Juelich dataset which was fitted once using a "BallStickStickStick" model. Figure 3.4 shows a centre slice of the dataset. Two regions are marked of which close-ups are found in figures 3.5 and 3.6.

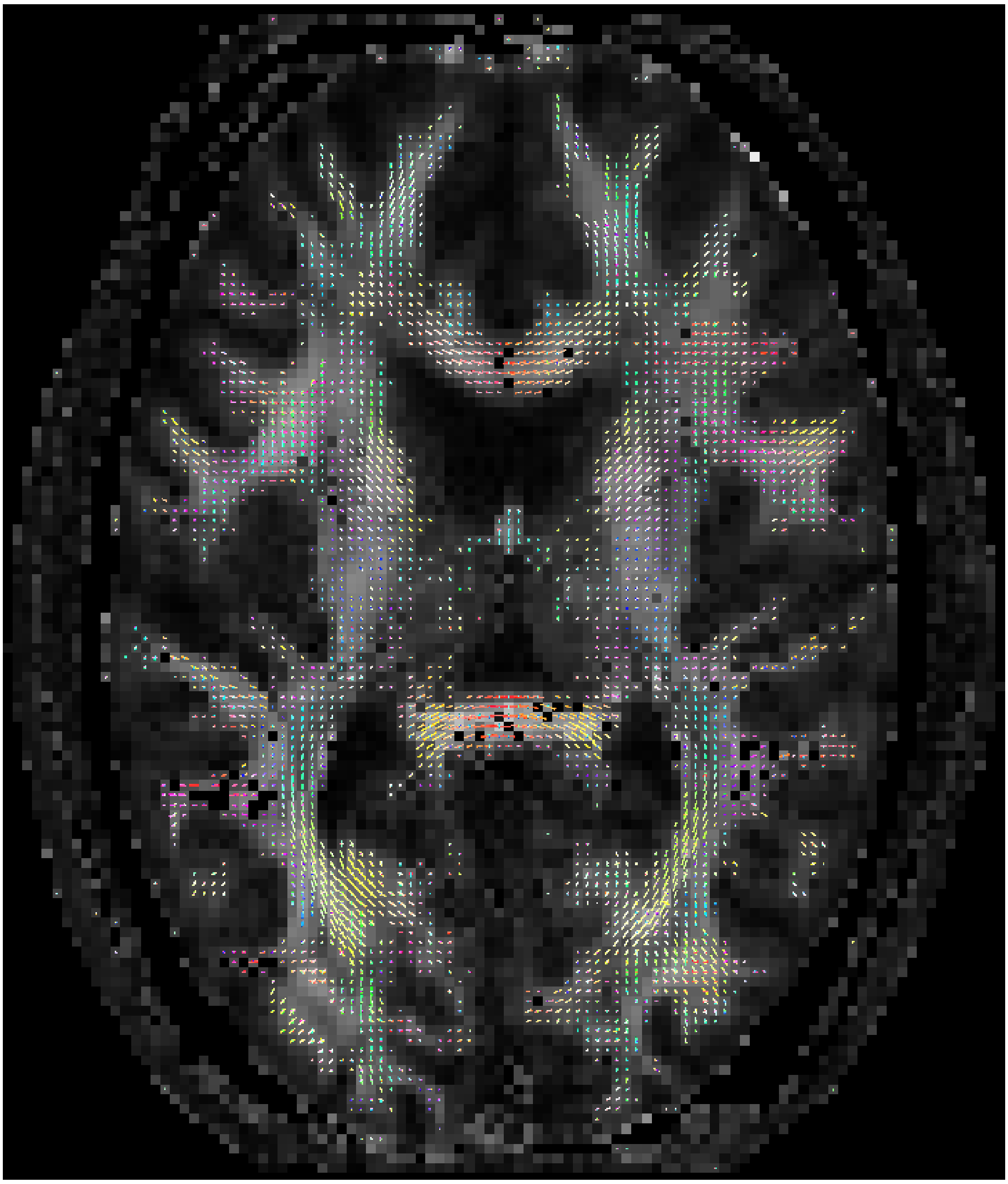


Figure 3.1: A central axial slice of the Skyra dataset. The background is the summed fraction of the three sticks with white indicating a high value and black a low value.

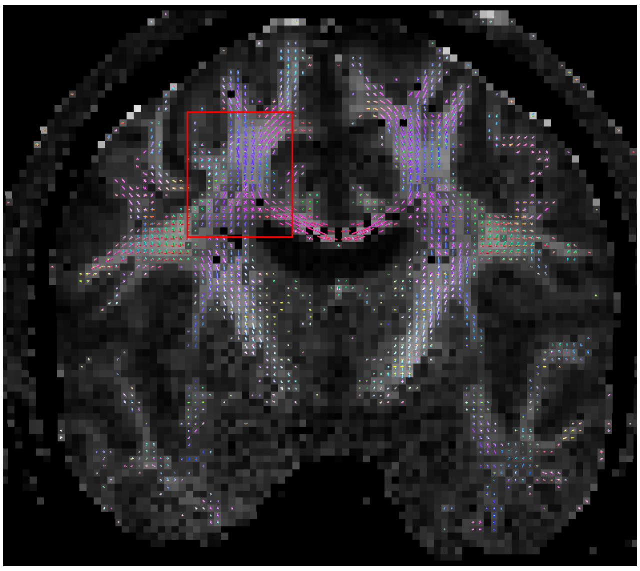


Figure 3.2: A coronal slice of the Skyra dataset. A close up of the region marked in red is found in figure 3.3.

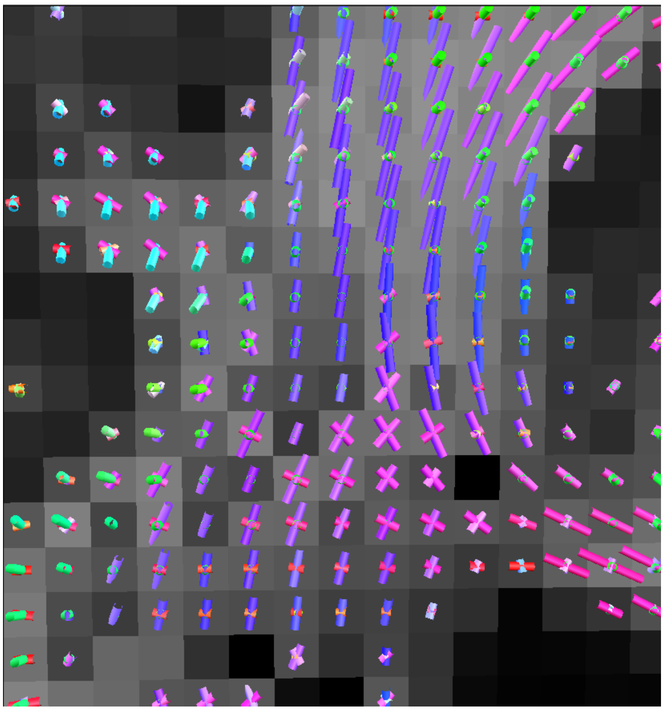


Figure 3.3: Close up of a part of a coronal slice of the Skyra dataset. The full slice is visible in figure 3.2.

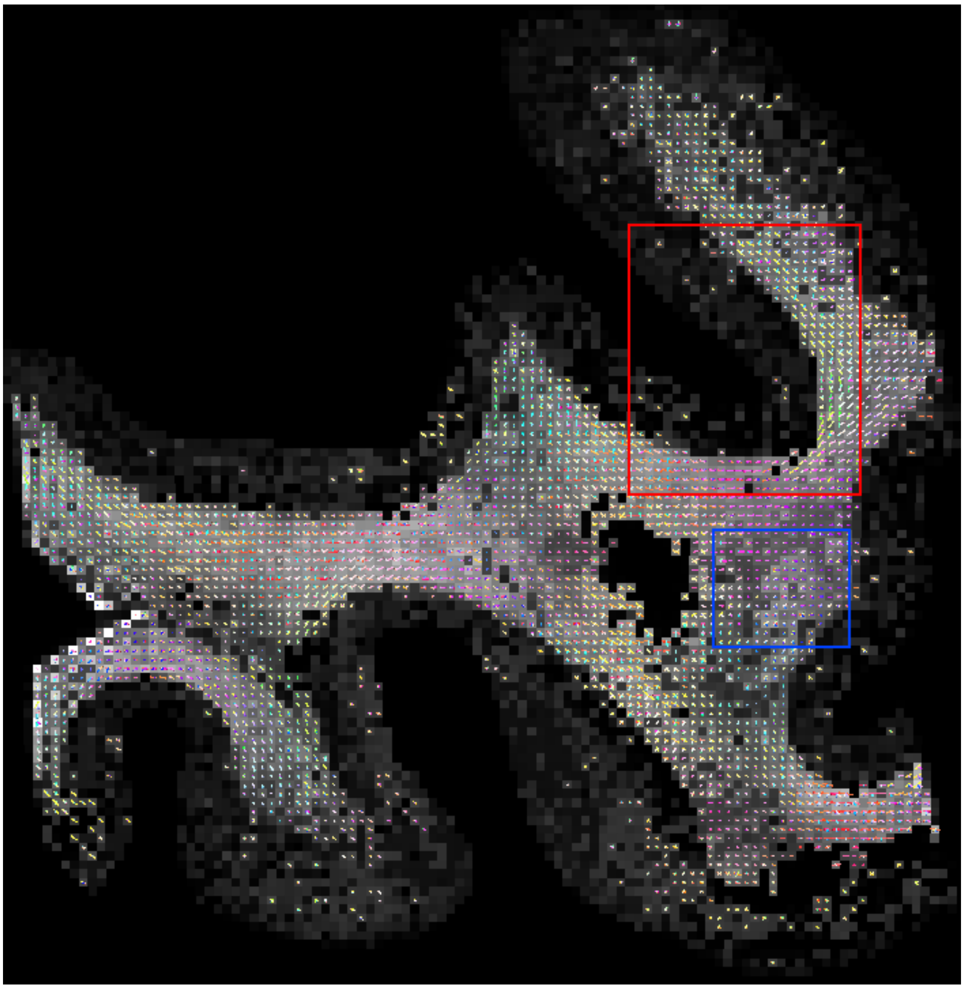


Figure 3.4: Figure showing an overview of a slice of the Juelich dataset. A close-up of the region marked in red is found in figure 3.5 and a close-up of the region marked in blue is found in figure 3.6.

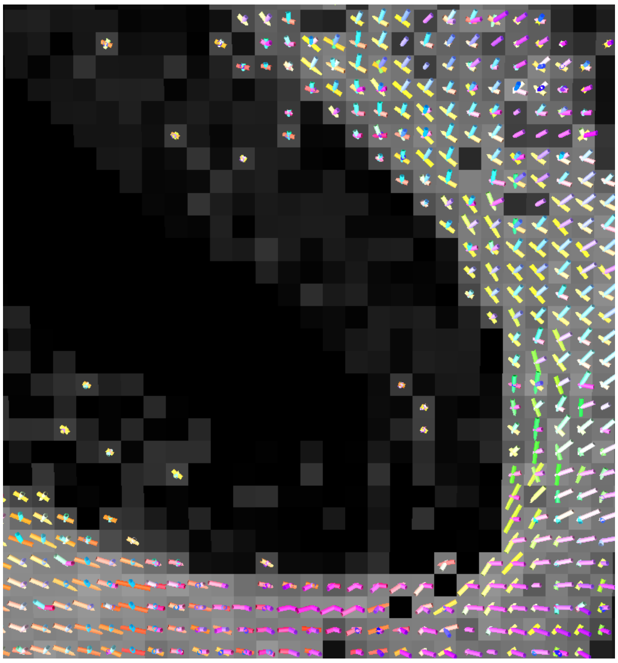


Figure 3.5: Zoomed-in region of the slice of figure 3.4. This figure shows how the fitted model found the track (in yellow) around the cortex.

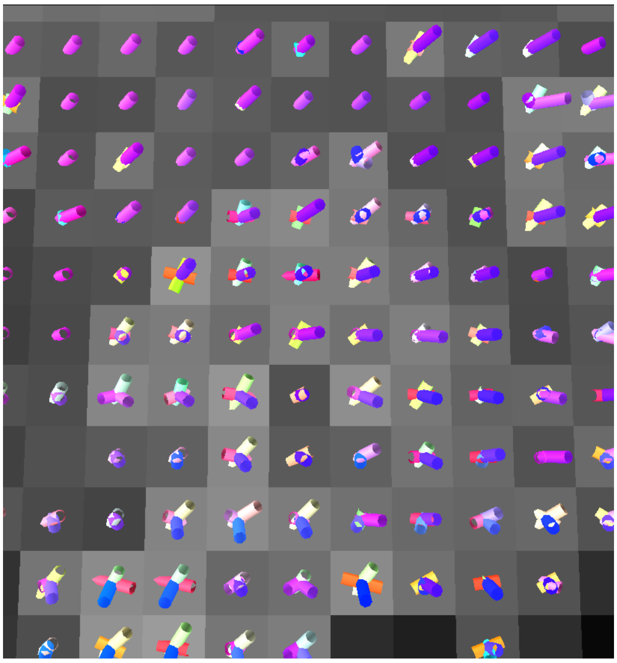


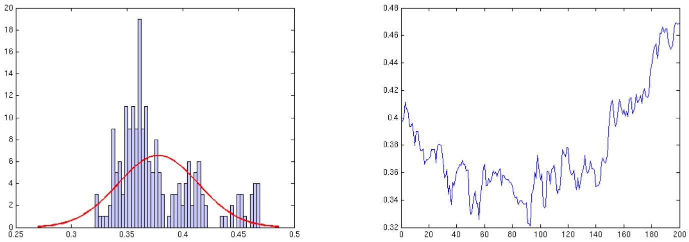
Figure 3.6: Zoomed-in region of the slice of figure 3.4. A nice illustration of the fit of three ”Sticks”.

3.2 Metropolis Hasting sampler

In this section two illustrations are given of the functionability of the MH sampler with ARD. For the first illustration we will apply the sampler to a single voxel of the Skyra dataset and show the samples of the model fraction. The second illustration shows the result of sampling an axial slice of the Skyra dataset.

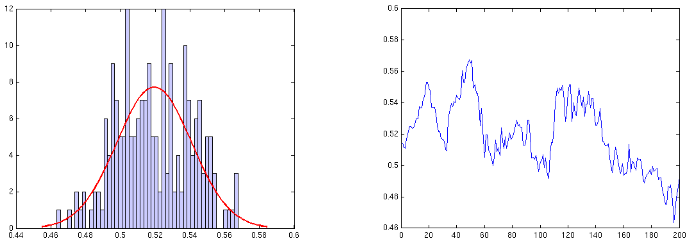
Sampling a voxel The voxel chosen for the demonstration of the single voxel is one of the Corpus Callosum of which it is known that it has one prevailing direction of diffusion. This suggests that ARD should limit the amount of sticks to one, or at most two. We will sample three "Stick" models to this voxel of which the first is initialized using the optimization routine and the other two will be governed by ARD. The sampler is initialized with 2000 steps without sampling after which 200 samples are taken with intermediate jumps of 10 steps. This gives a total of 4000 steps and 2000 samples. Of the resulting samples an illustration is given of the four model fractions. At the left is an histogram of the sampled values with in red a fitted Gaussian on the histogram. On the right the chain of the samples is given. This chain is an important cue for choosing the correct proposal distributions. If the chain appears to linger in one area too long the step-size is too small and many proposed samples will not be accepted. If this is the case the resulting samples will give a bad estimation of the sampled distribution and either more samples are needed or the step-size needs to be increased. Figure 3.7 gives the overview of the samples of the "Ball" model fraction, followed by figure 3.8, 3.9 and 3.9 of the samples of "Stick" model 1, 2 and 3 respectively. In the figures, the left figure gives an histogram of the samples with the weight of the model on the x-axis and the count of accepted samples with that probability on the y-axis. In this figure a graph is plotted of a Gaussian fit to the histogram. The mean of this is taken as the point-optimal model weight. The figure on the right gives the chain followed by the sampler. Here the x-axis is the sample number and the y-axis is the sampled probability. As predicted ARD limits the weighting of the second and third stick for lack of evidence as can be seen by the low probabilities of the weights of the accepted samples.

Sampling a slice The next illustration gives the sampling results of an entire slice of the Skyra set (the same slice as for figure 3.1 in the previous section). This slice was sampled again by initializing the first stick using the optimization routine; sticks two and three are again governed by ARD. The sampler is then run for a burn-in of 2000 jumps after which 200 samples are taken with intervals of 10 jumps. For each voxel and for each parameter the mean and variance are calculated and returned in a five-dimensional (5D) matrix (three for the location of the voxels, one for the parameters and the last for the mean and variance). For this illustration only the mean is used to generate the directions per voxel, which can again be visualized using TrackMark. The results are shown in figure 3.11. When we compare this sampling result with the optimization result of the former section the sampling result contains more noise. This can have multiple causes, for example, the number of burn-in jumps is too few, more intermediate jumps are required between samples or the proposal distributions need tuning.



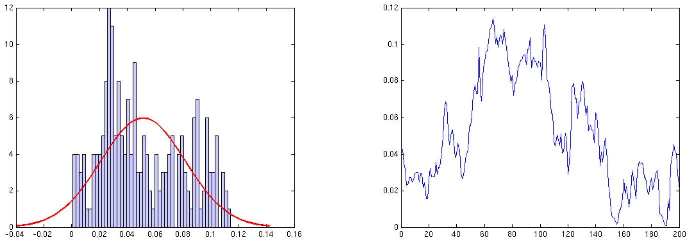
(a) Histogram of the samples (b) Chain of samples

Figure 3.7: Sampling results of "Ball" fraction of a single voxel



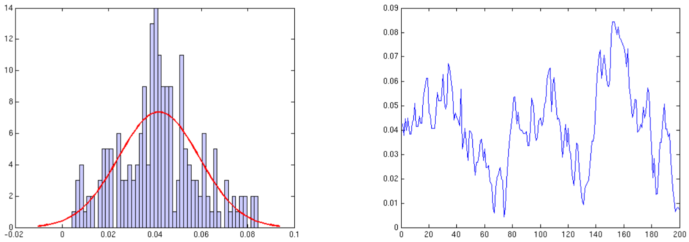
(a) Histogram of the samples (b) Chain of the samples

Figure 3.8: Sampling results of the first "Stick" fraction of a single voxel



(a) Histogram of the samples (b) Chain of the samples

Figure 3.9: Sampling results of the second "Stick" fraction of a single voxel



(a) Histogram of samples (b) Chain of the samples

Figure 3.10: Sampling results of the third "Stick" fraction of a single voxel

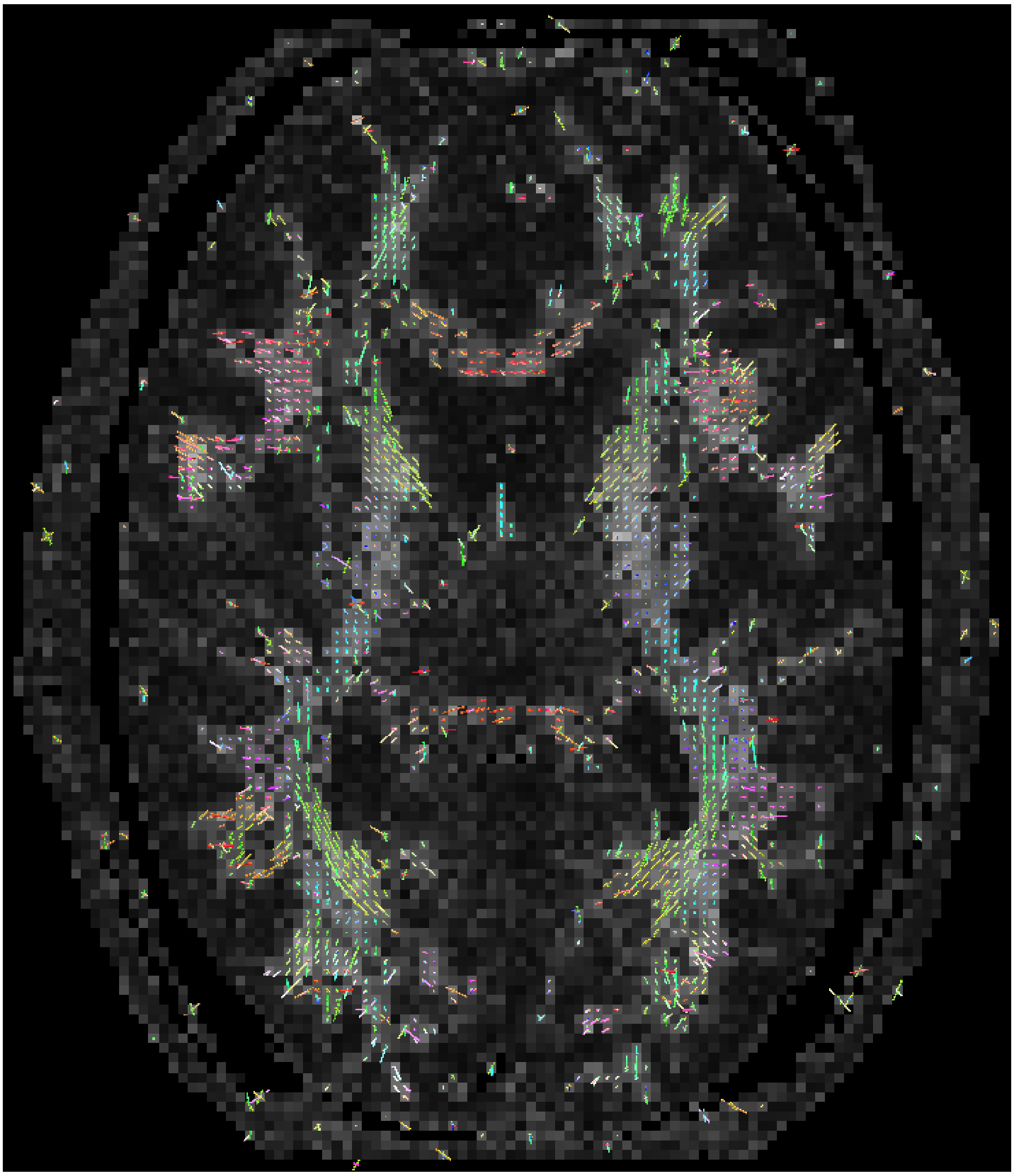


Figure 3.11: A central axial slice of the Skyra dataset. Sampled with a Metropolis Hastings sampler with Automatic Relevance Detection. The background is the summed fraction of the Stick models, with black for low value and white for high.

Chapter 4

Conclusions

The goal of this thesis was to develop a toolkit for local and possibly in the future global tractography. This goal is met for the toolkit contains multiple popular local diffusion models for use in both local and global tractography. The part of local directions is covered by the toolkit presented in this thesis and with it a foundation for global tractography, which is left for further development.

Before local or global tractography can be performed using parametric models an implementation of those parametric models is needed. In his paper Panagiotaki [17] lists a number of different local diffusion models currently in use in the field of Neuroscience. All of these methods are programmed in a toolkit called Camino [7] which can fit these models to a given Diffusion Weighted Magnetic Resonance Image (DW-MRI) dataset. Camino however has numerous drawbacks, which lead us to develop a new toolkit for fitting parametric models to DW-MRI data. This new toolkit, as presented in this thesis, contains all the parametric models of Camino and extends Camino with Automatic Relevance Detection of model parameters.

The toolkit The presented toolkit was built in Matlab, which allows it to run on any operating system capable of running Matlab. It contains a number of parametric diffusion models and a fitting routine which can fit a set of models to DW-MRI data. Next to that, a sampling routine was built which allows sampling from the posterior distribution of model parameters, using a Metropolis Hasting sampler. This sampling can be used for local probabilistic tracking, but is also a part of global tractography. Finally this toolkit has a few supportive functions for loading and normalizing DW-MRI dataset from various containers and was built in such a way that it is easily extendable to include new parametric models.

Further research The toolkit presented here is a large step in the direction of global tractography, but it is not yet the end point. The next step in this field of research would be to implement a datastructure for storing globally tracked fibres and to program the connection between global fibres and the local diffusion models. This can follow the structure of Reister et al. 2011 [18], but with different combinations of local diffusion models.

Bibliography

[1] D.C. Alexander. A general framework for experiment design in diffusion mri and its application in measuring direct tissue-microstructure features. *Magnetic Resonance in Medicine*, 60:439 – 448, 2008.

[2] D.C. Alexander, P.L. Hubbard, M.G. Hall, E.A. Moore, M. Ptito, G.J.M. Parker, and T.B. Dyrby. Orientationally invariant indices of axon diameter and density from diffusion MRI. *Neuroimage*, 52:1374 – 1389, 2010.

[3] Y. Assar and P.J. Basser. Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain. *Neuroimage*, 27:48 – 58, 2005.

[4] T.E. Behrens, M.W. Woolrich, M. Jenkinson, H. Johansen-Berg, R.G. Nunes, S. Clare, P.M. Matthews, J.M. Brady, and S.M. Smith. Characterization and propagation of uncertainty in diffusion-weighted mr imaging. *Magnetic Resonance in Medicine*, 50:1077 – 1108, 2003.

[5] T.E.J. Behrens, H. Johansen-Berg, S. Jbabdi, M.F.S. Rushworth, and M.W. Woolrich. Probabilistic diffusion tractography with multiple fibre orientations: What can we gain? *NeuroImage*, 34:144 – 155, 2007.

[6] S. Chib and E. Greenberg. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49:327 – 335, 1995.

[7] P.A. Cook, Y. Bai, S. Nedjati-Gilani, K.K. Seunarine, M.G. Hall, G.J. Parker, and D.C. Alexander. Camino: Open-Source Diffusion-MRI Reconstruction and Processing. In *14th Scientific Meeting of the International Society for Magnetic Resonance in Medicine, Seattle, WA, USA*, page 2759, 2006.

[8] D.C. Douglass and D.W. McCall. Diffusion in paraffin hydrocarbons. *The Journal of Physical Chemistry*, 62:1102, 1958.

[9] A. Einstein. *Investigations on the theory of the brownian movement*. Dover publications, 1956.

[10] R.D. Fields. White matter matters. *Scientific American*, 298:54 – 61, 2008.

[11] P. Hagmann, L. Jonasson, P. Maeder, J.P. Thiran, V.J. Wedeen, and R. Meuli. Understanding diffusion MR imaging techniques: from scalar diffusion-weighted imaging to diffusion tensor imaging and beyond. *Radiographics*, 26:205 – 232, 2006.

[12] S. Herculano-Houzel. The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in Human Neuroscience*, 3:3 – 31, 2009.

[13] S. Jbabdi and H. Johansen-Berg. Tractography: Where Do We Go from Here? *Brain Connectivity*, 1:169–183, 2011.

[14] K.R. Jessen and R. Mirsky. Glial cells in the enteric nervous system contain glial fibrillary acidic protein. *Nature*, 286:736 – 737, 1980.

[15] D.J.C. MacKay. Probable networks and plausible predictions - a review of practical bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6:469 – 505, 1995.

[16] K. Narayanan and J.W. Lichtman. The rise of the 'projectome'. *Nature Methods*, 4:307 – 308, 2007.

[17] E. Panagiotaki, T. Schneider, B. Siow, M.G. Hall, M.F. Lythgoe, and D.C. Alexander. Compartment models of the diffusion MR signal in brain white matter: A taxonomy and comparison. *NeuroImage*, 59:2241 – 2254, 2012.

[18] M. Reisert, I. Mader, C. Anastasopoulos, M. Weigel, S. Schnell, and V. Kiselev. Global fiber reconstruction becomes practical. *NeuroImage*, 54:955 – 962, 2011.

[19] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6:461 – 464, 1978.

[20] G.J. Stanisz, A. Szafer, G.A. Wright, and R.M. Henkelman. An analytical model of restricted diffusion in bovine optic nerve. *Magnetic Resonance in Medicine*, 37:103 – 111, 1997.

Appendix A

Parameter transformations

This table lists the transformation functions which transform the parameters of the various implemented models to optimization space ($f(x)$) and back to model space ($g(x)$).

Model parameter	Optimized parameter	Transformation f(x)	Transformation g(x)
d	θ_1	$\theta_1 = \sqrt{10^9 d}$	$d = \frac{\theta_1^2}{10^9}$
R	θ_2	$\theta_2 = \cos^{-1}(\sqrt{5e^4(R - 1e^{-7})})$	$R = 1e^{-7} + 20e^{-6}\cos^2(\theta_2)$
θ	θ_3	$\theta_3 = \theta$	$\theta = \theta_3 \bmod 2\pi$
ϕ	θ_4	$\theta_4 = \phi$	$\phi = \theta_4 \bmod 2\pi$
k	θ_5	$\theta_5 = \cos^{-1}(\sqrt{0.05k})$	$k = 20\cos^2(\theta_5)$
β	θ_6	$\theta_6 = \lvert\cos^{-1}(\sqrt{5e^4(\beta - 1e^{-7})})\rvert$	$\beta = 1e^{-7} + 20e^{-6}\cos^2(\theta_6)$
bins	θ_7	$\theta_7 = \cos^{-1}(\sqrt{0.01(bins - 1)})$	bins $= \lfloor 1 + 10\cos^2(\theta_7) \rfloor$
R_s	θ_8	$\theta_8 = \cos^{-1}(\sqrt{2.5e^4(R_s - 1e^{-7})})$	$R_s = 1e^{-7} + 40e^{-6}\cos^2(\theta_2)$
$d_{\perp_1}$	θ_9	$\theta_9 = \sin^{-1}(min(1, \sqrt{d_{\perp_1}/d}))$	$d_{\perp_1} = \sin^2(\theta_9) \cdot d$
$d_{\perp_2}$	θ_{10}	$\theta_{10} = \sin^{-1}(min(1, \sqrt{d_{\perp_2}/d_{\perp_1}}))$	$d_{\perp_2} = \sin^2(\theta_9) \cdot d_{\perp_1}$
α	θ_{11}	$\theta_{11} = \alpha$	$\alpha = \theta_{11} \bmod 2\pi$

In this table the parameter k, β and bins are specific to the "GDRCylinders" model and α is for rotating the "Tensor" model.
The weightings are transformed to optimization-space using the functions $f(\mathbf{w}) = \{cos^{-1}(\sqrt{w_i}) : i < n\}$ with $\mathbf{w}$ the model weightings and n the number of models. The function $g(\mathbf{w})$ uses the following algorithm:

```
t = cos(w)^2
t = [t, 1-sum(t)]
if(min(t) < 0)
        t = t - min(t)
end
return t/sum(t)
```

Appendix B

Priors for posterior distribution sampling

This table lists the prior function for each parameter used during the sampling process of the posterior pdf.

Model parameter	Prior function
d	$p(x) = I_{(x>0)}$
R	$p(x) = \Gamma(x\|3, 3)$
θ	$p(x) = I_{(x>0)} \cdot \|\sin x\| \cdot I_{(x \leq 2\pi)}$
ϕ	$p(x) = U(x\|0, 2\pi)$
k	$p(x) = \Gamma(x\|3, 3)$
β	$p(x) = \Gamma(x\|3, 3)$
bins	$p(x) = U(x\|1, 10)$
R_s	$p(x) = \Gamma(x\|3, 3)$
$d_{\perp_1}$	$p(x) = I_{(x>0)}$
$d_{\perp_2}$	$p(x) = I_{(x>0)}$
α	$p(x) = I_{(x>0)} \cdot \|\sin x\|$

In this table the parameter k, β and bins are specific to the "GDRCylinders" model and α is for rotating the "Tensor" model. The function $I_{(z)}$ is an indicator function returning a 1 if the expression z evaluates to true and a 0 if z evaluates to false. $\Gamma(x|\theta, k)$ is the Gamma pdf and $U(x|a, b)$ is the uniform distribution. The prior information for the model weightings restricts the weightings to values between 0 and 1: $p(x) = U(x|0, 1)$ and restricts the sum of the weightings to 1.