Large-scale simulation and deep learning of diffusion MRI signatures for the development of computational models of cerebral white matter ultrastructure

Alexis Brullé

université
PARIS-SACLAY

Large-scale simulation and deep learning of diffusion MRI signatures for the development of computational models of cerebral white matter ultrastructure

Simulation à large échelle et apprentissage profond de la signature en IRM de diffusion pour le développement de modèles computationnels de l'ultrastructure de la substance blanche cérébrale

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 575, electrical, optical, biophysics and engineering (EOBE)

Spécialité de doctorat: Physique et imagerie médicale

Graduate school: Sciences de l'ingénierie et des systèmes

Référent: Faculté des sciences d'Orsay

Thèse préparée au sein de l'unité de recherche **BAOBAB** (Université Paris-Saclay, CEA, CNRS) sous la direction de **Cyril POUPON**, Directeur de recherche.

Thèse soutenue à Paris-Saclay, le 03 Juillet 2024, par

Alexis BRULLÉ

Composition du Jury

Membres du jury avec voix délibérative

Christophe DESTRIEUX Professeur, Université de Tours, Praticien Hospitalier	Président
Jean-Philippe THIRAN Professeur, Ecole Polytechnique fédérale de Lausanne	Rapporteur et Examinateur
Arnault CACHIA Professeur, Université Paris-Cité	Rapporteur
Catherine OPPENHEIM Professeur, Université Paris Descartes	Examinatrice

THESE DE DOCTORAT

NNT: 2024UPAST080

université PARIS-SACLAY

ÉCOLE DOCTORALE
Physique et ingénierie : Electrons, Photons, Sciences du vivant (EOBE)

Titre: Simulation à large échelle et apprentissage profond de la signature en IRM de diffusion pour le développement de modèles computationnels de l'ultrastructure de la substance blanche cérébrale

Mots clés: IRM, réseaux de neurones, neuroanatomie, HPC

Résumé: L'équipe Ginkgo (Microstructure) du laboratoire BAOBAB/GAIA de NeuroSpin développe dans le cadre du flagship européen Human Brain Project un environnement appelé MEDUSA (Microstructure Environment Designer Using Sphere Atoms) qui permet la création de tissus virtuels réalistes représentatifs des environnements cellulaires à même d'être rencontrés dans le cerveau humain, et qui permet donc de simuler le mouvement brownien de l'eau lié au processus de diffusion et donc de prédire le signal IRM qu'on obtiendrait pour chaque tissu virtuel à l'aide d'une IRM pondérée en diffusion. L'apport majeur des techniques d'intelligence artificielle notamment pour l'apprentissage supervisé des grands jeux de données (ou Big Data) ouvre de nouvelles perspectives quant au développement de méthodes d'imagerie in vivo de la cytoarchitecture du cortex puisqu'il devient envisageable d'exploiter l'information microscopique embarquée dans le mouvement brownien des molécules d'eau (appelé aussi processus de diffusion de l'eau) présentes dans le cerveau et dont les trajectoires embarquent une empreinte de la cytoarchitecture locale du tissu qu'il devient alors possible de décoder à l'aide de méthodes d'apprentissage profond.

Title: Large-scale simulation and deep learning of diffusion MRI signatures for the development of computational models of cerebral white matter ultrastructure

Keywords: MRI, neural networks, neuroanatomy, HPC

Abstract: The Ginkgo (Microstructure) team of NeuroSpin's BAOBAB/GAIA laboratory is developing an environment called MEDUSA (Microstructure Environment Designer Using Sphere Atoms) as part of the European flagship Human Brain Project, which allows the creation of realistic virtual tissues representative of the cellular environments that can be found in the human brain, and therefore allows to simulate the Brownian motion of water related to the diffusion process and thus to predict the MRI signal that would be obtained for each virtual tissue using diffusion weighted MRI. The major contribution of artificial intelligence techniques, particularly for supervised learning of large data sets (or Big Data), opens new perspectives for the development of in vivo imaging methods of the cortex cytoarchitecture since it becomes possible to exploit the microscopic information embedded in the Brownian motion of water molecules (also called the water diffusion process) present in the brain, and whose trajectories embody an imprint of the local cytoarchitecture of the tissue which can then be decoded using deep learning methods.

Contents

List of Figures v

List of Tables x

Acronyms xiii

Résumé de la thèse xv

Général Introduction 1

1 Human brain anatomy 3
1.1 Macroscopic anatomy . . . 4
1.1.1 Gross anatomy . . . 4
1.1.2 Brain lobes . . . 8
1.1.3 Grey matter vs white matter . . . 10
1.1.4 Brain vasculature . . . 14
1.2 Microscopic anatomy . . . 16
1.2.1 Brain cells . . . 16
1.2.2 Cortex histology . . . 22
1.2.3 White matter histology . . . 25
1.2.4 Conclusion . . . 27

2 Toward Magnetic Resonance Imaging 29
2.1 Historical background . . . 30
2.2 Fundamentals of magnetic resonance imaging . . . 32
2.2.1 Nuclear Magnetic Resonance Principle . . . 32
2.2.2 Image encoding and reconstruction . . . 36
2.2.3 Conventional MRI sequence schemes . . . 38
2.3 Diffusion MRI . . . 40
2.3.1 Brownian Motion . . . 40
2.3.2 Sensitizing MRI to the diffusion process . . . 42
2.3.3 Models of the diffusion process . . . 47
2.4 Conclusion . . . 55

3 Brain biophysical multi-compartment models 57
3.1 Brain tissues as restricted environment . . . 58
3.2 Design of hardware microstructural phantoms . . . 60
3.3 Multi compartmental analytical models . . . 61
3.4 Computational models and dMRI simulator . . . 67

3.4.1 Simulation of cell population geometries . . . 67
3.4.2 Simulate the diffusion process using Monte-Carlo simulations . . . 75
3.4.3 Synthesis of the dMRI signal . . . 80
3.4.4 Computational model . . . 81
3.5 Conclusion . . . 81

4 Microstructure Environment Designer with Unified Sphere Atoms simulator 83
4.1 The MEDUSA simulator . . . 85
4.1.1 State of the art . . . 85
4.1.2 Object oriented software design . . . 86
4.1.3 Geometry simulator . . . 86
4.1.4 Diffusion process simulator . . . 103
4.1.5 DMRI sequences and simulator of the dMRI signal attenuation . . . 105
4.1.6 Conclusion . . . 107
4.2 Driving large simulation campaigns . . . 107
4.2.1 Simulation input dictionaries . . . 108
4.2.2 Meta simulation dictionary configuration . . . 112
4.3 Validation . . . 114
4.3.1 MEDUSA reliability regarding analytical solutions . . . 115
4.3.2 MEDUSA validation in the case of complex geometries . . . 116
4.3.3 Conclusion . . . 134

5 Driving a large MEDUSA simulation campaign 135
5.1 Computational architecture . . . 137
5.1.1 Kokkos framework . . . 137
5.1.2 TGCC facility . . . 138
5.1.3 Docker virtual containers . . . 140
5.2 Monitoring large scale simulation campaigns . . . 140
5.2.1 Tuning the geometry sampling . . . 140
5.2.2 Tuning the diffusion process simulation . . . 143
5.2.3 Tuning of the dMRI sequences . . . 143
5.3 Simulation campaign results and post-processing . . . 145
5.3.1 Compressed representation of the dMRI signal attenuation . . . 149
5.3.2 From geometric and diffusion data to CSV representations . . . 151
5.3.3 Data augmentation . . . 151
5.4 Conclusion . . . 153

6 Implementing a neural network decoding the white matter microstructure 161
6.1 Machine learning algorithms . . . 163
6.1.1 Linear regression . . . 163
6.1.2 K-nearest neighbour . . . 163
6.1.3 Support-vector machine . . . 164
6.1.4 Decision tree . . . 164
6.1.5 Random Forest . . . 164
6.1.6 Extra trees . . . 164
6.2 Neural network: an overview . . . 165

6.2.1 Historical background . . . 165
6.2.2 Multi-layer perceptron architecture . . . 167
6.3 MEDUSA neural network implementation . . . 174
6.3.1 General design of the decoding pipeline . . . 174
6.3.2 Tuning of the NEDUSA white matter neural network . . . 174
6.3.3 Training and testing the networks . . . 180
6.3.4 Discussion . . . 194
6.3.5 Conclusion . . . 196

7 Conclusion 197
7.1 Contributions . . . 197
7.1.1 Optimization of MEDUSA simulator and new input user-friendly dictionary . . . 197
7.1.2 Validation of MEDUSA . . . 197
7.1.3 Launching of a simulation campaign on HPC facility . . . 197
7.1.4 Implementing a neural network decoding the white matter microstructure . . . 198
7.2 Prospects . . . 198
7.2.1 Web servibe MEDUSA . . . 198
7.2.2 Implement a new neural network architecture . . . 198
7.2.3 Sampling more geometry parameters . . . 198
7.2.4 Test more parameters from the simulation . . . 198
7.2.5 Simulation of new white matter tissues . . . 199
7.3 Conclusion . . . 199

Appendix 199

Bibliography 205

List of Figures

1 Pipeline générale de MEDUSA . . . xvii
2 Visualisation de 3 échantillons de géométrie MEDUSA présentant respectivement 1, 2 et 3 populations de fibres . . . xviii
3 Visualisation de la géométrie d'échantillons synthétisés lors de la campagne de simulation avec 9 échantillons présentant des diamètres axonaux et des fractions volumiques croissantes . . . xix
4 Deux échantillons de la campagne de simulation . . . xx

1.1 Complexity of neuronal system . . . 3
1.2 Central nervous system . . . 5
1.3 Protection of the brain . . . 6
1.4 Organisation of the brain ventricles . . . 7
1.5 Encephalon structures . . . 7
1.6 Human brain lobes . . . 9
1.7 Cortex folding . . . 11
1.8 Basal ganglia, 3D illustration . . . 12
1.9 Coronal slice of a brain's basal ganglia . . . 13
1.10 A new deep white matter fiber bundles atlas of the human brain . . . 15
1.11 Major cerebral arteries . . . 16
1.12 Scheme of a neuron cell . . . 18
1.13 Cells composing cerebral cortex . . . 21
1.14 Brodmann's areas of the cerebral cortex . . . 24
1.15 Scheme of the cytoarchitectonics layers . . . 26

2.1 Patents and diagram of MRI hardware . . . 29
2.2 History of progress in MRI brain imaging as scanners with higher field strengths were developed . . . 31
2.3 Illustration of the NMR experiment . . . 34
2.4 Diagram showing the process of the excitation-relaxation phenomenon . . . 35
2.5 MRI gradients . . . 38
2.6 Spin echo sequence and K-space filling . . . 40
2.7 Bronian motion . . . 41
2.8 Chronogram of a PGSE sequence . . . 43
2.9 Spin dephasing during the PGSE experiment . . . 44
2.10 Displacement distribution profiles for tert-butanol and water in neuronal tissue at different diffusion times . . . 45
2.11 ADC as a function of frequency for diffusion between two infinite impermeable planes 46
2.12 PGSE and OGSE sequences . . . 47

2.13 Diffusion tensor . . . 49
2.14 Displacement and probability maps from diffusion-weighted imaging . . . 51
2.15 Evolution of the q-space sampling schemes . . . 52
2.16 Funk-Radon transform . . . 53
2.17 Spherical harmonics . . . 54

3.1 Illustration of the hindered diffusion process within brain white matter tissues . . . 57
3.2 Hindered water diffusion in white matter fibers . . . 59
3.5 Approximation of bovine optic nerve structure . . . 61
3.3 Hardware phantoms of the microstructure . . . 62
3.4 Phantom comprising microcapillaries and dMRI signal attenuation as a function of q-values . . . 63
3.6 NODDI maps . . . 65
3.7 Pioneering cellular membranes simulator . . . 68
3.8 Examples of packing methods developed to create white matter numerical phantoms . . . 70
3.9 Damaged neurites simulations . . . 71
3.10 Ultra realistic generative model of WM microstructure . . . 72
3.11 Biological and chemical materials as white matter phantoms . . . 73
3.12 Cerebral tissues segmentations and related geometric simulators . . . 74
3.13 Three particle-to-membrane interaction models . . . 76
3.14 Bias of synthetic diffusion-weighted signals . . . 79
3.15 Effects of relative step-length on convergences and biases of synthesized signals . . . 79

4.1 MEDUSA overall structure . . . 83
4.2 MEDUSA overall structure . . . 85
4.3 The elementary classes composing MEDUSA geometry framework . . . 87
4.4 UML structure of the Atom classes within MEDUSA . . . 88
4.5 UML structure of classes related to cells . . . 90
4.6 Waston distribution pattern . . . 91
4.7 Illustration of the induction of tortuosity within axonal fibers . . . 92
4.8 Illustration of the induction of beading within axonal fibers . . . 93
4.9 Creation of a myelin sheath and Ranvier nodes on axonal fibers . . . 93
4.10 3D rendering of single fiber population geometries with an increasing degree of realism 94
4.11 Simplified grey matter phantoms . . . 95
4.12 Illustration of the astrocyte creation procedure . . . 96
4.13 Illustration of the oligodendrocyte creation procedure . . . 97
4.14 UML structure of classes related to the population of cells . . . 98
4.15 Overlapping spheres atoms from the same cell . . . 99
4.16 Illustration of three simulated fiber populations and their associated total repulsion force norm function over iterations . . . 101
4.17 Illustration of the lookup table in MEDUSA . . . 102
4.18 MEDUSA samples of geometries associated with distributed particles cloud . . . 103
4.19 Particle trajectories simulated using the MEDUSA diffusion process simulator . . . 105
4.20 Exemples of dMRI sequences simulated with MEDUSA . . . 107
4.21 Overview of the MEDUSA input pipeline . . . 114
4.22 Analitical validation 1 . . . 115

4.23 Analitical validation 2 . . . 116
4.24 2d illustrations of Monte-Carlo simulations . . . 118
4.25 Heat map of the mean RMAE from simulated dMRI signals compared to analytical solutions . . . 119
4.26 Random leap RSL validation results . . . 121
4.27 Radom leap RSL computational time results . . . 122
4.28 Elastic bouncing RSL validation results . . . 123
4.29 Elastic bouncing RSL computational time results . . . 124
4.30 FOV validation 1 . . . 127
4.31 Illustration of the worst-case scenario during a "Remove overlap" procedure . . . 128
4.32 Illustration of the worst case scenario during an iteration stem of the Monte-Carlo procedure . . . 129
4.33 FOV test 2 . . . 130
4.34 Global computational time compared with geometry computational time . . . 131
4.35 Remove overlap validation results . . . 133

5.1 Joliot-Curie supercomputer . . . 135
5.2 TGCC's partitions overall characterization . . . 138
5.3 Overall simulation campaign diagram . . . 139
5.4 3D rendering of white matter samples . . . 144
5.5 Simulation campaign results: volume fraction and axon diameter . . . 146
5.6 Simulation campaign results: global angular dispersion . . . 147
5.7 Simulation campaign results: 3 fiber population with different volume fraction . . . 148
5.8 Overall diagram of the post-processing pipeline . . . 150
5.9 Illustration of the rotation algorithm . . . 152
5.10 Histograms of geometry parameters from simulations gathering one simulated fiber population . . . 154
5.11 Histograms of diffusion parameters from simulations gathering one simulated fiber population . . . 155
5.12 Histograms of geometry parameters from simulations gathering two simulated fiber populations . . . 156
5.13 Histograms of diffusion parameters from simulations gathering two simulated fiber populations . . . 157
5.14 Histograms of geometry parameters from simulations gathering three simulated fiber populations . . . 158
5.15 Histograms of diffusion parameters from simulations gathering three simulated fiber populations . . . 159

6.1 Artificially generated pictures depicting a multilayer perceptron neural network. These images have been generated with the online AI tool "dall-e 3" [openai 2024]. . . . 161
6.2 Illustration of the principle of important machine learning algorithms for supervised learning of tabular data . . . 165
6.3 Overall structure of a multilayer perception . . . 168
6.4 Effect of bias and weight on activation function . . . 169
6.5 Illustration of overfitting . . . 172
6.6 Layout of parameters for different neural networks . . . 175

6.7 Illustration of the overall decoding pipeline 176
6.8 ReLU plot 179
6.9 Models learning results number 1 181
6.10 Models learning results number 2 182
6.11 2D Illustration of a two fiber populations simulated sample with it means orientation vector in red re-oriented 185
6.12 Models learning results number 3 186
6.13 Prediction point cloud number 1 189
6.14 Prediction point cloud number 3 190
6.15 Prediction point cloud number 5 191
6.16 Features importance plot number 193

List of Tables

4.1 DMRI sequences settings for the simulation campaign, adapted from the settings from iCORTEX protocol. 119
4.2 Main features of the geometries simulated during the test campaigns 120

5.1 Tuning of the microstructural parameters 142
5.2 dMRI sequences settings for the simulation campaign 145

6.1 Detailed prediction performances 183
6.2 Detailed prediction performances after applying rotation 187

Acronyms

AD Axon diameter.

CFS Cerebro spinal fluid.

CNS Central nervous system.

DMRI Diffusion magnetic resonance imaging.

DTI Diffusion tensor imaging.

ERL Equal step length random leap.

FOV Field of view.

GAD Global Angular Dispersion.

HPC High-performance computing.

iCORTEX In vivo mapping of the CORTex cytoarchitecture, myeloarchitecture of brain functional networks using EXtreme scale analytics and EXtreme fields.

MC Monte-Carlo.

MEDUSA Microstructure Environment Designer with Unified Sphere Atoms.

MRI Magnetic resonance imaging.

NMR Nuclear magnetic resonance.

ODF Orientation distribution function.

OOP Object-oriented programming.

RF Radio frequency.

RSL Relative step lenght.

VF Volume Fraction.

Résumé de la thèse

L'imagerie médicale est une discipline dont les applications cliniques ou pré-cliniques sont limitées par des contraintes telles que la résolution spatiale et temporelle, le contraste, le coût ou le caractère invasif pour le patient. Il existe différentes techniques d'imagerie telles que l'échographie, la radiographie ou l'électroencéphalographie, chacune présentant des avantages spécifiques par rapport aux contraintes précédemment énoncées et idéale pour détecter des pathologies spécifiques ou explorer certaines parties du corps. Depuis sa commercialisation à la fin des années 1980, l'Imagerie par Résonance Magnétique (IRM) a prouvé son efficacité en réduisant le caractère invasif de l'imagerie chez les patients tout en permettant une exploration en profondeur des structures cérébrales. De plus, la modalité d'imagerie IRM dite de diffusion (IRMd), consacrée à la mesure d'un signal sensible au mouvement des molécules d'eau, a permis l'exploration de la microstructure du cerveau. Le défi que représente l'étude de la microstructure des tissus cérébraux constitue une nouvelle frontière scientifique à repousser afin d'étayer notre compréhension du cerveau et de doter les médecins de nouveaux outils améliorant leurs diagnostics.

C'est dans ce contexte que l'équipe Ginkgo (Microstructure) du laboratoire BAOBAB/GAIA de NeuroSpin a développé un environnement appelé MEDUSA (Microstructure Environment Designer Using Sphere Atoms), permettant la synthèse de tissus virtuels réalistes imitant l'environnement cellulaire cérébral humain à l'échelle microscopique tout en y simulant le mouvement Brownien de l'eau lié au processus de diffusion afin d'acquérir l'atténuation du signal IRMd qui en résulte. L'apport des techniques d'intelligence artificielle, notamment de deep learning, ouvre la voie au développement de méthodes d'imagerie in vivo des tissus de la substance blanche cérébrale exploitant l'information contenue dans le mouvement Brownien des molécules d'eau présentes dans le cerveau dont les trajectoires portent une empreinte de la microstructure locale des tissus.

Le sujet de thèse proposé vise à mettre en place, grâce à l'environnement logiciel Ginkgo/MEDUSA, des campagnes de simulation sur le "Très Grand Centre de Calcul" (TGCC du CEA, Bruyère-le-Châtel), afin de dresser une base de données regroupant un grande nombres d'échantillons caractérisés par différentes géométries cellulaires associées à leurs empreintes en IRMd. Cette base de donnée sera cruciale pour développer un modèle computationnel robuste de décodage de la microsctructure axonale au sein de la matière blanche cérébrale. Cette thèse bénéficie du soutien financier du consortium AIDAS entre le "Forschungszentrum Jülich" (FZJ) allemand et le "Commissariat à l'énergie atomique" (CEA) français.

Les paragraphes suivant visent à établir les différentes réflexions et les successifs développements ayant mené à la mise en place d'un modèle computationel de décodage de la microstructure de la matière blanche cérébrale.

Le cerveau humain comme terrain de jeu

Le cerveau est considéré comme l'une des structures les plus complexes du corps humain, en conséquent il est fondamental de l'étudier en détail sur le plan anatomique avant de développer des fantômes numériques réalistes de la microstructure de la matière blanche. Deux échelles d'étude sont à dissocier: l'échelle macroscopique et microscopique.

Plusieurs critères permettent de séparer l'espace macroscopique au sein des deux hémisphères cérébraux. On peut ainsi citer des critères morphologiques tels que les plissures (sillons) et lobes (gyri) cérébraux ou des critères fonctionnels répartissant les différentes fonctions cognitives à certaines régions du cerveau. On peut également distinguer les régions en fonction de la dichotomie matière grise / matière blanche, caractérisant respectivement les régions regroupant les corps cellulaires des neurones (ou soma) d'un part et leurs prolongements appelés axones (ou fibres nerveuses) d'autre part. La matière blanche porte ce nom du fait de la gaine de myéline blanche entourant les axones.

Concernant l'échelle microscopique, la cyto-architectonie est une approche histologique centrée sur l'observation de l'arrangement caractéristique des cellules dans les différentes couches du cortex cérébral. La myélo-architectonie quant à elle étudie l'arrangement des axones qui connectent les neurones les uns aux autres. L'étude plus fine du cerveau à l'échelle microscopique permet une caractérisation de la géométrie des axones et des cellules gliales (astrocytes, oligodendrocytes) de la matière blanche, confortant la configuration des paramètres de notre simulateur. Plus précisément dans le cas de la matière blanche, les diamètres axonaux mesurent entre 0.1µm et 9.0µm, avec une plus grande proportion de diamètre inférieur à 1.0µm, les fractions volumiques observées sont de 60 à 80% pour les axones, 15 % pour les cellules gliales, 20% pour l'espace extracellulaire et environ 5% pour les vaisseaux sanguins. De plus, on observe que 60% des échantillons histologiques présentent des populations de fibres nerveuses se croisant.

L'imagerie par résonance magnétique

L'imagerie par résonance magnétique (IRM) repose sur la résonance magnétique nucléaire (RMN) des noyaux des molécules d'eau (protons ou spins) soumis à des impulsions électromagnétiques à la fréquence de Larmor. L'eau étant abondante au sein des tissus cérébraux et du fluide cérébrospinal, la technologie IRM est particulièrement adaptée à l'exploration non-invasive du cerveau humain.

L'IRM se distingue de la RMN par la localisation du signal issu de la résonance des protons grâce à l'usage de multiples gradients électromagnétiques. Le signal IRM peut être pondéré suivant différentes modalités (anatomique, quantitative, fonctionnelle) dont celle dite de diffusion (IRMd). Le signal issu d'une acquisition en IRMd est sensible au mouvement Brownien de l'eau dont les trajectoires restreintes portent l'empreinte de la microstructure locale du tissu.

Modèles biophysiques pour le cerveau

Différents modèles analytiques et computationnels visent à décoder la microstructure des tissus cérébraux à l'aide de leur empreinte IRMd. L'approche analytique repose sur une représentation simplifiée des tissus par des formes géométriques simples, permettant la résolution de l'équation de la diffusion. L'approche computationnelle vise à combler la simplicité des modèles analytiques en faisant appel à des algorithmes d'apprentissage automatique nourris d'échantillons simulés regroupant d'une part la paramétrisation de la géométrie des cellules et d'autres part son empreinte en IRMd. La création d'une base de données regroupant les échantillons simulés nécessite le développement

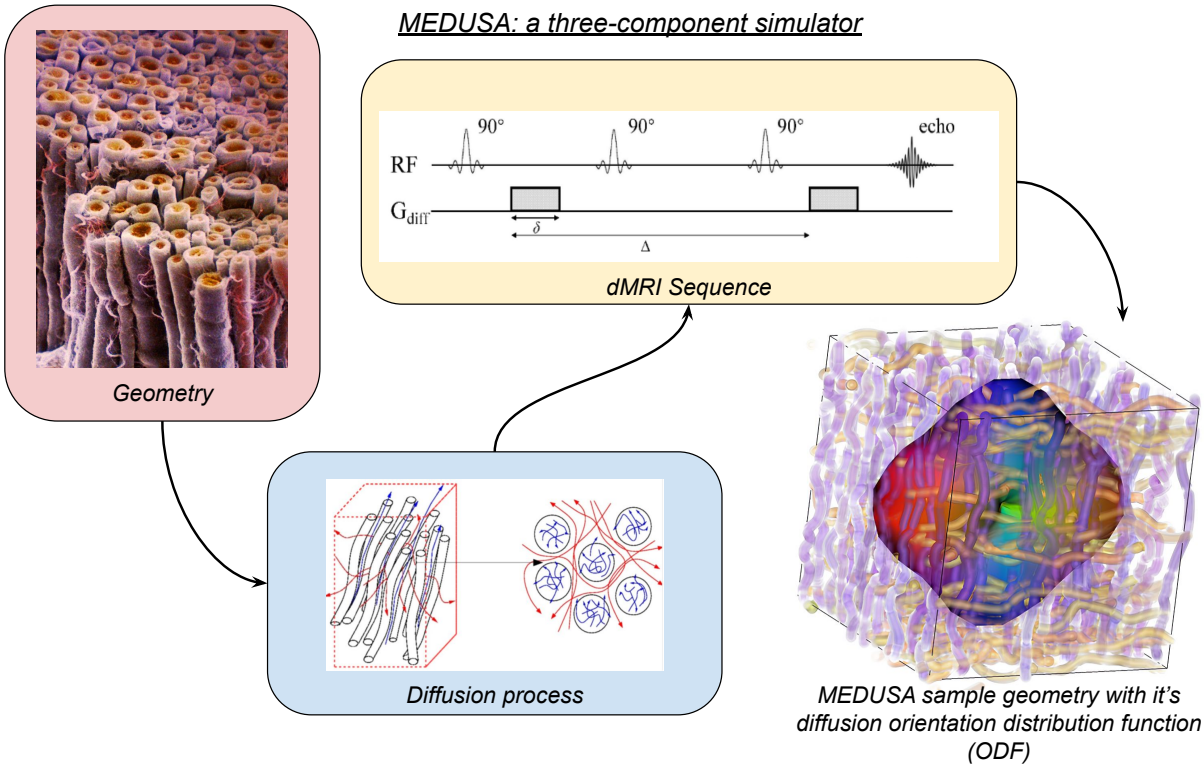


Figure 1: *Pipeline générale de MEDUSA, regroupant les simulations de géométries cellulaires, de diffusion et de synthèse du signal IRMd.*

d'un simulateur combinant trois composantes: la synthèse de l'environnement cellulaire à l'échelle microscopique, le mouvement Brownien de l'eau et l'empreinte IRMd résiduelle.

Concepteur d'environnement de microstructure avec simulateur d'atomes de sphères unifiées (MEDUSA)

Le simulateur MEDUSA est un simulateur regroupant les trois composantes (géométrie, diffusion, signal IRMd) nécessaire à l'élaboration d'un dictionnaire d'échantillons de la microstructure de la matière blanche associé à son empreinte IRMd. MEDUSA bénéficie d'une approche unique pour générer les géométries cellulaires reposant sur l'utilisation d'une décomposition des cellules individuelles en motifs simples appelés "atomes". L'usage d'atomes, allié à l'optimisation computationnelle de son environnement logiciel (Ginkgo framework), permet à MEDUSA de simuler des échantillons uniques et ultra réalistes de la microstructure de la matière blanche cérébrale dans des temps réduits. En effet, on note que la simulation de la géométrie d'un échantillon de trois populations de fibres, pour un champ de vue de $30\mu m^3$, un diamètre axonal distribué autour de $1\mu m$ et une fraction volumique totale avoisinant les 80%, le temps estimé est en moyenne de 14min pour une simulation effectuée sur une station avec 32 CPU parallélisés.

MEDUSA simule également le mouvement Brownien de l'eau via son simulateur de processus de Monte-Carlo déplaçant itérativement et aléatoirement des millions de particules d'eau au sein de la géométrie synthétisée. Enfin, MEDUSA intègre le signal issu de l'empreinte IRMd des trajectoires des particules d'eau, celui ci étant pondéré en fonction des séquence IRMd sélectionnées et réglables en amont de la simulation.

Une batterie de tests fut menée afin de valider la justesse des résultats émis par le simulateur MEDUSA. Ces tests permirent également d'établir le réglage optimal des différents paramètres de la simulation

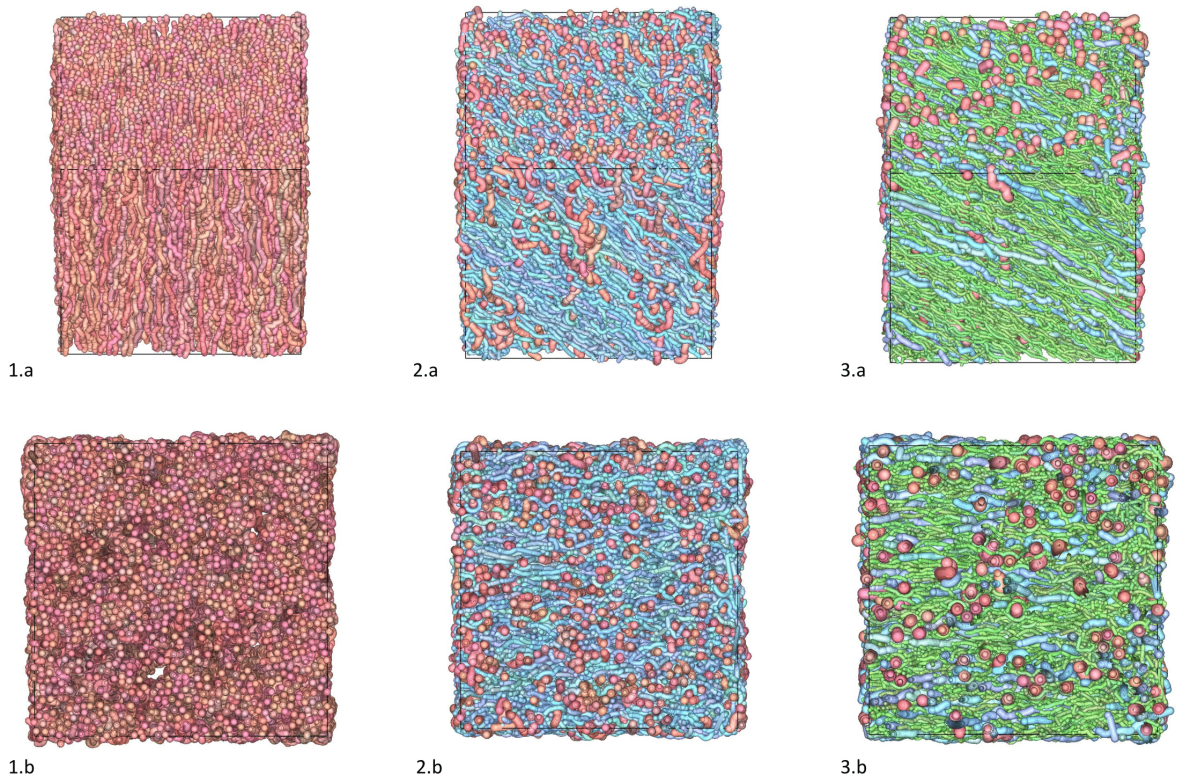


Figure 2: *Visualisation de 3 échantillons de géométrie MEDUSA présentant respectivement 1, 2 et 3 populations de fibres en vue de face (haut) et de dessus (bas).*

tant en terme de réalisme du processus de diffusion simulé qu'en terme d'efficacité computationnelle. Riche des enseignements de nos campagnes de tests, MEDUSA fut calibré et prêt à être employé dans le cadre d'une large campagne de simulation.

Mener une vaste campagne de simulation MEDUSA

Cette thèse vise à proposer un modèle informatique pour décoder la microstructure de la matière blanche en utilisant un algorithme d'apprentissage automatique entraîné sur des échantillons simulés. Pour alimenter correctement l'algorithme d'apprentissage automatique, un nombre significatif d'échantillons doit être simulé; cet objectif est facilité par l'efficacité informatique du simulateur MEDUSA, qui permet de générer des échantillons réalistes dans des temps de calcul réduits. Dès le début, le choix a été fait d'exécuter la campagne de simulation sur une installation de calcul à haute performance (HPC), car le nombre prévu d'échantillons simulés doit être suffisamment important pour prendre en compte toutes les configurations plausibles de la microstructure de la matière blanche.

Le Très Grand Centre de Calcul (TGCC, CEA, DAM Île de France, Bruyères-le-Châtel) a accueilli les différentes campagnes de simulation MEDUSA ayant mené à la synthèse d'environ 40 000 échantillons. Chaque échantillon est unique et contient une à trois populations de fibres axonales en son sein. Un algorithme de bruitage et de rotation a permis d'étoffer notre base de données en la faisant passer de 40 000 à 4 000 000 d'échantillons.

Concevoir un réseau de neurones décodant la microstructure de la matière blanche cérébrale

L'algorithme d'apprentissage automatique élaboré lors de cette thèse repose sur différents réseaux

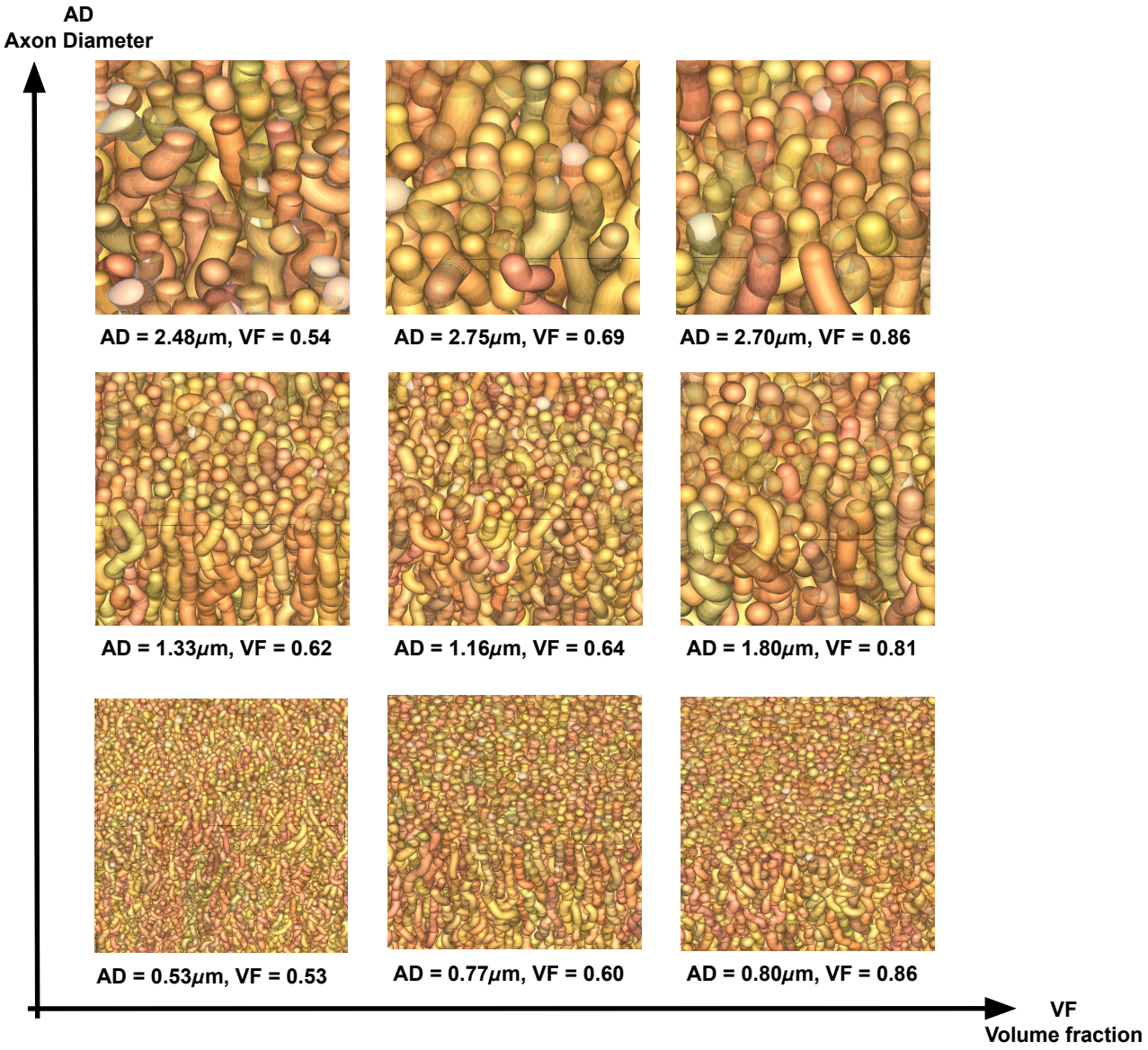


Figure 3: *Visualisation de la géométrie d'échantillons synthétisés lors de la campagne de simulation avec 9 échantillons présentant des diamètres axonaux et des fractions volumiques croissantes. Les fractions volumiques en abscisses sont compris entre 0.5 et 0.9 et les diamètres axonaux en ordonnée entre 0.5μm et 3μm.*

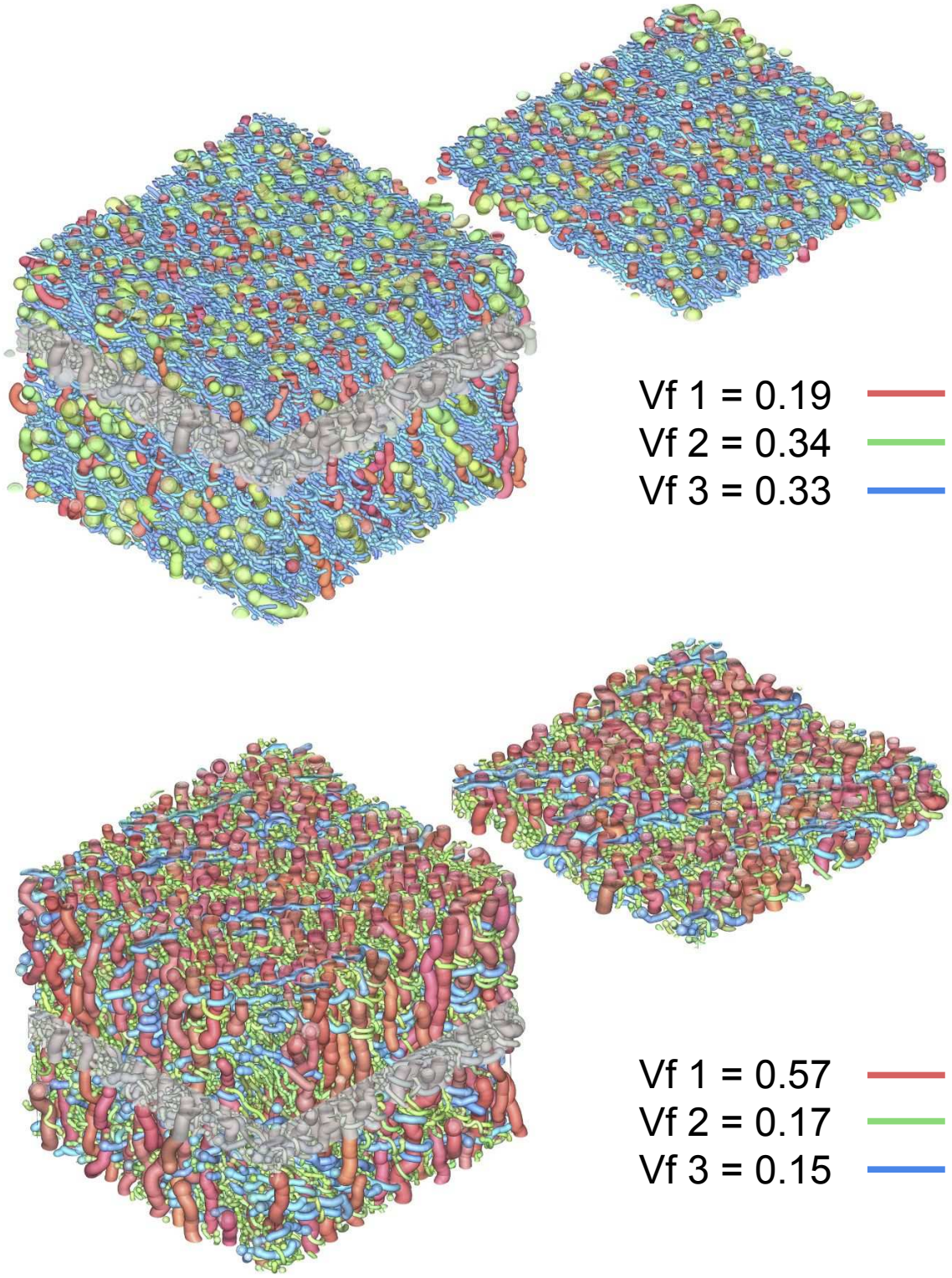


Figure 4: *Deux échantillons de la campagne de simulation sont présentés. Ils comprennent tous deux 3 populations de fibres populations de fibres. Les populations sont différenciées par leur couleur RVB spécifique.*

de neurones conçus et optimisés afin de répondre au problème ciblé par cette thèse : le décodage de la microstructure de la matière blanche. Ce problème consiste à associer une empreinte IRMd à la géométrie issue de la microstructure, par conséquent la fonction que l'algorithme doit apprendre est une régression (supervisée car les données sont étiquetées).

Les paramètres génératifs de la microstructure décodés par le réseau sont la distribution des diamètres axonaux des populations de fibres, leurs fractions volumiques, leurs orientations ainsi que leurs dispersions angulaires. De plus, du fait que les échantillons simulés se composent de une à trois populations de fibres, un paramètre supplémentaire comptabilisant le nombre de populations est saisi par un réseau spécifique dont le résultat permet d'aiguiller le décodage. Ainsi, 4 réseaux de neurones différents ont été conçus et optimisés: l'un comptabilisant le nombre de populations et les trois autres respectivement dédiés à décoder la microstructure d'échantillons contenant une, deux ou trois populations.

Suite à l'apprentissage des réseaux de neurones basé sur la base de données d'entraînement MEDUSA, une étude approfondit de l'impact des différents paramètres constituant la base de donnée fut menée afin d'estimer la qualité des prédictions du modèles. Les résultats furent prometteurs dans le cas d'échantillons constitué d'une seule population de fibres, en conséquent, une application direct des modèles prédictifs MEDUSA est à l'avenir souhaitée sur de véritables acquisitions issu du protocole iCORTEX et centrée sur le corps calleux.

Apports de la thèse, futurs développements et conclusion

Le sujet présenté par cette thèse fait appel à de nombreux domaines scientifiques: anatomie du cerveau, physique de l'IRM, simulateurs informatiques, gestion de base de données, calcul sur super calculateur et conception de réseaux de neurones. De nombreux développements ont été portés tels que l'optimisation du simulateur MEDUSA afin de le rendre transposable à un calcul sur super-calculateur et d'améliorer l'instanciation de ces paramètres d'entrées. De plus, un dictionnaire d'échantillons digitaux de géométrie microscopique de la matière blanche associés à leur empreinte en IRMd a été constitué. Enfin, une pipeline de décodage de la microstructure de la matière blanche a été conçue s'appuyant sur de multiple réseaux de neurones entraînés.

A l'avenir, de nouvelles campagnes de simulations MEDUSA devront être menées comportant des géométries plus complexes à étudier via la présence de cellules gliales ou de micro vaisseaux. Une amélioration de la pipeline de décodage sera nécessaire pour cartographier plus fidèlement les paramètres de la microstructure.

Pour conclure, l'approche computationnelle d'un modèle de décodage de la microstructure de la matière blanche, appuyé d'échantillons synthétisés par simulateur, est prometteuse et constitue une alternative aux modèles analytiques. Les travaux présentés ont de nombreuses applications pratiques chez le patient via notamment la constitution d'un outil de diagnostique supplémentaire pour les médecins. Enfin, le sujet de cette thèse est porteur et multi-disciplinaire, ouvrant la voie à de nombreux développements et réflexions futures, facilités par la flexibilité et la portabilité de l'environnement logiciel Ginkgo MEDUSA.

General Introduction

Context and motivations

Medical imaging is a discipline in which the quality and feasibility of clinical or pre-clinical applications are constrained by boundaries such as spatial or temporal resolution, contrast capacity, cost of implementation, or invasiveness for the patient. Various imaging techniques such as echography, radiography, or electroencephalography exist, each with specific benefits regarding the previous constraints and ideal for detecting specific pathologies or exploring particular body parts. Since its commercial democratization at the end of the 80s, Magnetic Resonance Imaging (MRI) has proved its effectiveness in reducing the invasiveness of imaging in patients while enabling in-depth exploration of the brain's structures metabolism, and functional network. Part of it, diffusion MRI, based on the measurement of a signal sensitive to the movement of water molecules, has made it possible to explore the brain's microstructure, which is at the heart of this thesis. Quantifying the brain tissue microarchitecture with a high degree of accuracy remains a real challenge that could open an overcome to the creation of disruptive tools to understand better the impairments occurring at cellular scales aiming the development of brain diseases and thus providing immersive tools for the diagnosis.

It is in this context that the Ginkgo (Microstructure) team of NeuroSpin's BAOBAB/GAIA laboratory is developing an environment called MEDUSA (Microstructure Environment Designer Using Sphere Atoms), which allows the creation of realistic virtual tissues representative of the cellular environments that can be found in the human brain, and therefore allows to simulate the Brownian motion of water related to the diffusion process and thus to predict the MRI signal that would be obtained for each virtual tissue using diffusion-weighted MRI. The significant contribution of artificial intelligence techniques, particularly for supervised learning of large data sets, opens new perspectives for the development of in vivo imaging methods of brain white matter tissues since it becomes possible to exploit via deep learning methods the microscopic information embedded in the Brownian motion of water molecules present in the brain, whose trajectories embed an imprint of the local microstructure of the tissue.

This thesis aims at launching (using Ginkgo/MEDUSA software platform) large simulation campaigns on the CEA HPC facility (Très grand centre de calcul, CEA DAM Ile de France, Bruyères-le-Châtel) to develop a novel computational model for the decoding of brain white matter microstructure compatible with clinical use, aiming in the long term to provide neuroradiologists with a new monitoring tool.

This project benefited from the financial support of the AIDAS joint institute between the German "Forschungszentrum Jülich" (FZJ) and the French "Commissariat à l'énergie atomique" (CEA) and also benefited from a strong collaboration between the group of Pr. Markus Axer (INM-1, FZJ), and the Ginkgo team (Baobab, Neurospin, CEA).

Thesis organization

The structure of this thesis seeks to correspond to the path of thoughts and works achieved during these past 4 years. Thus, the first three chapters deal in detail with the state of the art necessary for a proper understanding of the subject, with [chapter 1] presenting the human brain's anatomic structures, with a distinction between gross anatomy and microscopic anatomy. [Chapter 2] focuses on presenting the fundamentals of magnetic resonance imaging, focusing on diffusion MRI modality at the heart of this thesis. Completing the chapters dedicated to the state-of-the-art, [chapter 3] presents the existing analytical models to decode the brain microstructure based on its dMRI signature and lists the different features of digital simulators developed by the dMRI community aiming to synthesize the microstructure geometry and its dMRI footprint to train machine learning algorithm for computational decoding models. [Chapter 4] introduce the functioning of the MEDUSA simulator, developed by the Ginkgo team and designed as an all-in-one simulator enabling the synthesis of virtual brain tissues, diffusion process, and dMRI signature. The MEDUSA framework was used to build a large dictionary of simulated realistic numerical phantoms of white matter brain tissues associated with their synthesized dMRI signature during a simulation campaign whose results and features are presented in [chapter 5]. [Chapter 6] is dedicated to the presentation of the decoding pipeline based on deep learning machine learning algorithms purposely designed to predict white matter microstructure axonal features such as axonal diameter, volume fraction, global angular orientation, or fiber mean orientations based on the dMRI features collected during the simulation campaign. Finally, [chapter 7] is dedicated to the presentations of the ongoing works regarding MEDUSA and future potential applications based on the presented achievements as well as a general conclusion of the thesis.

Chapter 1

Human brain anatomy

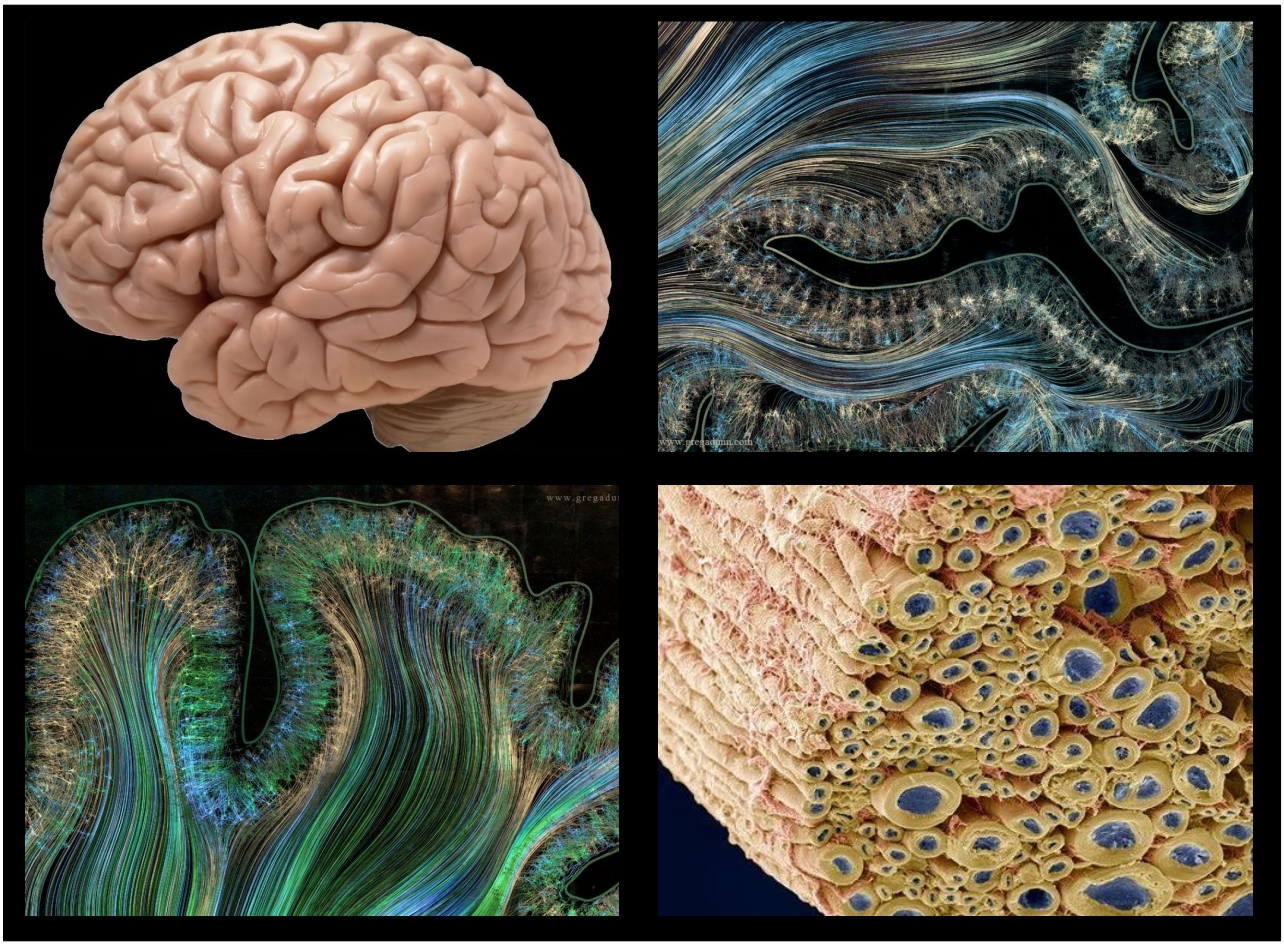


Figure 1.1: *Complexity of neuronal system, adapted from [Dunn 2017] and Greg Gun micro engraving technique and Steve Gschmeissner artistic axon microscopic photos [Gschmeissner 2024].*

Chapter outline

1.1 Macroscopic anatomy 4
1.1.1 Gross anatomy 4
1.1.2 Brain lobes 8
1.1.3 Grey matter vs white matter 10
1.1.4 Brain vasculature 14
1.2 Microscopic anatomy 16
1.2.1 Brain cells 16
1.2.2 Cortex histology 22
1.2.3 White matter histology 25
1.2.4 Conclusion 27

Frequently referred to as one of the most complex structures of the human body, the brain will be our playground throughout this thesis. Its structure must be understood to interpret and decode the MRI signal from it or develop realistic numerical phantoms of its tissues. With this in mind, the purpose of this chapter is to present the anatomic background of the brain on which this thesis relies, from macroscopic to microscopic scale.

1.1 Macroscopic anatomy

1.1.1 Gross anatomy

Central nervous system - The human central nervous system (CNS) comprises the spinal cord and brain also known as encephalon [figure 1.2].

The spinal cord is a 45 cm long cylinder flattened from front to back with a 1-2cm diameter. From an axial perspective, two areas can be spotted: white matter around the perimeter, which contains the axons of sensory/motor neurons, and grey matter in the center, shaped like a butterfly, comprised of the cell bodies of neurons. It should be noted that this layout is opposite to the one in the encephalon because, in the case of the spinal cord, it allows the innervation of the 31 pairs of spinal nerves along the spinal cord that emanate from the white matter. It has two bulges corresponding to the roots that innervate the upper and lower limbs.

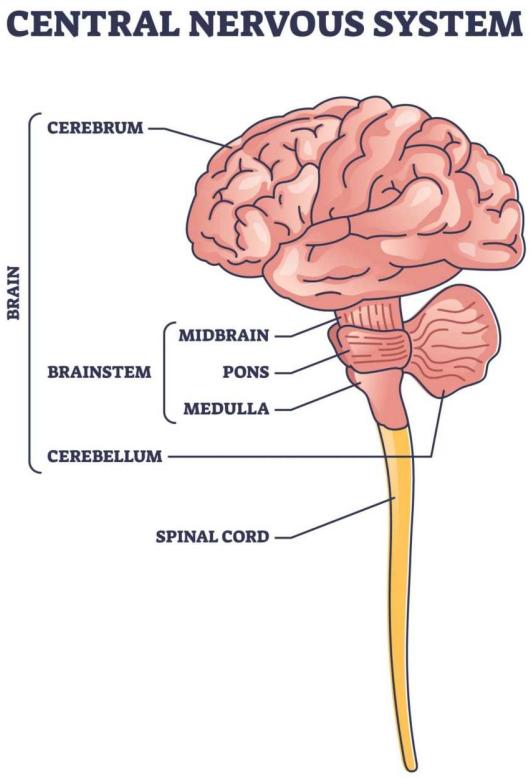


Figure 1.2: *Central nervous system separated into two parts: the spinal cord and the brain, also called encephalon. The encephalon is composed of the cerebrum, the cerebellum, and the brainstem. The brainstem comprises the midbrain, the pons, and the medulla. Adapted from [ShutterStock 2024].*

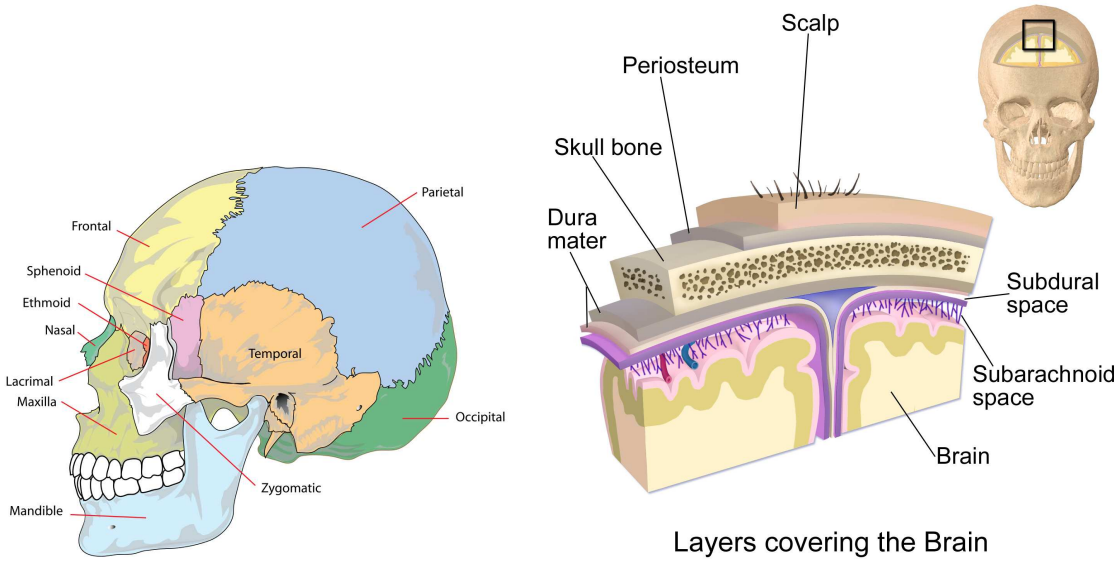
In the following sections, we will focus on the description of the encephalon.

Protection of the encephalon - The eight protective bones of the skull [figure 1.3] protects the encephalon: one frontal bone, two parietal bones, two temporal bones, one occipital bone, one ethmoid bone, and one sphenoid bone. Three protective membranes below the skull [figure 1.3] called meninges, cover both the encephalon and spinal cord:

- The dura mater - Thick, white layer that delimits expansions via the brain scythe and cerebellum tent, which divide the intracranial volume. The outer layer adheres to the inside of the bone and is highly vascularized and innervated. The scythe of the brain insinuates itself between the two hemispheres. The dura mater wraps the arachnoid membrane.

- The arachnoid - Named for its spiderweb-like appearance, located between the pia mater and the dura mater is a thin, non-vascularized, and non-innervated layer that adheres to the inner surface of the dura mater; it's separated from the pia mater by the subarachnoid space containing the cerebrospinal fluid (CFS).

- The pia mater - Thin, transparent, fragile, innervated, and non-vascularised membrane. The Pia mater accompanies the arteries which penetrate the parenchyma.



(a) *Human skull and bones.* (b) *Meninges scheme and organization.*

Figure 1.3: *Protection of the brain. Adapted from [Healthliteracyhub 2024] and [Wikimedia 2020].*

Ventricular structures - The encephalon is structured around four cavities called ventricles filled with CFS, allowing the inner nutrition of the brain. The choroid plexus, a set of cells hosted inside the ventricles, secrete the CSF:

- 1st and 2nd ventricles (lateral) - Symmetric and paired, they supply the brain hemispheres with CSF and follow the inner structure of the brain thanks to their anterior temporal/occipital horns.
- 3rd ventricle - Supplies the base of the brain in the diencephalon.
- 4th ventricle - Supplies the cerebellum.

These ventricles are interconnected: the first and second ventricles are linked to the third ventricle by the inter-ventricular foramen and between the third and fourth ventricles by the aqueduct of Sylvius [figure 1.4]. The CSF ensures mechanical and immunological protection of the brain as well as its flotation of it. It participates in homeostasis to regulate the distribution of substances between brain cells; the glymphatic system allows brain tissues to eliminate biological waste [Bacyinski et al. 2017]. The volume of CSF is about 125mL at any time in the brain and about 250mL is generated daily.

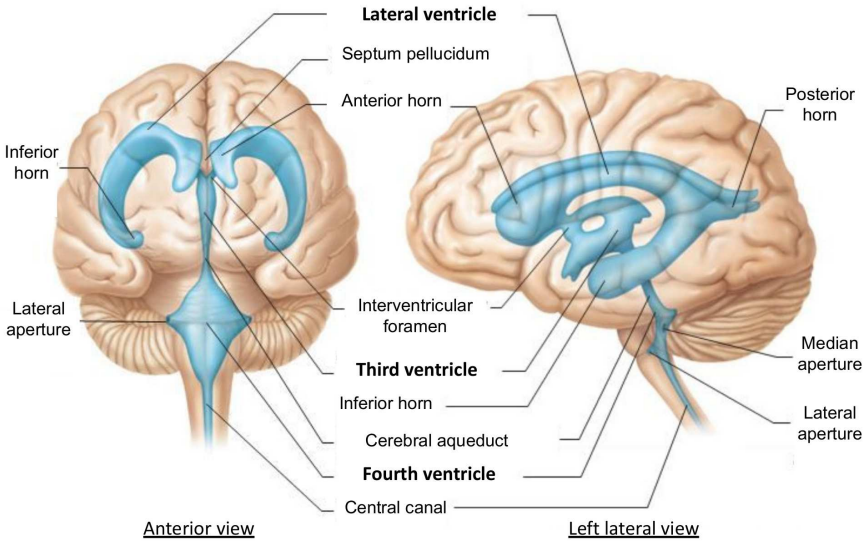


Figure 1.4: *Organisation of the brain ventricles, frontal and lateral views, adapted from [Quizlet 2024].*

Encephalon - The encephalon is composed of the cerebrum, cerebellum, and brainstem [figure 1.5]. The brainstem is located between the bottom of the spinal cord and the cerebrum at the top and is composed from the bottom to the top of the medulla oblongata, the pontella, and the mesencephalon. It is crossed by the major ascending (sensitivity) and descending (motor) pathways. Its ependymal cavity expands to form the fourth ventricle.

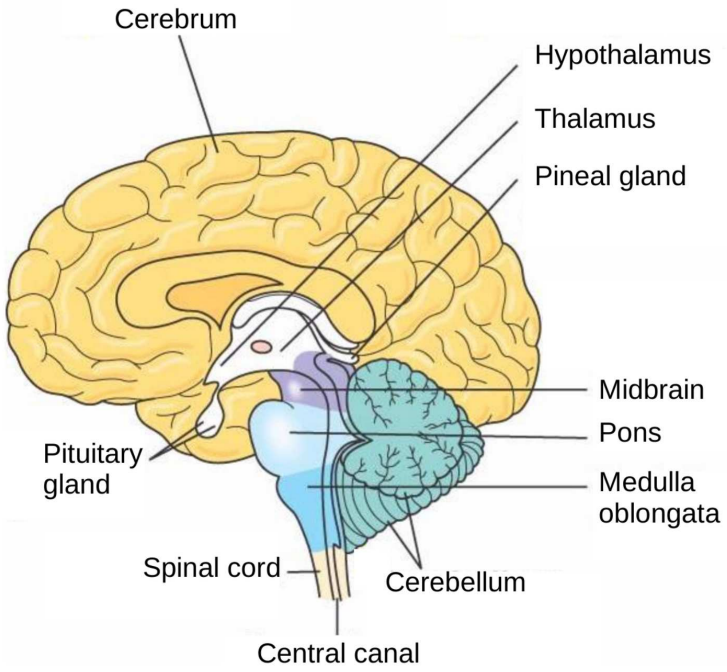


Figure 1.5: *The encephalon can be divided into three main structures: 1. the cerebrum is the biggest part of the brain, made of two hemispheres; 2. the brainstem consists of the midbrain, the pons, and the medulla oblongata. It connects the encephalon to the spinal cord. 3. the cerebellum is located in the posterior cranial fossa and is separated from the overlying cerebrum by a layer of dura mater, the tentorium cerebelli. Figure adapted from [Staff 2014].*

The cerebellum (Latin for little brain) is smaller than the cerebrum and plays a vital role in motor control. Anatomically, it appears to be a separate structure attached to the bottom of the brain,

tucked underneath the cerebral hemispheres. Its cortical surface is covered with finely spaced parallel grooves, in striking contrast to the broad, irregular convolutions of the cerebral cortex. These parallel grooves conceal that the cerebellar cortex is a continuous thin layer of tissue tightly folded in an accordion style.

We will describe specifically the anatomy of the cerebrum in the following sections.

1.1.2 Brain lobes

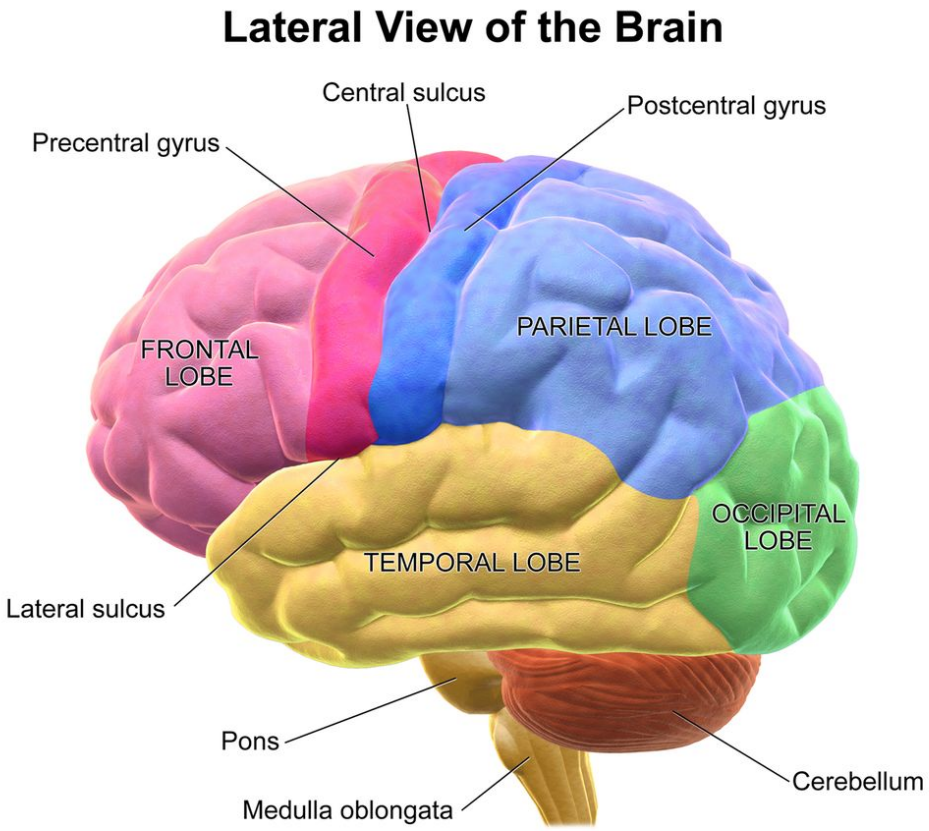
We can take different approaches to divide the areas inside the cerebrum. One of them is to distinguish its lobes. The cerebrum's outer layer of neural tissue, the cerebral cortex, is a wrinkled structure built on ridges (gyri) and grooves (sulci). The foldings create a greater surface area in the confined volume of the skull and are essential for brain wiring and functional organization. These wrinkles can be regrouped by lobes and associated with different functions. Each lobe can be found in both the left and right hemispheres of the brain:

- The frontal lobe - Located in the anterior part of the brain, it extends back to a fissure known as the central sulcus. The functions linked to this area are motor control, reasoning, emotion, and language. It contains the prefrontal cortex, responsible for higher-level cognitive functions such as reasoning skills and decision-making; the premotor and primary motor cortices, which are involved in planning and coordinating movements; and Broca's area, involved in speech production.

- The parietal lobe - Processing all sensory information, it is located behind the frontal lobe. It contains the somatosensory cortex, which surface encodes the sensory information coming from the different parts of the body. The encoding on the surface of the somatosensory cortex follows the proximity of anatomical features.

- The occipital lobe - It is almost exclusively devoted to visual processes such as visual interpretation, object and facial recognition, and distance perception. It is located at the back of the brain and contains the primary visual cortex, which is responsible for interpreting incoming visual information from the retina. The occipital cortex is organized retinotopically, which means there is a close relationship between the position of an object in a person's visual field and that object's representation in the cortex.

- The temporal lobe - Located on the flank of the head this area is associated with hearing, memory, emotion, and some aspects of language. This lobe contains the auditory cortex that processes auditory information, called Wernicke's area, which is crucial for speech comprehension.



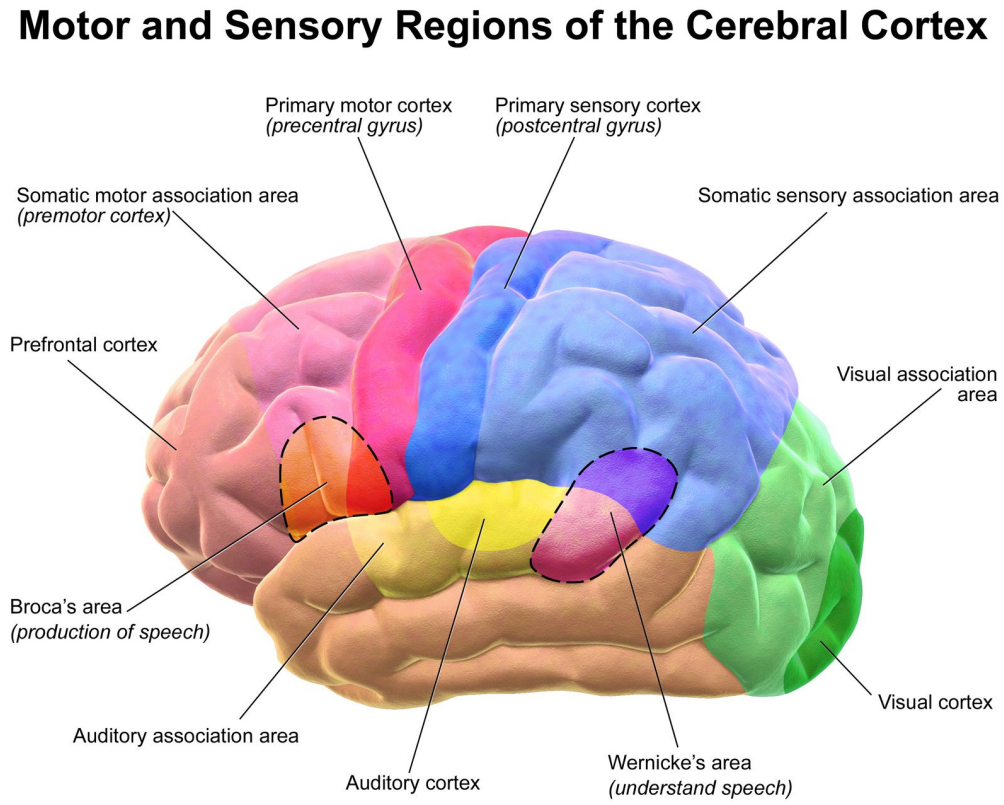


Figure 1.6: **a.** *Lateral view of the four external lobes of the human brain: frontal, parietal, temporal, and occipital. The central and lateral sulci and the precentral and postcentral gyri are also shown.* ***b.*** *Main functional areas of the brain. Adapted from [Staff 2014].*

To be more specific, two other structures can be considered as lobes:

- The limbic lobe - Subdivided into two limbic lobes: the superior limbic lobe (lobus limbicus superioris) and the inferior limbic lobe (lobus hippocampi or lobus limbicus inferioris), it is a complex set of brain structures and an arc-shaped brain region that is crucial in the process of memory, learning, motivation, and emotion, as well as endocrine functions and some autonomic—automatic, unconscious—bodily functions. It is affected in many neurodegenerative and neuropsychiatric diseases, including schizophrenia, Alzheimer's disease, and some forms of epilepsy [Gupta 2017].

- The insular cortex - It is a cortical region hidden in a lateral view by the temporal and frontal parietal lobes. It is believed to be involved in consciousness and taste analysis.

1.1.3 Grey matter vs white matter

Another way to separate areas within the cerebellum and all the CNS is through the white and gray matter dichotomy. This designation results from the fact that grey matter visually appears darker than the rest of the nervous tissue [figure **??**], which is essentially made up of bundles of axonal fibers sheathed in whitish myelin and so known as white matter. This dichotomy is not perfectly accurate because the switch from gray to white matter reveals a transition zone sharing both the properties of gray and white matter.

1.1.3.1 Grey matter

Grey matter is essentially composed of neuronal cell bodies and dendritic trees as well as certain glial cells (see 1.2.1). In the cerebrum, it is found in the outer layers of the cortex and subcortical regions. Grey matter can also be found in the cerebellum and the brainstem, but this will not be described in this section.

Cortex - Located at the periphery of the brain, the grey matter is said to define a cortex. It comprises a non-uniform stack of layers of nerve cells differentiated by cytoarchitectonic type, cell density, and connectivity. Its thickness varies according to the location; indeed [Amunts and Zilles 2015] reported a thickness of 4.2-5.7mm in the primary motor cortex, while the primary somatosensory cortex depicts a thickness of only 2.4-2.7mm. Anatomically, the cortex is folded to extend at its surface while fitting within the skull. This folding process occurs during neurogenesis and is driven by neuroepithelial cells, forming the wall of the closed neural tube in early embryonic development [figure1.7]. This implies that humans are a gyrencephalon species, contrary to lissencephalon species such as rats or mice with smooth cortex.

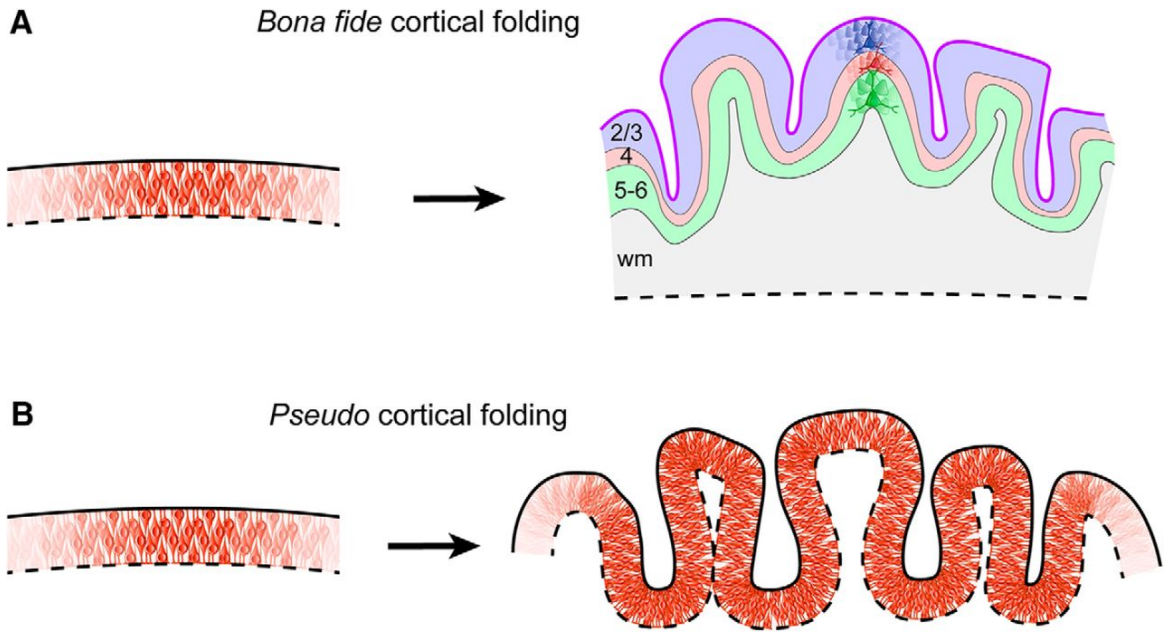


Figure 1.7: *Schematic drawings illustrating different developmental cerebral cortex folding types.* ***A****, true cortical folding (also called bona fide) is characterized by the folding of the pial surface (purple) and the underlying six neuronal layers (colored bands). In contrast, the surface of the white matter (wm) remains smooth.* ***B****, pseudo cortical folding is typically characterized by the overgrowth of the early neuroepithelium during neurogenesis. Adapted from [Borrell 2018].*

Two types of cortex can be differentiated :

- The neocortex - Located in the superficial portion of the forebrain, it is a highly organized structure of a few millimeter thicknesses (between 1 and 4.5 for the adult brain) [Fischl and Dale 2000] that processes sensory, motor, language, emotional, and associative information. It is the most significant component of the cerebral cortex, filling 90% of it and comprising six layers of neurons grouped according to their primary input or output circuitry. 80% of the neurons here are excitatory and project axons to other regions of the neocortex or deep brain structures such as the thalamus, basal ganglia, cerebellum, hindbrain, or nuclei within the spinal cord. Therefore, the neocortex plays a role in coordinating the activity of neurons throughout the central nervous system.
- The allocortex is a much smaller area taking up just 10% of the cortex and found in the olfactory system and hippocampus and has just three or four layers of neuronal cell bodies in contrast to the six layers of the neocortex.

Subcortical nuclei - In addition to cortices, grey matter can be found deeper in the brain, around the cerebral ventricles clumped together in groups of subcortical nuclei. These structures don't have the laminar organization of the cortices and can be found in the cerebrum, cerebellum, brainstem, or diencephalon. Two subcortical nuclei subgroups can be established:

- Basal ganglia - The basal ganglia are associated with various functions, including control of voluntary motor movements, procedural learning, habit learning, eye movements, cognition, and emotion. Their dysfunction results in a wide range of neurological conditions, such as disorders of behavior, control, and movement, including Tourette syndrome, obsessive-compulsive disorders, addiction, Parkinsonian syndromes, Huntington's disease, and dystonia. Among basal ganglia structures, we can cite the following: the caudate nuclei, the putamen, the nucleus accumbens, the globus pallidus, the ventral pallidum, the substantia nigra, and the subthalamic nucleus [figure1.8, 1.9].

- Diencephalon nuclei - The diencephalon acts as a primary relay and processing center for sensory information and autonomic control. It comprises four structures: the thalamus, the hypothalamus, the epithalamus, and the subthalamus. The multiple communication pathways between its structures and other body parts make the diencephalon functionally diverse. These connections include pathways to the limbic system (seat of memory and emotion), basal ganglia, and primary sensory areas, such as auditory or visual.

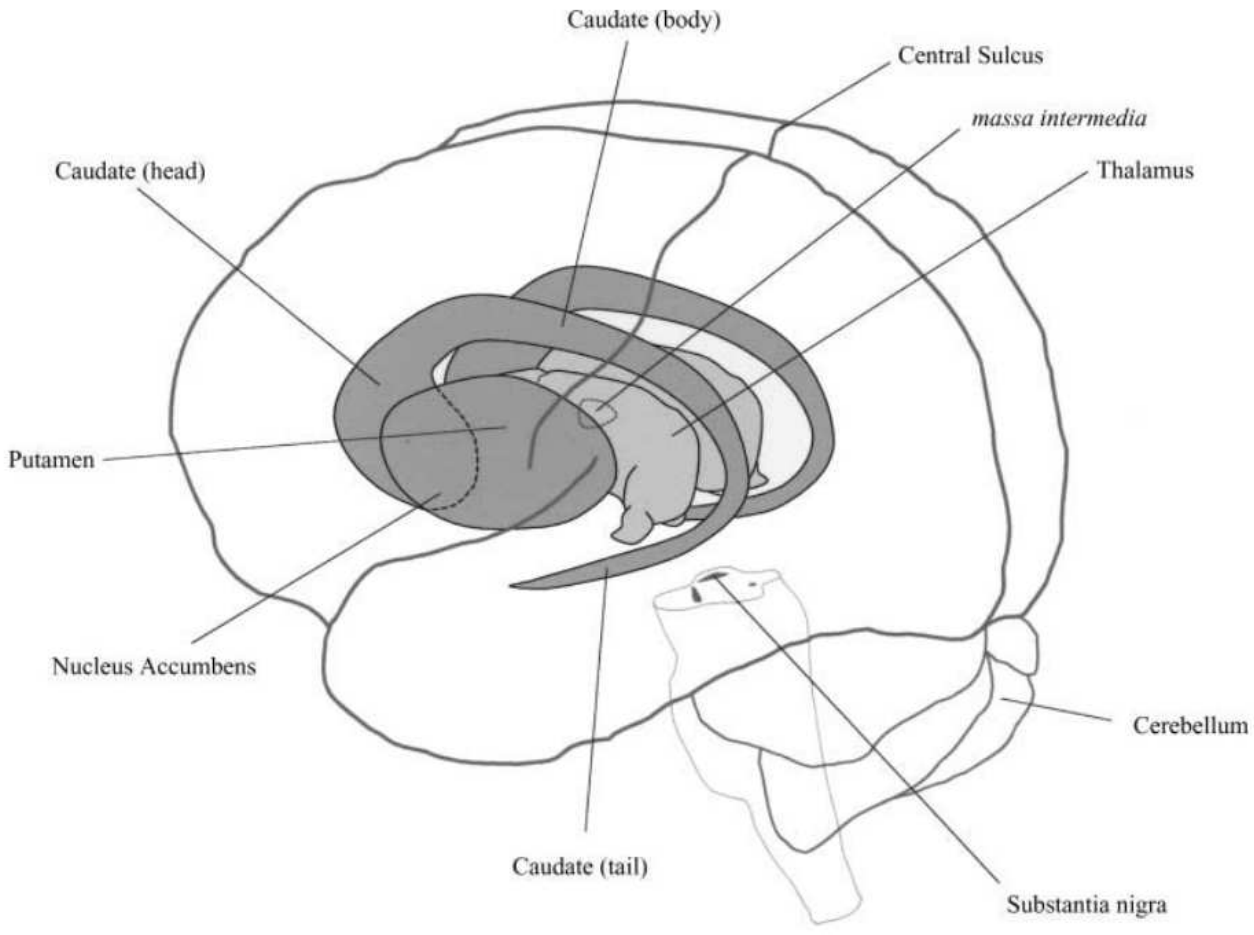


Figure 1.8: *Basal ganglia, 3D illustration. Adapted from [Anatomie-amsterdam 2024]*

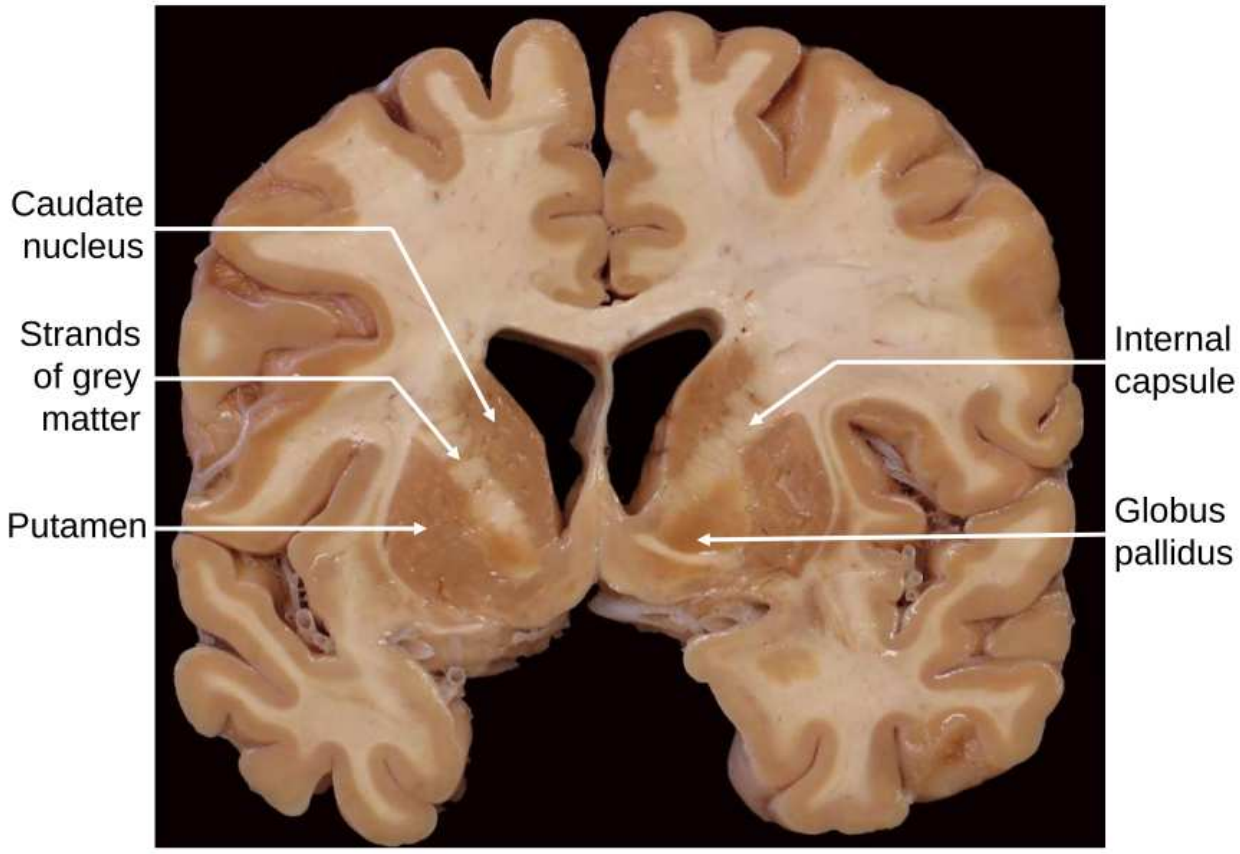


Figure 1.9: *Coronal slice of a brain's basal ganglia. The strands of grey matter within the internal capsule, connecting the caudate nucleus to the putamen, are well visible in this illustration. The putamen forms with the globus pallidus, the lentiform nucleus. Adapted from [Herrero, Barcia, and Navarro 2002]*

1.1.3.2 White matter

White matter consists of the myelinated axons of the neurons of the central nervous system. It is located deep in the brain and on the surface of the spinal cord. Afferent and efferent white matter axonal tracts interconnect cortical areas and gray matter nuclei of the brain and spinal cord. This section describes projection, commissural, and associative fibers found in white matter.

Projection fibers - Projection fibers associate the cortex and underlying structures. These fibers may be efferent or afferent, which means conducting information outwards or inwards of the cortex. Efferent/descending bundles are motor tracts, whereas afferent/ascending bundles form sensory tracts.

Commissural fibers - Commissural fibers associate both cerebral hemispheres. They include:

- The Corpus callosum - It is the largest bundle of commissural fibers located above the diencephalon and forms the roof of the lateral ventricles. It reciprocally connects the cortical areas of the right and left hemispheres, which are generally homologous. The corpus callosum comprises the rostrum, the genu, the body, and the splenium [figure1.10].
- The anterior commissural fibers - It includes interhemispheric fibers, which associate the two regions anterior temporalis, amygdala, and olfactory areas. This bundle is ten times thinner than the corpus callosum and is crucial in pain sensation, precisely sharp, acute pain.
- The posterior commissural fibers - It includes fibers involved in visual reflexes.

Association fibers - Association fibers associate different cortical regions in the same hemisphere. They can be short or long.

- Long association fibers are the inferior fronto-occipital fasciculi, the inferior longitudinal fasciculi, the middle longitudinal fasciculi, the superior longitudinal fasciculi, the ventral visual streams, the arcuate fasciculi, the frontal aslant tracts, the uncinate fasciculi, the cingulum and the fornix. See [figure1.10] for a clear picture of these bundles.
- Short association fibers also called "U-fibers" [Meynert 1885] or "superficial white matter fibers bundles" because of their shorter length, around 90 millimeters in the human brain and the shape it takes to link to close cortical regions.

1.1.4 Brain vasculature

Vasculature purpose is dual: first is to carry oxygen through the blood necessary to produce ATP, a fundamental metabolic energy molecule. The second is to evacuate brain waste. The brain accounts for an average of 2% of body weight yet consumes almost 18% of the oxygen at rest. Cerebral vascularisation has a specific organization we can schematize as a three-level system [figure 1.11]. The supply pathways represent the first level. These are made up of a vascular tripod with the two internal carotid arteries at the front and the basilar trunk at the back, formed by the junction of the vertebral arteries. The second level consists of anastomoses between the carotid and vertebra-basilar systems via the intermediary of the basilar trunk. The cerebral arteries represent the third level. They are characterized by a basal course at the base of the brain, in which they give rise to perforating branches. These arteries vascularise deep cerebral structures (white matter and nuclei). Numerous anastomoses between arterial territories complete this system. In some cases, these can partially preserve the brain from a fall in cerebral blood flow.

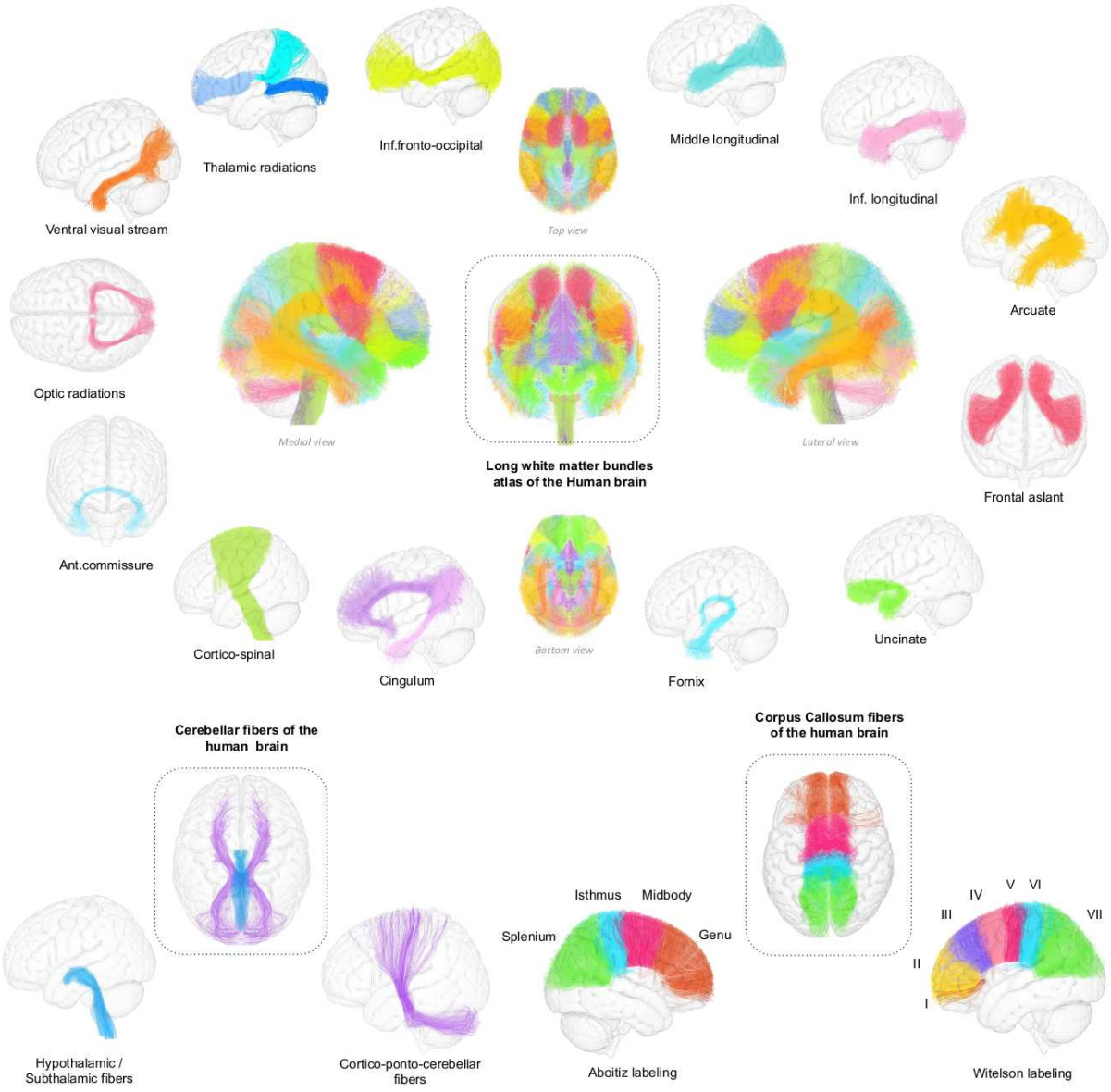


Figure 1.10: *A new deep white matter fiber bundles atlas of the human brain. 3D renderings of the 45 white matter bundles composing our long white matter atlas and including the middle longitudinal bundle, the inferior longitudinal bundle, the arcuate, the frontal aslants, the uncinate, the fornix, the cingulum (dorsal and ventral), the motor and sensory ascending and descending fibers (cortico-spinal), the anterior commissure, the optic radiations, the ventral visual stream, the thalamic radiations (anterior, superior and posterior), the inferior fronto-occipital bundle; the cerebellar fiber components: the hypothalamic and subthalamic fibers, and the cortico-ponto-cerebellar fibers; The corpus callosum (CC) components : (left) the four subparts of the CC following Aboitiz labeling (splenium, isthmus, midbody, genu) and (right) the seven subparts of the Witelson labeling. (inf: inferior). White matter bundles are all represented with different colors and superimposed to the cortex pial surface computed from the MNI template and rendered with transparency [Chauvel et al. 2023].*

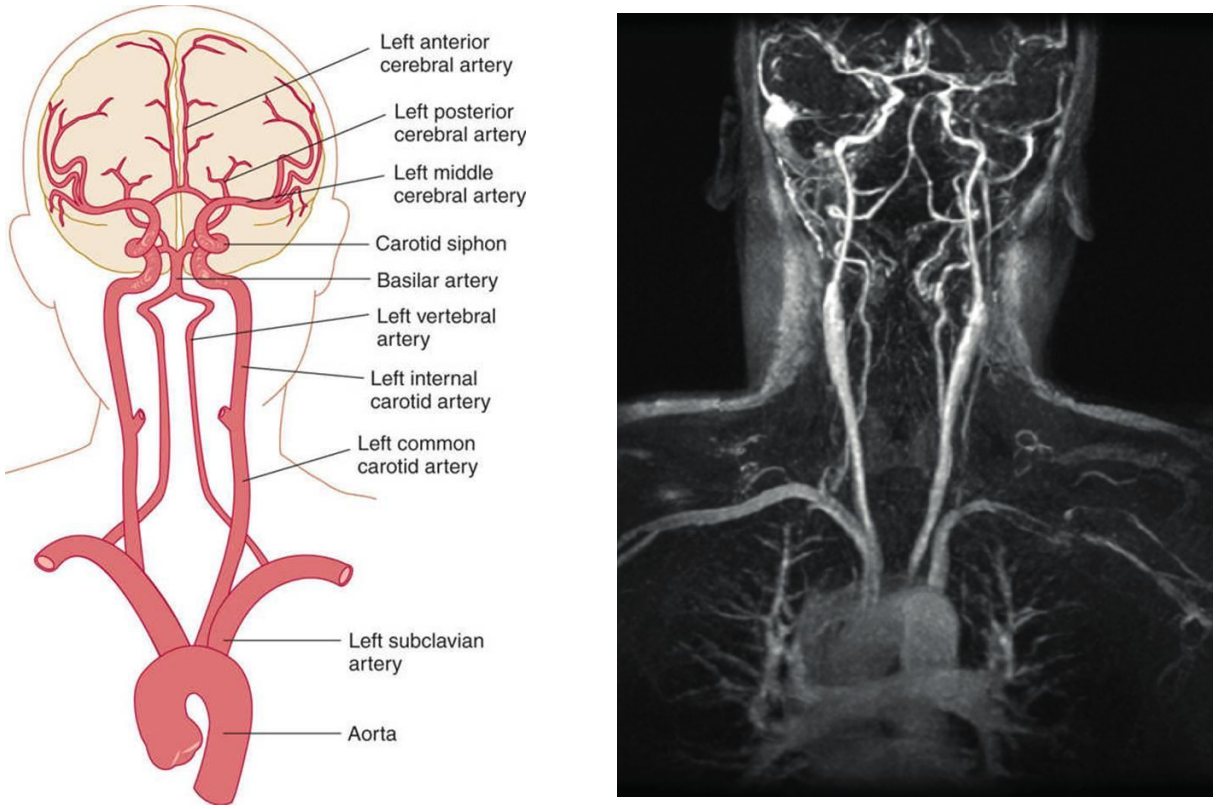


Figure 1.11: a) *Major cerebral arteries, adapted from [Neupsykey 2024]. b) Magnetic resonance angiography, anterior view, adapted from [InstitutDeRadiologie 2024].*

1.2 Microscopic anatomy

The brain can be observed at various scales; thus, we will describe its organization at microscopic scales in this section. Recent studies point out that at this scope, the observable structure within brain tissues is made of cells, around 100 billion connected neurons, and as many glial cells that handle neural tissue integrity [Von Bartheld, Bahney, and Suzana Herculano-Houzel 2016]. First, we will characterize the different cell species populating the brain tissue at the microscale, then highlight the different approaches to parcellate the brain into areas sharing homogenous microstructure.

1.2.1 Brain cells

The following section will present the cells found in cerebral tissues starting with the most common one: the neurons.

1.2.1.1 Neurons

A neuron or nerve cell is an electrically excitable cell that fires electric signals called action potentials across a neural network. Neurons communicate with other cells via synapses, specialized connections commonly using chemical neurotransmitters to pass the electrical signal from the presynaptic neuron to the target cell through the synaptic gap. The neuron comprises a cell body (soma), dendrites, and axon extensions, which will be described hereafter.

Soma - The soma is the core of the neuron and the site of significant metabolic activity. The soma usually measures between 4 to 100 micrometers in diameter. A cytoplasm wraps it and contains the

nucleus that encloses the DNA and many organelles, including granules called Nissl corpus and free ribosomes, synthesizing proteins. The Soma also has different organites, such as the mitochondrion responsible for cell energy supply or the Golgi apparatus, which regulates the transport of molecules through the membranes and transforms the proteins and the lysosomes [figure 1.12].

Dendrites - Dendrites are extensions of the soma with which they share some organites. They measure between 1 to 2 µm and their diameter varies from 0.2 to 3 µm for dendrites close to their soma. For neurons in the cerebral cortex, the dendritic trees are mostly isotropic, whereas elsewhere, like in the layers of the hippocampus, the trees can be highly anisotropic [Fiala, Harris, and Spacek 1999].s

Axons - They are long tubular extensions of the neuron that typically conduct electrical impulses known as action potentials away from the soma. Axons may extend from a few millimeters in the case of superficial white matter bundles to over one meter in corticospinal neurons [Schüz and Braitenberg 2002]. Fiber inner diameters are reported to range from 0.16 to 9 µm in the cerebrum with a large majority of axons with average diameter values below 1.0µm [Aboitiz, R.S. Fisher, and E.Zaidel 1992; Liewald et al. 2014], and are up to 15 µm in the human spinal cord [Häggqvist and al 1936]. Myelin is a lipide and dielectric material that wraps around axons in the nervous system to protect the axonal membrane and provide efficient conduction of neural signals. Fiber diameter is related to conduction velocity: in myelinated axons, conduction velocity increases approximately linearly with axon diameter [Gasser and Grundfest 1939].

Larger axons and a thicker myelin sheath create faster conduction, but there is a trade-off between myelin thickness and axon size due to spatial constraints imposed by the brain dimensions. Data derived from serial reconstructions of the adult mouse cortex showed that the degree of myelination can even vary along a single axon. Thus, neurons located in superficial cortical layers exhibit both myelinated and large unmyelinated segments, contrasting with the pattern observed, for instance in the spinal cord or the optic nerve where myelinated axons display regular internodes [Tomassy et al. 2014] called Ranvier nodes [Salzer 1997] [figure 1.12]. In general, larger axons tend to have thicker myelin sheaths, but the relationship between the two is not linear [Berthold, I. Nilsson, and Rydmark 1983]. There is a link between axon size and myelin thickness captured by a parameter called the myelin g-ratio, defined as the ratio of the inner (axon) to the outer (axon plus myelin) diameter of the fiber. The g-ratio that provides maximum conduction speeds, has a broad optimal value around 0.77 [Chomiak and Hu 2009; Stikov et al. 2015].

The Ranvier nodes internode distance d has been extensively studied in [Rushton 1951] leading to the maximum conduction relationship :

$$\frac{d}{D} \propto kg\sqrt{\log\left(\frac{1}{g}\right)}, \tag{1.1}$$

where K is a constant, D is the external diameter of the axon (including the myelin sheath) and g is the g-ratio defined as the ratio between the axonal membrane and the external myelin sheath outer membrane diameters.
Neurite - Semantically, we refer to as "neurite," any projection of the cell body of a neuron. This projection can be either an axon or a dendrite. This designation comes from the fact that it can be challenging to differentiate axons from dendrites in imaging.

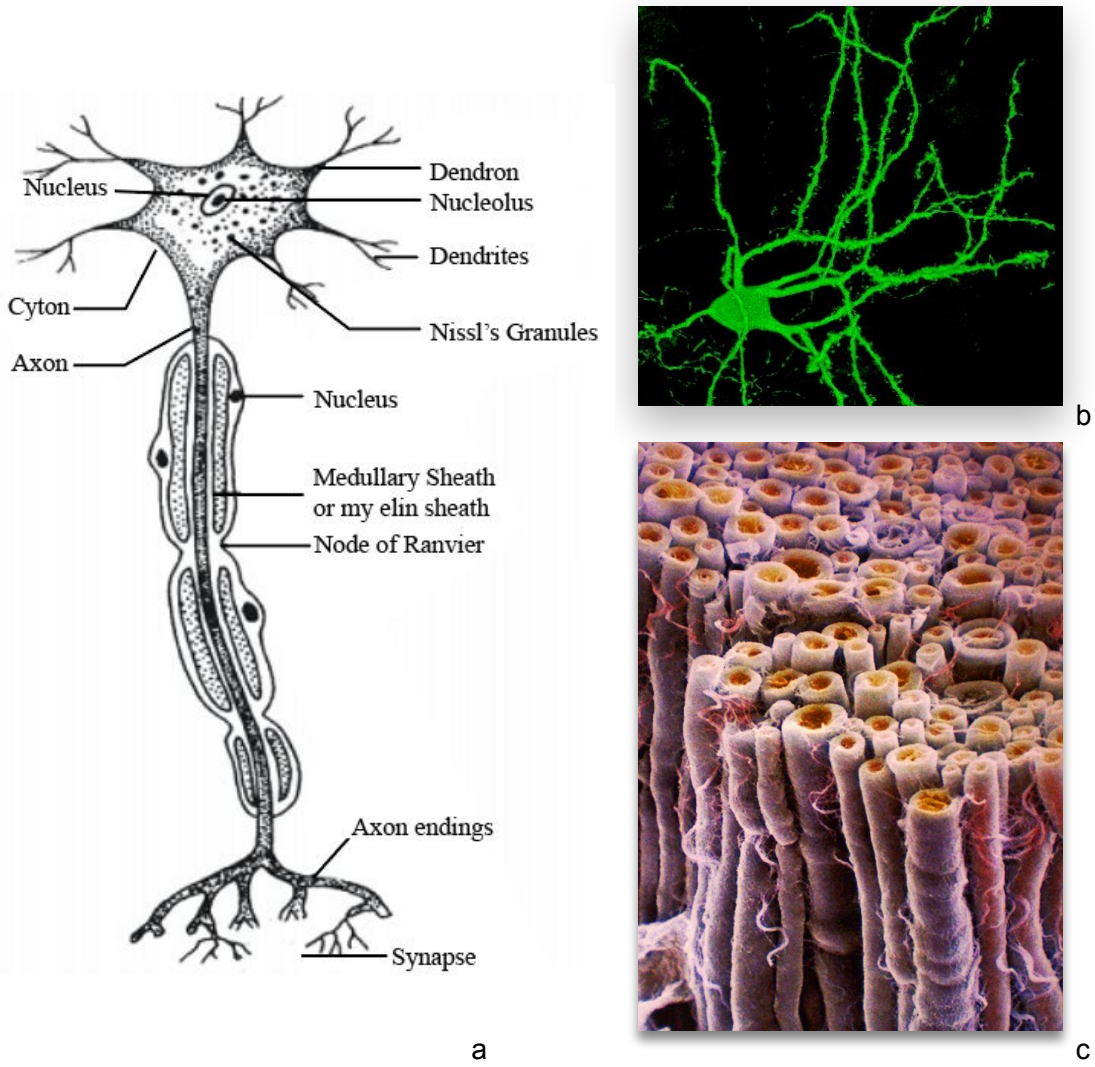


Figure 1.12: a) *Scheme of a neuron cell exhibiting its main components: the cell body or soma found in grey matter, the dendrites, the axon with a myelin sheath interrupted by Ranvier nodes. The nerve impulse is received by the dendrites and is propagated along the axon towards the axon terminals and the synapses, adapted from Sarthaks.* b) *Confocal image of an eGFP filled striatal medium spiny neuron, from [Carlson et al. 2011].* c) *Artistic view of a cross-section of a bundle of nerve fibers adapted from Steve Gschmeissner work.*

1.2.1.2 Glial cells

In 1858, Rudolf Virchow first proposed that neuroglia comprised the brain's connective tissue and was composed of cellular elements [Virchow 1858]. Later, Camillo Golgi visualized astrocytes within the nervous system and developed the concept that these cells constituted the glue of the brain [Golgi 1885]. Glial cells are 80% water and not electrically excitable, they do not conduct action potentials like their neuronal counterparts.

Astrocytes - The term astrocyte, which referred to the stellate morphology of these cells, was first used in 1893 by Michael von Lenhossek [Lenhossék 1893]. Among glial cells, astrocytes are the most numerous in the central nervous system [figure 1.13]. They participate in all essential CNS functions, including blood flow regulation, energy metabolism, ion and water homeostasis, immune defense, neurotransmission, and adult neurogenesis [Lundgaard et al. 2014]. Their somas vary between 10−15µm in diameter, with an overall diameter of about 15µm if we consider the processes [N. Oberheim et al. 2009]. These cells were subdivided into fibrous and protoplasmic astrocytes by Kolliker and Andriezen [Kölliker 1889; Andriezen 1893].

- Protoplasmic astrocytes - They are distributed relatively uniformly within the cortical gray matter and have many fine processes (50µm long on average). Their density varies between 10 000 and 30 000 per mm^3. The surface area of their processes is huge (up to 80 000 m^2) and covers almost all neuronal membranes within reach.

- Fibrous astrocytes - They are organized along white matter tracts, within which they are oriented longitudinally in the plane of the fiber bundles, their processes are long (up to 300µm), though much less elaborate with fewer branching compared to protoplasmic astroglia. The density of fibrous astrocytes is around 200,000 cells per mm3. Experimental evidence points to fibrous astrocytes as being crucial in facilitating normal myelination during development, maintaining the right environment for oligodendrocytes, and also ion buffering and metabolic supply throughout adulthood [Lundgaard et al. 2014; N. Oberheim et al. 2009]. Fibrous astrocyte processes also create numerous extensions ("perinodal" processes) that connect axons at nodes of Ranvier, the sites of action potential propagation in myelinated axons. The cell bodies are equally spaced, probably because they provide structural support for the axons tract [N. Oberheim et al. 2009; N. A. Oberheim, Goldman, and Nedergaard 2012].

Oligodendrocytes - It was not until 1919 that oligodendrocytes and microglia were first identified as separate cell types [Kettenmann and Verkhratsky 2008]. Oligodendrocytes are glial cells with few processes predominantly found in white matter. Oligodendrocyte's main function is to create myelin which insulates axons in the central nervous system. Oligodendrocytes also participate in the development of nodes of Ranvier and determine their periodicity. They were first described in [Rio-Hortega 1928] [figure 1.13], where these cells were classified into four main phenotypes (I-IV) depending on their morphological appearance, the number of processes, and the size of the fibers they connect.

- Type I - Characterized by having a small rounded cell body (15–20 m diameter) and a high number (from 5 to 20 or more) of very fine and approximately 100 − 200µm length processes emerging in multiple directions and towards 10 to 30 thin myelinated axons (diameter < 2µm). They are present in gray and white matter.

- Type II - They are only present in white matter but similar to type I. They are polygonal or cuboidal in shape (20–40 m) with fewer and thicker processes than type I, which are directed to axons and attached to them longitudinally.

- Type III - Paladino had intuited that myelin had a neuroglial origin [Paladino 1892]. These are also less abundant than types I and II and have a 200 to 500µm internodal length. They are present in white matter with thick myelinated fibers (4 to 15µm in diameter) in the brain stem and spinal cord. They are distinguished by one to four processes emanating from a bulky cell body and directed toward axons.

- Type IV - These cells don't have processes and form a single long myelin sheath (as great as 1000µm internodal length) on the largest diameter axons; type IV oligodendrocytes are located almost exclusively around the entrances of the nerve roots into the CNS.

It seems likely that signals from axons of different caliber regulate oligodendrocyte phenotype divergence. The dimensions of the myelin sheath determines the conduction properties of the axons in the unit, whereby axons with long, thick myelin sheaths (type III/IV oligodendrocyte axon units) conduct faster than those with short, thin myelin sheaths (type I/II oligodendrocyte–axon units). In addition to these classical myelin-forming oligodendrocytes, a small population of non-myelinating oligodendrocytes known as "satellite oligodendrocytes" are present in the grey matter, where they are usually applied to neuronal perikarya. The function of these satellite oligodendrocytes remains poorly understood.

Microglia - Microglial cells are the immunocompetent cells residing in the CNS. Their soma is around 10µm in diameter, and total coverage (with processes) is about $15 - 30$µm [figure 1.13]. Microglia form the brain's immune system, activated by various brain injuries and diseases. Microglial cells represent about 10% of all glial cells in the brain. In the mature CNS, microglial cells may appear in three distinct states: the resting microglia, activated microglia, and phagocytic microglia. In the normal brain, microglial cells are resting and characterized by a small soma and numerous very thin and highly branched processes. Every microglial cell is responsible for a clearly defined territory of about $50\ 000 m^3$ in volume. There is a clear morphological difference between microglial cells residing in the grey versus white matter: the former extend processes in all directions, whereas the latter processes are usually aligned perpendicularly to the axon bundles. Microglial cells are equipped with numerous receptors, which make them perfect sensors of the status of the CNS tissue: brain injury is immediately sensed, which initiates the microglia activation process. This process turns microglia into an activated (or reactive) state, and some of the activated cells proceed further to become phagocytes. Both reactive microglia and phagocytes provide an active brain defense system [Verkhratsky and Kirchhoff 2007]. Some experimental evidence reported in [Traiffort et al. 2020] points to astrocytes and microglia as being crucial in both developmental and repairing oligodendrogenesis and myelination.

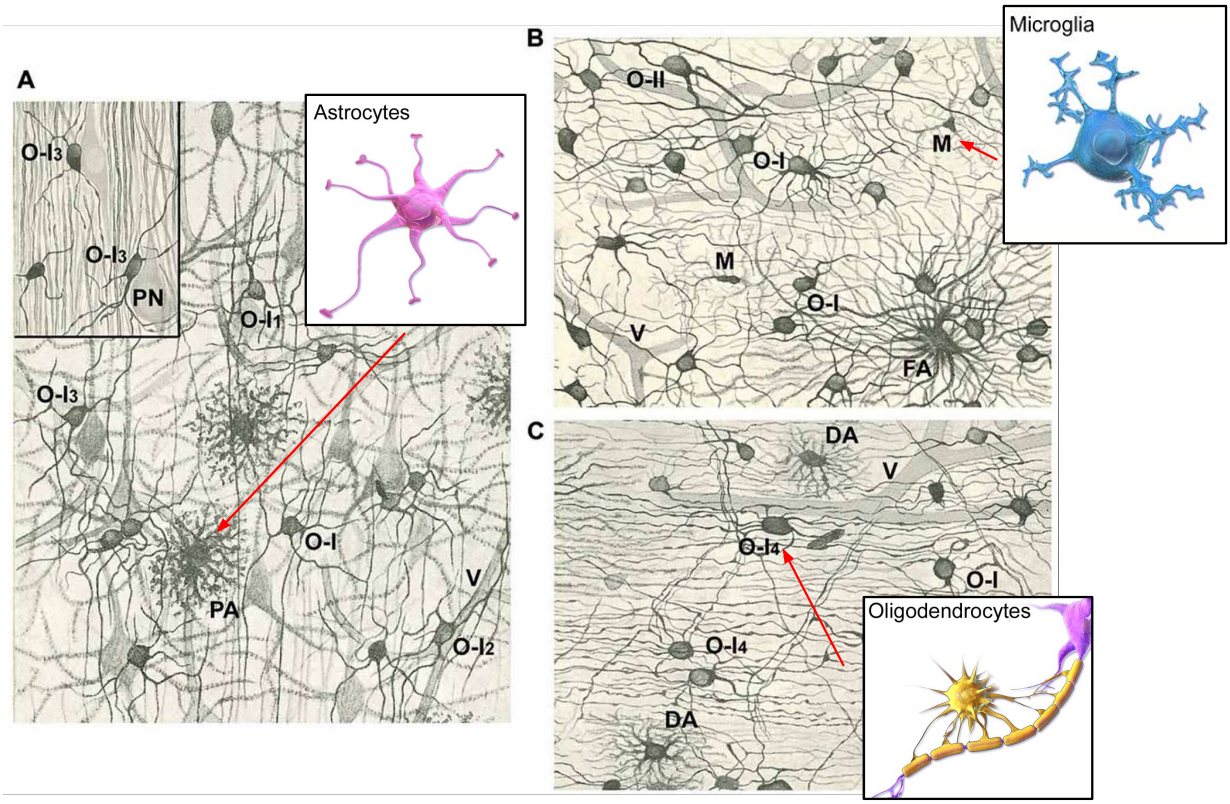


Figure 1.13: *Drawings of the cerebral cortex (A) and white matter (B, C) after staining with the Golgi-Hortega method or the silver carbonate procedure by Hortega (inset in A). (A) Notice pyramidal neurons (PN), protoplasmic astrocytes (PA), vessels (V), and type I oligodendrocytes (OLGs; O-I) with a variable number of processes, many of them divided in "Y" or "T". Some OLGs have processes mainly oriented in the direction of projecting axons (O-I1), while others have a perivascular (O-I2) or perineuronal (O-I3; see inset) localization. (B) Note a fibrous astrocyte (FA), some OLGs of the first type (O-I) and one of the second type (O-II), as well as microglia cells (M). (C) See type I OLGs similar to those in (A, B) (O-I) or with long processes that follow axons (O-I4) and two dwarf astrocytes (DA). Vessels (V) are also drawn in (B, C). Magnification in Figures (A–C) is similar. Modified from [Rio-Hortega 1928].*

Ependymal cells - The ependyma is a simple ciliated epithelium that lines the ventricular surface of the central nervous system, extending from the lateral ventricles to the filum terminale [Del Bigio 2010]. Mature ependymal cells appear along the neural tube as the ventricular zone regresses [Bystron, Blakemore, and Rakic 2008] and are characterized morphologically by a cuboidal to columnar shape and a round nucleus with fine stippled chromatin patterns and inconspicuous nucleolus. On the surface of the ependymal layer is a network of supra-ependymal axons. In 1992, Sarnat alluded to a transport role for moving small molecules between CSF and the brain [Sarnat 1992]. In a 1995 review, Sarnat dealt with reactions of ependymal cells to injury, he emphasized the general inability of ependymal cells to proliferate, the development of reactive astroglial changes and buried clusters of ependymal cells at sites of erosion, and how viral infection of the ependyma could result in hydrocephalus [Salzer 1997]. Adult lacks ependymal cells over large areas of their lateral ventricles.

Microvasculature - The microvasculature comprises fine capillaries. Capillary walls contain pericytes that have contractile properties and are lined by a highly specialized blood-brain barrier that regulates the entry of solutes and ions and maintains the integrity of the composition of CSF. They are also crucial for the production of CSF [Agarwal and Carare 2021]. The density of the brain capillaries varies within the brain, depending on location and energy needs, with higher capillary density in gray versus white matter [Cipolla 2016]. The densities of capillaries and neurons are proportionally related, with typically about 10µm of capillaries per cortical neuron [Karbowski 2011]. The complex cross-talk between all entities and cell types of the brain microvasculature is collectively known as the neurovascular unit (NVU). The NVU is a structure formed by neurons, interneurons, astrocytes, and basal lamina covered with smooth muscular cells and pericytes, endothelial cells, and extracellular matrix. Each component is intimately linked to each other, establishing an anatomical and functional whole, which results in a highly efficient system of regulation of cerebral blood flow [Sweeney, Sagare, and Zlokovic 2015].

1.2.2 Cortex histology

At the beginning of the twentieth century, the Cécile Vogt-Mugnier and Oscar Vogt couple started a research program aimed at the parcellation of the cerebral cortex into morphological and potentially functional units. For the detection of the local structural differences upon which their parcellations were based, they consistently made use of two histological techniques, which reduce the ultra-complexity of the cortex to manageable proportions, via the Nissl technique and the Weigert technique [Nissl 1904]. The Nissl technique stains only neuron granules (called Nissl bodies) and somas of the neurons in blue, leaving their dendritic and axonal processes almost entirely unstained. The Weigert technique stains only the myelin sheaths surrounding the neuronal axons. These two techniques yielded the emergence of two new investigational fields, which were designated by Vogt in 1903 as cytoarchitectonics and myeloarchitectonics [Nieuwenhuys and Broere 2020]. The Vogts delegated the cytoarchitectonic analysis of the cerebral cortex to Korbinian Brodmann, who collaborated with them. The myeloarchitectonic part of the research program of the Vogt's was initiated by Vogt himself and continued by numerous popular neuro anatomists [Vogt 1910; Hopf 1951; Sanides 2013; Strasburger 1937]. Recently, Nieuwenhuys [Nieuwenhuys 2013] scrutinized the total body of myeloarchitectonic studies of the Vogt-Vogt school. It was concluded that the data available is adequate to establish a myeloarchitectonic map of the entire human neocortex, which was recently published [Nieuwenhuys, Broere, and Cerliani 2015a; Nieuwenhuys, Broere, and Cerliani 2015b]. It allowed a direct comparison of the cytoarchitectonics and myeloarchitectonics on the human neocortex, proposed

by Vogt's school [figure 1.15].

1.2.2.1 Cytoarchitectonic parcellation of the cortex

There is no doubt that among the many proposals of brain mapping, the one that achieved the most historical popularity was the one from Korbinian Brodmann [Bruner 2022]. Indeed, his work [Brodmann 1909] offered an unprecedented view of histological structure and organization of cells within the cortical tissue, also called cytoarchitecture [figure 1.14]. Cytoarchitectonics information, which are variations in cell shape, size, and density allowed him to depict 43 different areas within the cortex thanks to the Nissl staining technique [Nissl 1904]. His map has been further improved and discussed by Von Economo and Koskinas [Economo and Koskinas 1925] in 1925 or Sarkisov in 1945 [Sarkisov, Filimonoff, and Preobrashenskaya 1949] and is still nowadays subject to controversy due to its histological observations based on a single brain, thus not relevant for interindividual variability or to the fact that compared with the about 180 areas defined using other approaches [Amunts and Zilles 2015], the Brodmann's approach probably underestimated the actual cytological diversity. Nevertheless, the map proposed by Brodmann is still nowadays considered a cornerstone of brain anatomy and histology because Brodmann's map simplicity is part of its strength by supplying a handier mapping that can be used in various fields [Bruner 2022].

Nowadays, brain parcellation has moved forward from the early pioneering methods and it is based on a far more complex array of criteria, principles, and methods [Eickhoff, Yeo, and Genon 2018; M. F. Glasser et al. 2016; Essen, Donahue, and M. Glasser 2018]. part of them, the advent of diffusion MRI, allowing to decode the microstructure non-invasively, coupled with diffeomorphic registration, frameworks are now enabling the construction of atlases of brain microarchitectural features from different subjects that complement and improve the former atlases by accounting for the human brain's interindividual variability. To finish, many of the areas defined by Brodmann have since been correlated closely to diverse cortical functions; area number 17, for instance, corresponds to the primary visual cortex. Some of these correlations are, however, still under debate [Tremblay et al. 2011].

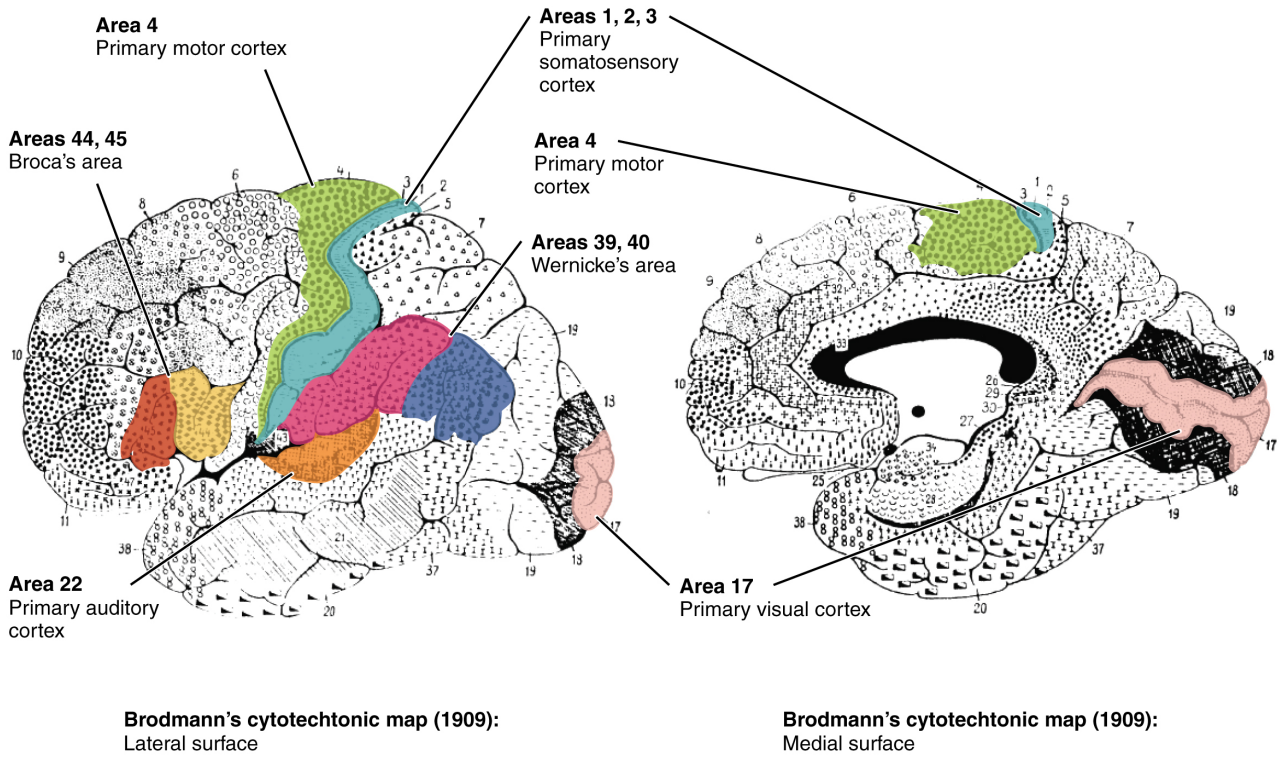


Figure 1.14: *Brodmann's areas of the cerebral cortex. Each area presents a distinct cytoarchitecture and is labeled with a specific number (image taken from illuminamente, based on original work by Korbinian Brodmann.*

The cytoarchitectonic view of a neocortex sample [figure 1.15] enables to identify six cytoarchitectonics layers. These layers are:

- (I) - The cell-poor zonal layer or molecular layer, with few neurons composed mainly of glial cells, apical dendrites, and axons parallel to the pial surface.
- (II) - The external granular layer mainly comprises small granular cells that receive afferents from and send efferents to other cortical areas.
- (III) - The external pyramidal layer containing small pyramidal cells. Layers I- III are referred to as supragranular layers and constitute the primary origin and termination of intracortical connections.
- (IV) - The internal granular layer, including dense packing of granular cells.
- (V) - The internal pyramidal layer, containing large pyramidal cells whose axons leave the cortex and run down to subcortical structures and Martinotti cells.
- (VI) - The multiform layer or polymorphic layer contains few large pyramidal neurons and many small multiform neurons. Layers V-VI are infragranular layers that are most developed in motor cortical areas.

1.2.2.2 Myeloarchitectonic parcellation of the cortex

The cortex contains numerous myelinated fibers, which follow two principal orientations within the cortical ribbon, tangential and radial. The tangential fibers tend to form local concentrations or bands, some visible with the naked eye in unstained sections [Nieuwenhuys 2013]. The radial fibers are concentrated in bundles. Their representation as myeloarchitectonic layers is shown in [figure

1.15] and can be compared to the cytoarchitectonic organization of the cortex. In both schemes, the neocortex is subdivided into six layers. The corresponding myeloarchitectonic layers are:

- (1) - The zonal layer is differentiated into four sublayers: the narrow sublayer 1o, which contains only very few fibers, and the external, intermediate, and deep sublayers 1a, 1b and 1c, of which 1a contains clearly more fibers than 1b and 1c.
- (2) - The dysfibrous layer which contains, just like sub-layer 1o, only very few fibers.
- (3) - The suprastriate layer has three sub-layers, of which the superficial sublayer 3a1 is richer in fibers than the remaining sublayers 3a2 and 3b. The presence of the end segments of the radial bundles characterizes sublayer 3b.
- (4) - The external stria or outer stripe of Baillarger forms a dark band of tightly packed, tangential fibers.
- (5a) - The intrastriate layer is generally relatively poor in tangential fibers, thus contrasting with the bordering stripes of Baillarger.
- (5b) - The internal stria or inner stripe of Baillarger is a dense plexus of tightly packed tangentially oriented fibers.
- (6) - This layer is subdivided into the pale substriate lamina 6a1 and laminae 6a2, 6b1 and 6b2, which show an increasing wealth of tangentially oriented fibers. Sublayer 6b2 forms the zone of transition to the subcortical white matter.

1.2.3 White matter histology

The white matter comprises three main components: neuronal fibers extending from neuron bodies, glial cells (including astrocytes, microglia, and oligodendrocytes), and blood vessels. Most of the space is occupied by neuronal fibers (between 60% and 80% of the space) [Mottershead et al. 2003; Perge et al. 2009; Stikov et al. 2015]. The extracellular space constitutes about 20% of the volume [Bourne 2012; Voříšek et al. 2002], blood vessels constitute less than 3% [Voříšek et al. 2002], and glial cells fill the remaining 15% of the white matter volume. Their density is constant in the brain, approximately 10^5 cells/mm^3 according to [S. Herculano-Houzel 2014]. If we consider the myelin sheath as separated from its axon, approximately 25-30% of the volume is occupied by myelin sheaths in adults [Mottershead et al. 2003; Perge et al. 2009; Stikov et al. 2015]. Moreover, 70% to 95% of nerves are myelinated fibers [Biendenbach, DeVito, and A. C. Brown 1986; Liewald et al. 2014]. Fibers are bundled together to form homologous populations of fibers sharing similar orientations or shapes. In a typical white matter imaging 3D voxel of a millimeter scale, we can observe different fiber populations crossings, kissing, or splitting, and studies in the field of diffusion MRI [T. E. J. Behrens et al. 2007] have reported a percentage of around 60% of voxels containing such features. Furthermore, [T. E. J. Behrens et al. 2007] only found that one-third of voxels contain two fiber populations.

Fiber inner diameters are reported to range from 0.16µm to 9.0µm in the human brain [Aboitiz, R.S. Fisher, and E.Zaidel 1992], [Liewald et al. 2014]. Most of the axons have a diameter below 1.0µm. Mean axon diameter distribution heterogeneity across white matter regions leads to a significant variation of nerve fibers density, around 10^5 cells/mm^3 [S. Herculano-Houzel 2014]. Within a fiber population, we observe a macroscopic angular dispersion that characterizes the global misalignment

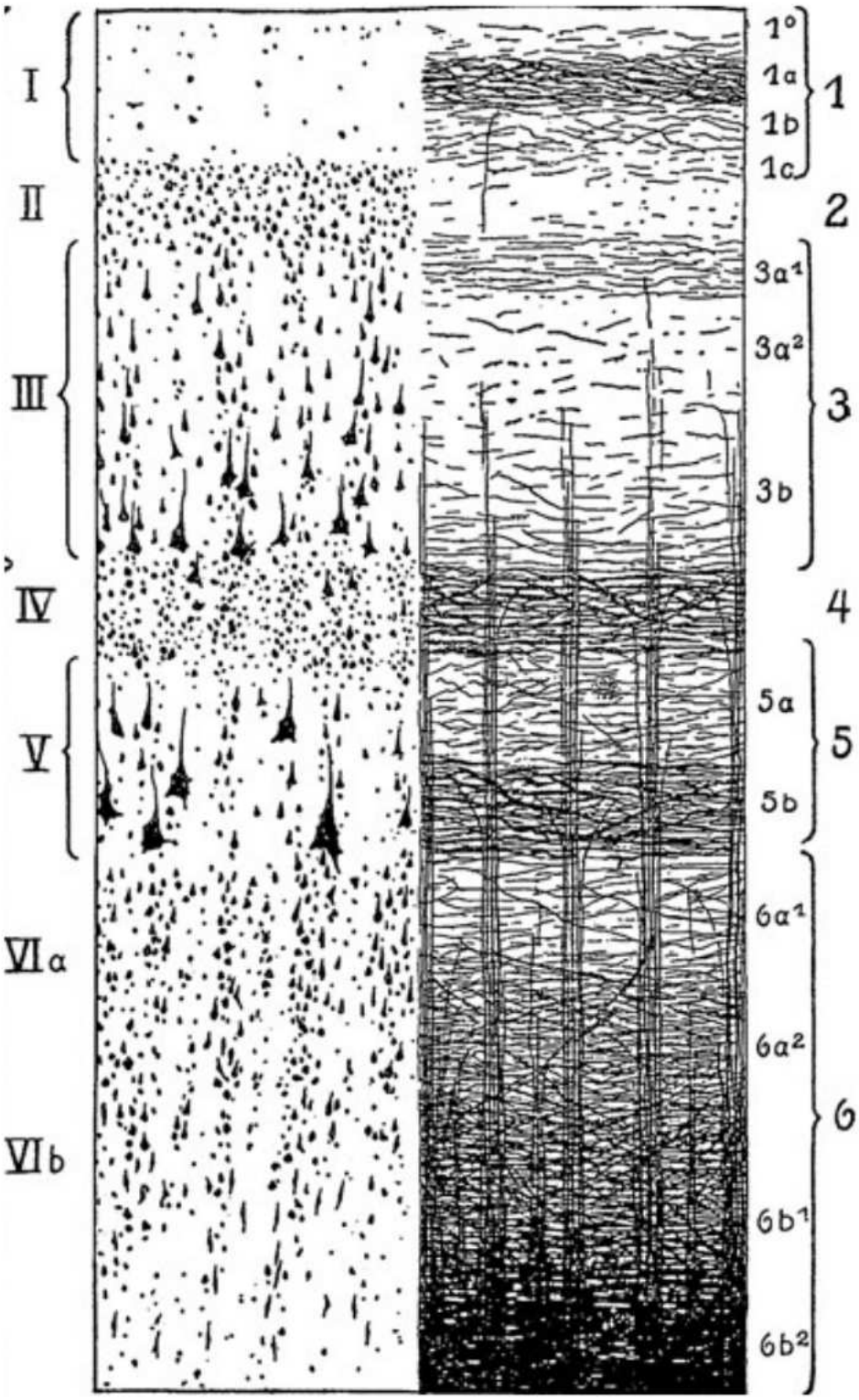


Figure 1.15: *Scheme of the cytoarchitectonics layers (designated with Roman numbers) and the myeloarchitectonics layers (designated with Arabic numbers), from [Nieuwenhuys 2013].*

of axons, as well as a local tortuosity, which can be defined as the ratio between the geodesic distance along the curvilinear frame specified by the centroid axis of the axonal fiber and the Euclidean distance between the two extremities of the fiber. Values of microscopic misalignments of axons estimated up to 18 degrees were reported in [Ronen et al. 2014].

1.2.4 Conclusion

This chapter introduced the primary material necessary to understand the various structures forming the brain at macro and micro-scales. All these components are crucial to remember when we try to picture a brain with MRI techniques, as every image produced results from integrating the observation of physical processes across scales. In the framework of this thesis, we will exploit one particular imaging modality, diffusion MRI, which will be detailed exclusively in the next chapter.

Chapter 2

Toward Magnetic Resonance Imaging

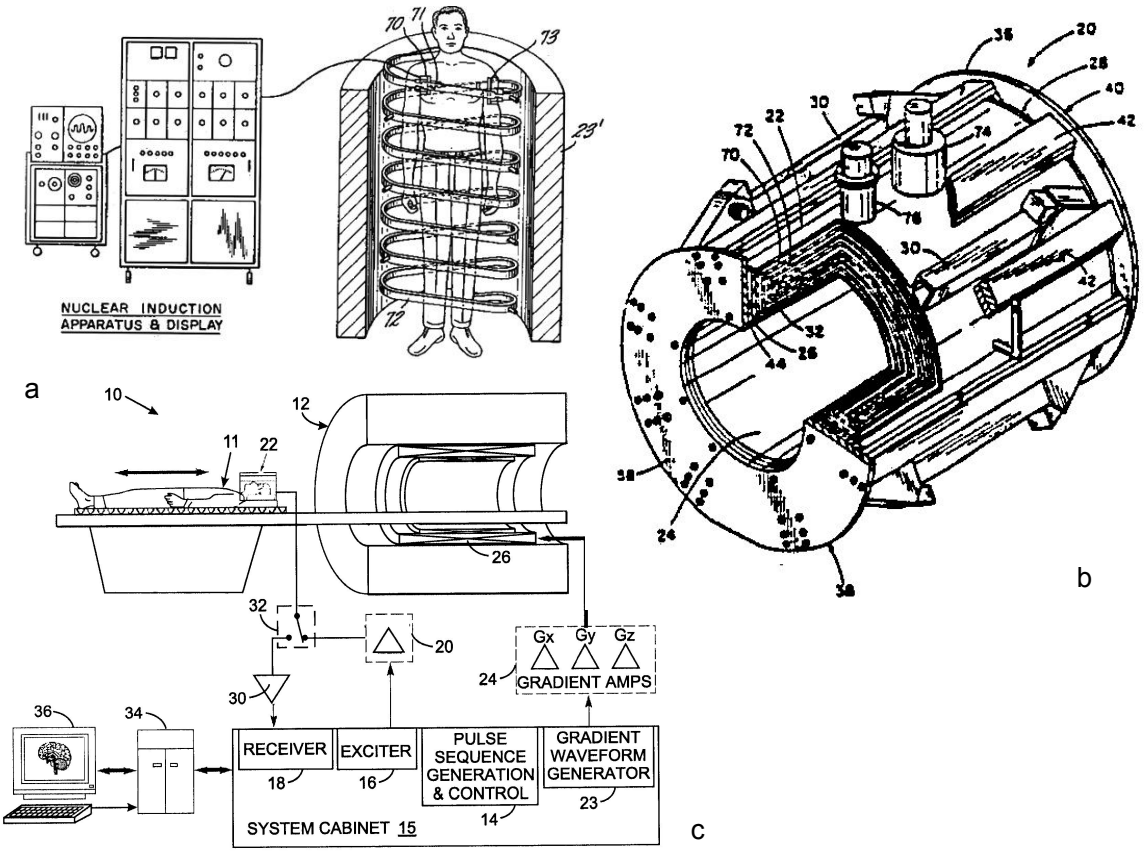


Figure 2.1: *a) Damadian MRI patent drawing, adapted from [Wikimedia 2020]. b) "A magnet system for providing a localized substantially homogeneous field for use in medical magnetic resonance imaging and its incorporation into a method of siting", adapted from a 1987 patent [Breneman, J. R. Purcell, and Burnett 1987]. c) MRI global functioning, adapted from [Hotcore 2024].*

Chapter outline

2.1 Historical background 30
2.2 Fundamentals of magnetic resonance imaging 32
2.2.1 Nuclear Magnetic Resonance Principle 32
2.2.2 Image encoding and reconstruction 36
2.2.3 Conventional MRI sequence schemes 38
2.3 Diffusion MRI 40
2.3.1 Brownian Motion 40
2.3.2 Sensitizing MRI to the diffusion process 42
2.3.3 Models of the diffusion process 47
2.4 Conclusion 55

Magnetic resonance imaging (MRI) has revolutionized the exploration of the human brain since the '80s. Not only, does MRI give access to the brain anatomy but can also be used to explore brain functions or metabolism. The main impacts of MRI on neuroscience come from two remarkable coincidences: first, the water molecule's magnetic resonance properties, and second, the human body is made of at least 60% water. In addition, the brain, with its CFS is a likely organ to image with MRI if we consider that the scientific community of the last century was looking for a great substitute to invasive imaging methods (such as surgery or X-ray) to investigate the brain in detail. In this chapter, we will give a short overview of the history of its development and then focus on a specific MRI modality at the heart of this thesis work, diffusion-weighted MRI (DMRI).

2.1 Historical background

MRI is a relatively recent field of research that started about 60 years ago. Yet, this short period has seen over 600.000 publications containing the term "MRI" as a keyword and over 40.000 in 2020 alone [Viard, Eustache, and Segobin 2021].

Factually, we can start the story of MRI with the discovery of nuclear magnetic resonance (NMR). In 1938, Isidor Rabi first described and measured this NMR phenomenon [Rabi et al. 1938] by launching a beam of particles in a static magnetic field and observing the induction of radio frequencies whose frequencies were dependent on the charge of the particles and the magnitude of the magnetic field. Bloch and Purcell replicated the experiment in solids and liquids and established the well-known phenomenological equations known as the Bloch equation.[Packard 1946; Bloch 1946; E. M. Purcell, Henry Cutler Torrey, and Pound 1946]. Further properties were discovered in 1950, when Erwin Hahn detected spin echoes and free induction decay [Hahn 1950], and in 1952 when Herman Carr produced a one-dimensional NMR spectrum [Carr and E. M. Purcell 1954].

At this point, the main issue was the absence of spatial localization of the NMR signal until Vladislav Ivanov and Raymond Damadian proposed a patent for an MRI device respectively in 1960 and March 1971 [MacWilliams 2003]. These attempts at devising the first MRI prototype were largely unsuccessful

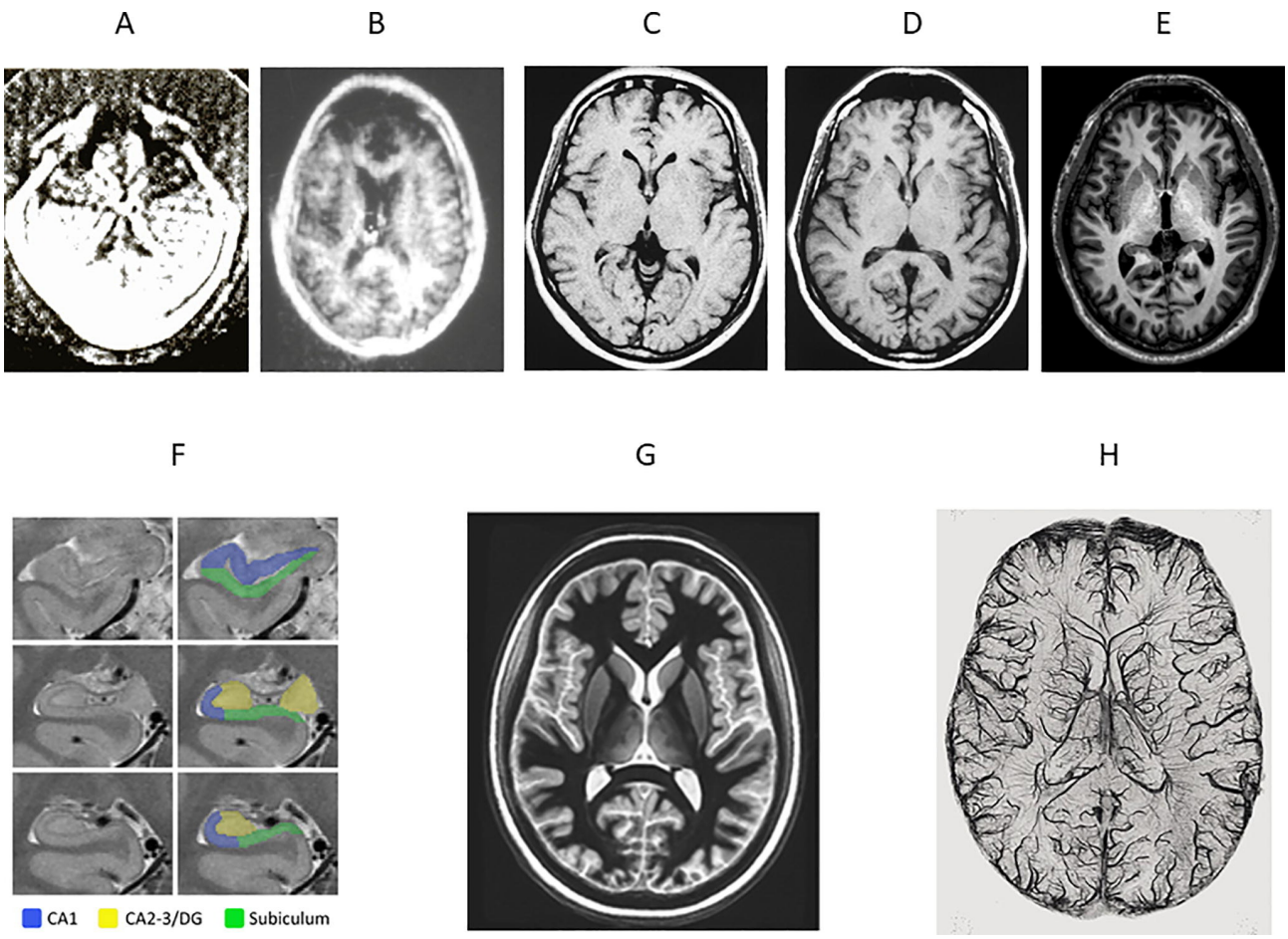


Figure 2.2: *Adapted from [Viard, Eustache, and Segobin 2021]: History of progress in MRI brain imaging as scanners with higher field strengths were developed. (A) The first brain image published in New Scientist in 1978 showing a person's eye-balls and ventricles [Clow and Young 1978] (B) The T1-weighted Brain MRI presented by Professor Godfrey Hounsfield during his Nobel Prize lecture in 1979 from (ISMRM) (C) T1-weighted image at 0.5 T, (D) 1.5 T [Maubon et al. 1999] and (E) 3 T. High-resolution of distinct regions of the brain are now observed with dedicated sequences like a proton-density image (F) of the hippocampus acquired at 3 T [Postel et al. 2019] where the CA1, CA2/CA3 and subiculum subfields are distinguishable to the trained eye. Part (G) shows the acquisition of a white-matter nulled sequence at 7 T [Su et al. 2019], which is now clinically available, and (H) shows MRI from an ultra-high field scanner at 9.4 T [Nowogrodzki 2018].*

until the fall of 1971 when Lauterbur came up with the idea of using magnetic field gradients along the three dimensions of the space with the back-projection technique typically used in computed tomography. This technique was used to acquire the first MR images of two water-filled tubes in 1973 [Lauterbur 1973]. Later, Peter Mansfield introduced echo-planar imaging [Mansfield 1977], a technique that reduced scanning time from hours to seconds. Raymond Damadian showed the first tumor image in a mouse's thorax in 1973, and by 1977, his research group published the first MRI body scan of a human subject [MacWilliams 2003]. In 1978, Ian Young and Hugh Clow collected the first MRI data of a human brain [figure 2.2].

The first clinical MRI systems were installed in the early 1980s, giving birth to a long series of methodological innovations and offering clinicians a pivotal tool to explore the human body non-invasively. As proof of NMR and MRI research's significant impact on physics, the Nobel Prize in Physics was delivered to Rabi in 1944, Bloch/Purcell in 1952, and Lauterbur/Mansfield in 2003.

2.2 Fundamentals of magnetic resonance imaging

Various imaging modalities exist using nuclear magnetic resonance for multiple purposes. Anatomical MRI mainly relies on exploring the differences in the relaxation of the nuclear magnetization between tissues to create some contrast mechanisms revealing their nature. Diffusion MRI exploits the movement of water molecules within tissues to probe their micro-architecture and has been widely used to explore brain connectivity. However, MRI is not limited to imaging the anatomy and can be used to probe dynamic processes, such as the blood circulation in the vascular tree using perfusion MRI or the measure of brain activity using functional MRI relying on the fact that cerebral blood flow and neuronal activation are coupled. Finally, NMR can also probe the chemistry occurring in the brain with NMR-spectroscopy able to provide insights about the brain metabolism involving various chemical processes.

In this section, we will start with an overview of the principle of magnetic resonance imaging.

2.2.1 Nuclear Magnetic Resonance Principle

The NMR phenomenon can be observed in any nucleus that has a non-zero spin, a quantum-mechanical property conveying an intrinsic form of angular momentum. Atoms composed of an odd number of nucleons, such as hydrogen, fluorine, sodium, and phosphorus (1 H, 19 F, 23 Na, and 31 P, respectively) possess a net dipole moment that makes it possible to precess or "spin around itself", thus inducing a magnetic field. Body tissues are mostly made of water and fat compounds, so they are extremely rich in hydrogen atoms 1 H, whose nucleus consists of a unique proton with a $\frac{1}{2}$ spin. For this reason, most MRI applications focus on proton imaging. In the case of this thesis, we will focus only on proton MRI imaging, but imaging with other non-zero spin atoms exists and is called X-nuclei imaging.

We will describe the nuclear magnetic resonance phenomenon from its three consecutive steps: polarization, resonance/excitation, and relaxation.
Polarization - Magnetic properties of protons can be described both with classical and quantum mechanics. In the case of classical and intuitive mechanics, the angular momentum or spin S of each proton can be envisioned as the consequence of the rotational motion of each proton around an axis. Protons are electrically charged bodies; this rotation creates a magnetic momentum µ, leading to [equation 2.1]:

$$\mu = \gamma S, \tag{2.1}$$

with γ the gyromagnetic ratio; $= 42.75$M Hz/T for the proton. In the absence of a magnetic field, each magnetic moment µ is randomly oriented, such that at the level of a grasp of protons $\mu = 0$ [figure 2.3]. Following quantum mechanics, the $\frac{1}{2}$ spin of a proton is either up or down. With this in mind, when an external static magnetic field B_0 is applied to protons, their magnetic moment spins align to B_0 in a parallel or anti-parallel way corresponding to respectively a low energy or high energy level and follows a precession movement around the field axis at the Larmor angular frequency ω_0, defined as:

$$\omega_0 = \gamma B_0. \tag{2.2}$$

The nuclei follow this B_0 polarization which results in a net magnetization vector M. At equilibrium in a static field, the net magnetization vector M is equal to M_0, which is parallel to B_0 (set along the +Z direction). M_0 is not measurable since its strength is minimal compared to B_0. M would become measurable if it was tilted in the transverse XY plane, resulting in a non-null transverse component M_{xy}. This leads to the next step of MRI: spin excitation and resonance.

Excitation - Spins can be flipped by the application of a rotating magnetic field, denoted B_1, of rotation frequency ω_r in the transverse plane [figure 2.3]. This tilting is caused by a transfer of energy between B_1 and the spins when the resonance condition $\omega_r = \omega_0$ is met, ω_0 being the Larmor frequency of the spins. At the quantum level, it results in a change in the energy levels of spins from the low-energy "parallel" state to the high-energy state at 180° or 90°. Because Larmor frequency lies in the range of radio frequencies, B_1 field is called a radio frequency (RF) pulse. The tissue magnetization's tilt angle depends on the RF wave's intensity and duration. For a simpler mathematical representation, studying the tilt in an XYZ frame rotating at the precession frequency is common.

Relaxation - When the B_1 field is stopped, the net magnetization M return to its equilibrium state aligning with B_0. This process is called relaxation, and it is accompanied by the emission of energy in the form of RF waves, which correspond to the signal recorded by the MRI system. It follows two phenomena obeying very different mechanisms: longitudinal relaxation, which corresponds to the regrowth of the longitudinal component, and transverse relaxation, which corresponds to the decay in transverse magnetization.

- Longitudinal relaxation - It's the return of excited spins to their low-energy (parallel) level. This is spin-lattice relaxation. The tissue in which the spins are embedded behaves like a viscoelastic medium, allowing the return of the longitudinal magnetization to its equilibrium value with a rapid progression at the beginning and then damped as equilibrium is approached, thus following exponential decay characterized by the time T1 relaxation time ([figure 2.4], [equation (2.3)]). T1 is the time the longitudinal magnetization takes to return to 63% of its final value and is of the order of 200 to 3000 ms for a field of 1.5 T. It is characteristic of tissues and depends on the mobility of the molecules in which the protons are engaged thus T1 relaxation time is shorter for large molecules.

$$M_z(t) = M_0 e^{(1\ -\ \frac{t}{T_1})} \tag{2.3}$$

- Transverse relaxation - It is due to spin dephasing linked to spin-spin interactions, which create field heterogeneities and, therefore, a discrepancy in precession frequencies. As the spins do not have the same frequency, they rapidly dephase, resulting in an exponential decrease of the transverse magnetization M_{XY} characterized by the $T2$ relaxation time taken for the transverse magnetization to return to 37% of its initial value ([figure 2.4] and [equation (2.4)]). Transverse relaxation is much faster than longitudinal relaxation: $T2$ is always shorter than or equal to $T1$ by almost an order of magnitude. In practice, this decay is faster than expected because of further local magnetic field heterogeneities (caused by air/tissue interfaces or paramagnetic materials) yielding a $T2^*$ relaxation time, such that $T2^* < T2$ [equation (2.5)]. The spin echo sequence (presented further on) stops $T2^*$ relaxation time, giving access to the true $T2$ value characterizing the tissue.

$$M_{xy}(t) = M_{xy}(0) e^{-\frac{t}{T_2^*}} \tag{2.4}$$

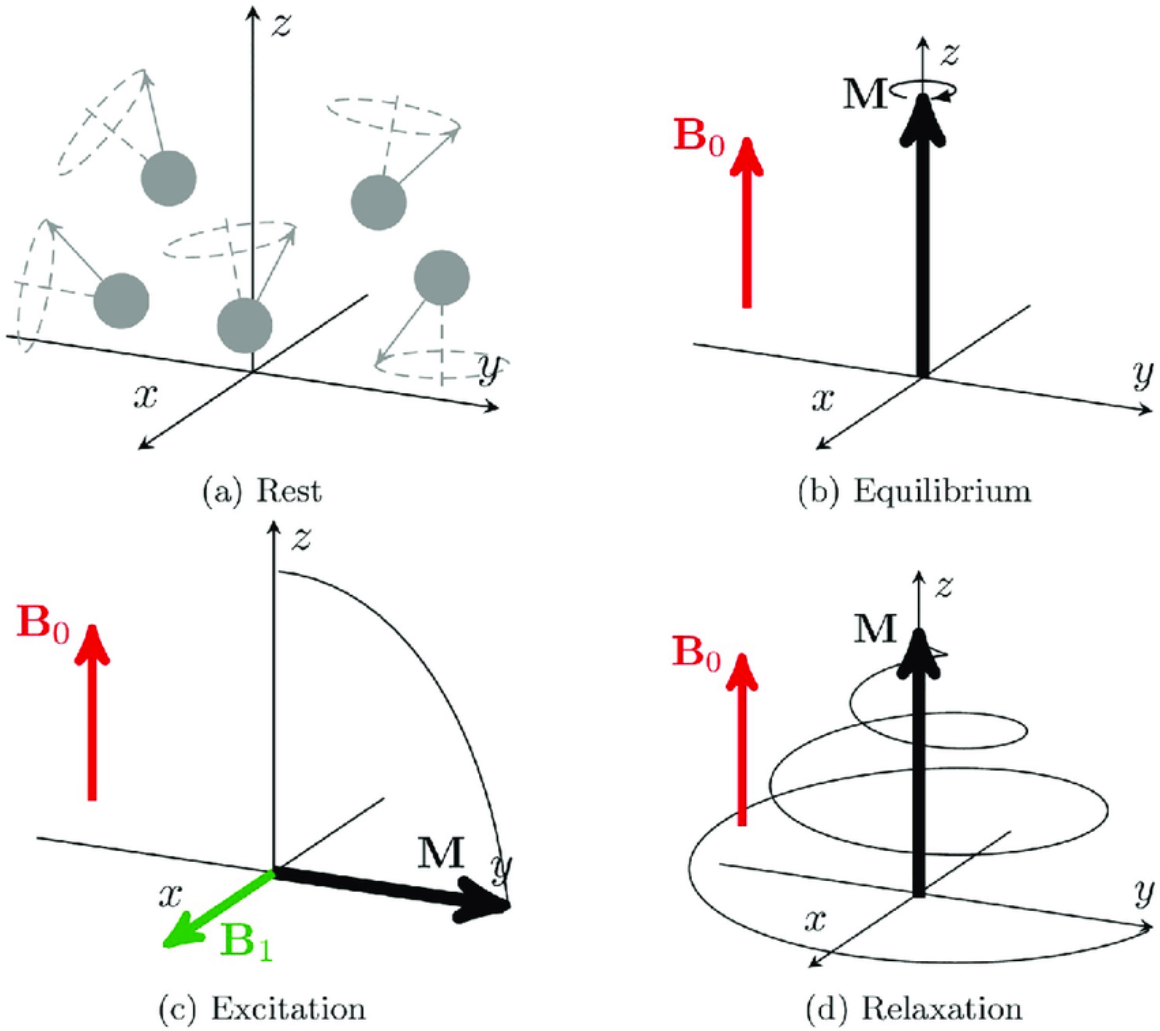


Figure 2.3: *Illustration of the NMR experiment from [Puiseux et al. 2021]. (a) When no magnetic field is applied, the spins are randomly oriented. When a $\boldsymbol{B}_0$ field is applied along the z-axis, all spins precess around the z-axis and (b) an equilibrium magnetization arises (polarization), oriented along the same axis. The equilibrium magnetization is shifted towards the transverse xy-plane by the effects of an RF-pulse ($\boldsymbol{B}_1$) applied at resonance frequency (c). When the RF excitation is released, the magnetization relaxes towards its equilibrium value (d) with a precession frequency that depends on the magnetic properties.*

$$\frac{1}{T2^*} = \frac{1}{T2} + \frac{1}{T2'} \tag{2.5}$$

The signal resulting from the transverse magnetization is called "Free Induction Decay" [figure 2.4] and oscillates at ω_0, with an exponential decay driven by the $T2^*$ relaxation time that can be acquired using a reception coil placed perpendicular to the transverse plane within the MRI scanner. The corresponding MRI signal is a solution to the phenomenological Bloch equations [Bloch 1946], describing the temporal evolution of the three spatial components of the magnetization M as a function of the static field B_0. With a 90 degrees RF pulse applied in the transverse plane, the solution to Bloch's equations is:

$$\begin{cases} M_x(t) = M_0 e^{-\frac{1}{T2}} sin(\omega t) \\ M_y(t) = M_0 e^{-\frac{1}{T2}} cos(\omega t) \\ M_z(t) = M_0 (1 - e^{-\frac{1}{T1}}) \end{cases} \tag{2.6}$$

[Equation 2.6] provides the temporal evolution to the net magnetization of spins after polarization.

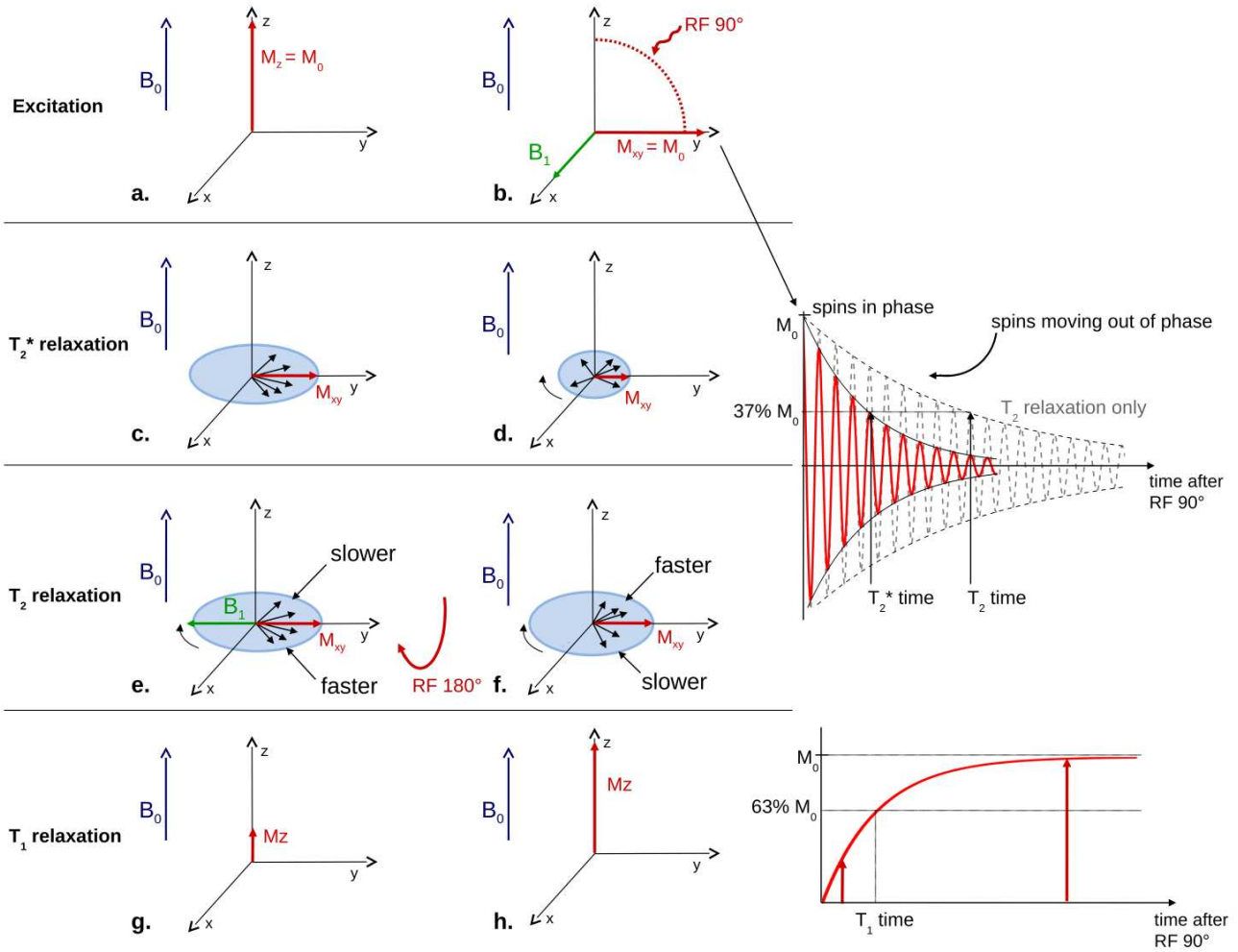


Figure 2.4: *Diagram showing the process of the excitation-relaxation phenomenon after a 90° RF pulse is applied at equilibrium. a) The net magnetization* $\boldsymbol{M}_0$ *(red arrow) is at equilibrium, aligned with the static magnetic field* $\boldsymbol{B}_0$*. b) When a 90° RF pulse is applied (*$\boldsymbol{B}_1$*), the longitudinal magnetization* $\boldsymbol{M}_z$ *becomes null and* $\boldsymbol{M}_0$ *lies entirely in the XY plane. c) & d) The net transverse magnetization decays as the spins (shown by the small black arrows) move out of phase. The overall term for the observed loss of phase coherence (dephasing) is* $T2^*$ *relaxation, which combines the effect of* $T2$ *relaxation and additional dephasing caused by local inhomogeneities in the magnetic field. e) % f) The additional decay caused by field inhomogeneities can be reversed by applying a 180° refocusing pulse. The resultant decaying signal is the Free Induction Decay (FID). g)& h) The longitudinal magnetization gradually recovers to its equilibrium value. The recovery of* $\boldsymbol{M}_z$ *is an exponential process with a time constant* $T1$*. Courtesy of Cyril Poupon*

2.2.2 Image encoding and reconstruction

This section explains how to transform NMR signal into imaging with the addition of spatial encoding to the NMR signal.

2.2.2.1 2D image encoding

The term 'encoding' used here reflects the fact that the spatial information obtained is not in the form of direct spatial coordinates (such as (x, y, z)). Spatial encoding is based on the successive application of magnetic field gradients [figure 2.5] to tag the magnetization coming from any point of the acquired volume.

Slice selection - The first step in spatial encoding is to select the target slice plane. A magnetic field gradient, called the Slice Selection Gradient (GSS), is applied in a direction perpendicular to the plane of interest. This field induces a field variation linearly depending on the slice position, which adds to B_0. Due to the Larmor equation, this linear field variation induces a linear precision frequency. At the same time, B_1 tilting magnetic field is applied using a ω_r frequency corresponding to the resonance of the protons in the plane of interest. Thus, tilting of the magnetization is only achieved for the protons belonging to this plane [figure 2.5]. All the hydrogen nuclei located outside the cut plane remain unexcited and will therefore provide no contribution to the signal since an RF pulse cannot be a pure harmonic (which would require it to have an infinite duration); it covers a certain bandwidth, which depends on the shape of the pulse and its duration [IMAIOS 2024].

The slice thickness can, therefore, be adjusted by tuning the RF bandwidth of the selective pulse and the amplitude of the slice selection gradient:

- Using a fixed amplitude gradient, the wider the bandwidth, the greater the number of protons excited by the pulse and the thicker the slice.
- Using a fixed bandwidth, the larger the gradient magnitude, the more significant the variation in precession frequency along the direction perpendicular to the selected plane, and the thinner the slice.

Phase encoding - The second step involves applying a phase encoding gradient (GPE) within the selected plane. When applied, it modifies the precession frequencies of the spins, inducing a phase shift that persists when interrupted. All protons precess at the same frequency but with different initial phases. More precisely, protons located on the same line, perpendicular to the direction of the gradient, will all have the same initial phase linearly depending on the line position in the selected plane.

Frequency encoding - The final stage of spatial encoding involves applying a frequency encoding gradient (GFE) in the third direction while the signal is collected. This modifies the precession frequencies in the GFE direction for the duration of its application. This creates columns of protons, precessing at the same frequency. The frequency information is included as this gradient is applied simultaneously with the signal collection.

While spatial frequency encoding is performed in a few milliseconds during signal reading, the spatial phase requires repeating the imaging sequence. In a conventional spin echo sequence, a single phase encoding step is performed per repetition, with TR repetition times of up to 3 seconds. All the signals within the same slice define what is called a K-space. The time required for the acquisition of an entire volume of N_S slices, with N_{PE} phase encoding lines, is given by the following equation:

$$T_A = TR * N_{PE} * N_S \tag{2.7}$$

With T_A the acquisition time, TR the repetition time, N_{PE} the number of phase encoding steps and N_S the number of slices composing the 3D volume.

Filling K-space The phase and frequency spatial encoding corresponding to the k_x and k_y k-space coordinate results in fact in the construction of a Fourier k-space that can be inverted to get back to the spatial domain using a 2D Fourier transfer [figure 2.6]. Therefore, The final image is reconstructed by applying an inverse 2D Fourier transform to the K-space. Low spatial frequencies near the K-space center contain contrast and general shape information. High spatial frequencies located at the periphery of the K-space correspond to the higher spatial resolution of the image, as they correspond to the abrupt changes in grey levels, which means the contours. The higher the spatial frequency of information, the better the spatial resolution.

2.2.2.2 3D image encoding

Whereas 2D imaging consists of a succession of selective slides, 3D imaging consists of selecting a full 3D volume and adding a further encoding mechanism to tag the position in this third direction. Here are the specificities of such acquisitions:

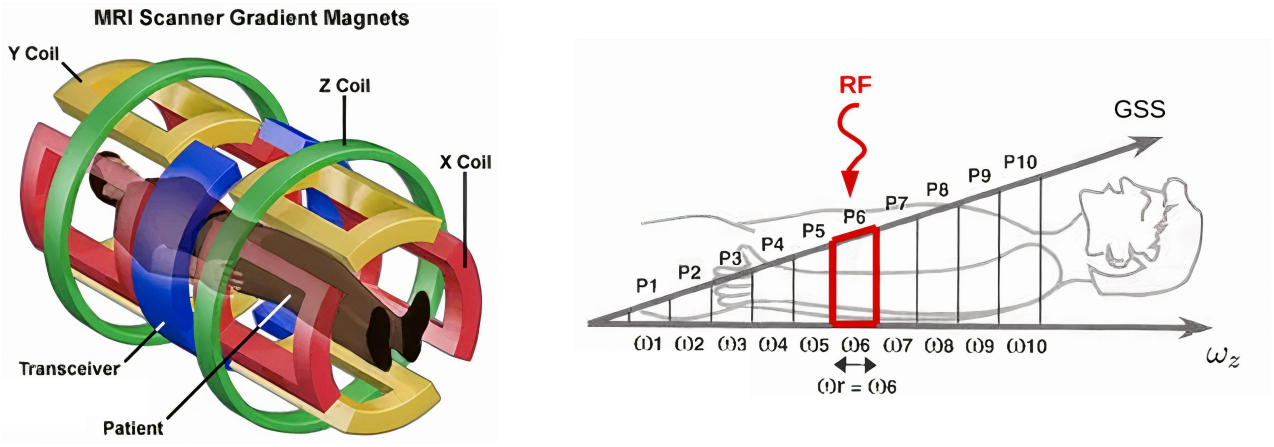


Figure 2.5: *On the left side, gradient magnets representation, adapted from [Unil 2024]. On the right side, slice selective excitation of a human torso using an RF pulse applied concomitantly with a Slice Selection Gradient (GSS) along the Z direction. The frequency increases from ω1 to ω10 in each plan P perpendicular to the Z-axis. If $\omega_r = \omega_6$, only the protons in P6 will be excited and contribute to the MR signal [Kastler and Vetter 2018].*

- The slice thickness the GSS selects corresponds to the entire volume.
- The third spatial dimension is encoded using another phase gradient.
- The number of repetitions changes by a factor equal to the number of cuts/partitions in the third dimension to fill the entire 3D K-space.
- A 3D Fourier reconstruction replaces the 2D Fourier transform reconstruction.

This 3D image encoding has several consequences:

- Given the large amount of information to be acquired to fill the 3D K-space, TR (gradient echo type) are used to avoid motion during the whole 3D acquisition of the K-space.
- At each repetition, the signal comes from the whole volume, thus enhancing the signal-to-noise ratio. Partitions can, therefore, be chosen to be thinner than conventional 2D slices.
- The entire volume of interest is explored, with no spacing or gaps between sections, which can be reformatted into planes.
- Because of the two-phase and positive encoding mechanisms, folding, and truncation artifacts can be observed along these two directions.

The following equation gives the acquisition time TA required to acquire a 3D image:

$$TA = TR * N_{Py} * N_{PE} * N_{PZ} \tag{2.8}$$

With N_{PE} and N_{PZ} respectively, the number of encoding steps along the phase and partition directions.

2.2.3 Conventional MRI sequence schemes

We have described how a volume is acquired, slice by slice or all at once, by applying specific gradients to perform an image frequency and phase encoding. The combination of gradients and radio

frequencies form an MRI sequence. Indeed, like in a music sheet, the MRI sequence orchestrates several events aiming to manipulate the magnetization that creates specific image contrasts, enabling the distinction of various tissues. The echo time (TE) and repetition time (TR) need to be adequately tuned to emphasize these contrast mechanisms. TE corresponds to the time elapsed between spin excitation and signal acquisition, and TR is the delay between two sequence kernel applications corresponding to an MRI pattern repeated during acquisition. The spin and gradient echo sequences, two fundamental MRI sequence schemes, are presented hereafter.

Spin echo sequence - Historically, spin echo was the first sequence introduced to perform imaging. It consists of a starting 90° RF pulse followed by a 180° rephasing pulse at $TE/2$, with a signal readout at TE [figure 2.6]. Each repetition of this pattern fills a line in the K-space, each time with a different phase encoding. The 180° rephasing pulse is used to compensate for the constant field heterogeneities and to obtain a $T2$-weighted echo rather than a $T2^*$-weighted echo [equation (2.5)]. To reflect $T1$ relaxation, a short TR is reacquired to forbid the longitudinal magnetization regrowth, which depends on $T1$. The decrease in TR will weigh the image in $T1$ as differences between tissues will be highlighted. To reflect $T2$ relaxation, a long TR is required to recover the longitudinal magnetization fully. During the next flip, the influence of $T1$ on the magnitude of the signal will be minimized. Combined with a long TE, the different tissues will be better highlighted according to their $T2$. Tissues with long $T2$ will show a hypersignal, whereas structures with short $T2$ will show a hyposignal. The proton density spin echo sequence minimizes the influence of both $T2$ and $T1$ by combining short TE and long TR. The major disadvantage of $T2$-weighted spin echo sequences is their long TR, which results in prohibitive acquisition times. While $T1$-weighted spin echo sequences can be used in clinical practice, providing good-quality anatomical imaging, other types of faster sequences are preferred for $T2$ weighting.

Gradient echo sequence - The gradient echo generally relies on an excitation tilting the magnetization. The sequence differs from spin echo because the flip angle is lower than 90° and there is no 180° rephasing RF pulse. A flip angle of less than 90° reduces the amount of magnetization tilted. The return to equilibrium occurs more quickly because residual magnetization is still along M_z. Thanks to these modifications, TR and TE's can be significantly reduced. Thus, the main advantage of gradient echo is the time saved, which allows 3D imaging to be carried out. Unlike spin echo, constant field heterogeneities are not compensated for in gradient echo by a 180° pulse. The signal obtained is $T2^*$-weighted rather than $T2$-weighted. Gradient echo sequences are more sensitive to magnetic susceptibility artifacts than spin echo sequences.

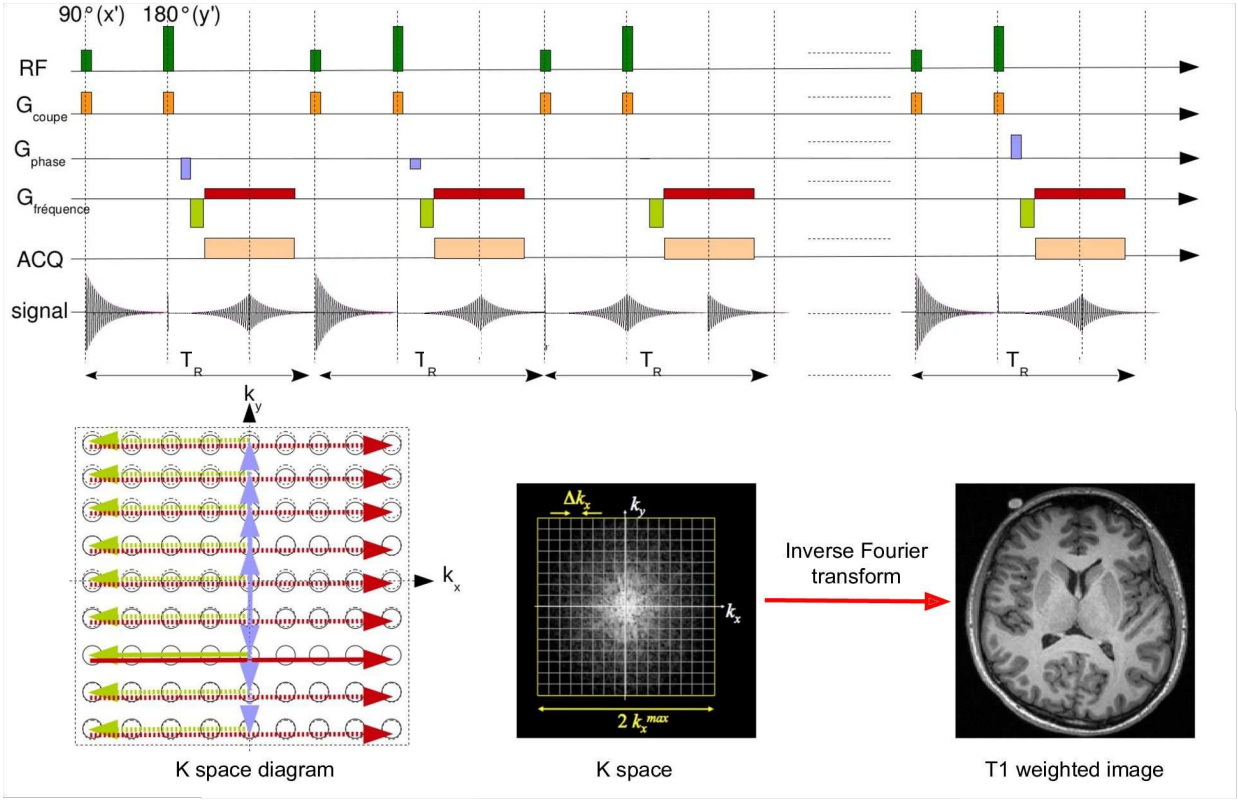


Figure 2.6: *Up: 2D acquisition spin echo sequence diagram. Bottom left: K-space filling procedure. Bottom right: final image reconstruction example. Courtesy of Cyril Poupon.*

2.3 Diffusion MRI

Diffusion MRI (dMRI) is an imaging technique sensitive to the local displacement of water molecules within biological tissues. This has proven to be a great tool over the years for the early diagnosis of brain stroke, the study of brain structural connectivity, and a precious tool for pre-operative planning in neurosurgery [Johansen-Berg and T. E. Behrens 2013; Jones 2010; Chauvel et al. 2023]. During this section, we will first present the diffusion process phenomenon and then describe how to make MRI receptive to it. Finally, we will show the diffusion process models for dMRI.

2.3.1 Brownian Motion

At the microscopic scale, particles follow a random motion following a thermal equilibrium, defined by a given temperature [figure 2.7]. It was called Brownian motion after the botanist Robert

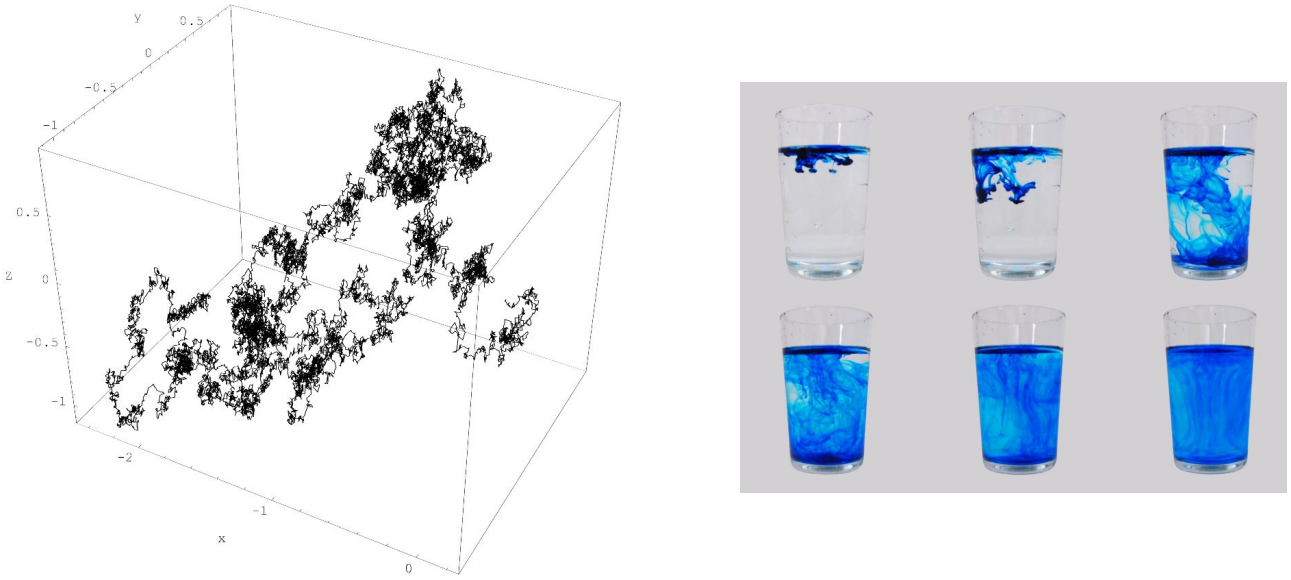


Figure 2.7: *On the left, a 3D simulation of Brownian motion, adapted from [CultureMath 2024]. On the right are the different diffusion experiment steps, adapted from [ScienceSparks 2024].*

Brown, who first described the phenomenon in 1827 while looking through a microscope at the displacement of pollen grains immersed in water [R. Brown 1827]. In 1855, Fick was the first to demonstrate that the existence of a concentration gradient of molecules led to a diffusion process [Fick 1855].
The first and second Fick's law are expressed respectively as:

$$\mathrm{J} = -D\nabla n(\mathrm{r}, t) \tag{2.9}$$

$$\frac{\partial n}{\partial t} = D\nabla^2 n, \tag{2.10}$$

with J the diffusion flux (amount of substance crossing a unit area per unit time), D the diffusion coefficient, which measures the speed of molecules (diffusivity), r and t respectively the space and time variables, n the liquid concentration.

The diffusion equation is the same as the heat equation proposed in 1827 by Joseph Fourier [Fourier 1827], Fick used it without having proof that the Brownian diffusion process is involved. It was 60 years later that Einstein recognized that Brownian motion was associated with diffusion [Einstein 1905], the flux of particles arising from a gradient in concentration. In the case of Brownian motion, however, no macroscopic concentration gradient is needed, as the molecules undergo a process of "self-diffusion" arising from local concentration fluctuations. Albert Einstein rewrote Fick's laws for the diffusion of molecules in a concentration gradient in terms of diffusion under probability gradients. This step enabled a description of Brownian motion as a stochastic process. Here is the rewriting of the diffusion law proposed by Einstein:

$$\frac{\partial P(\mathrm{r}|\mathbf{r}', t)}{\partial t} = D\nabla^2 P(\mathrm{r}|\mathbf{r}',\mathrm{t}) \tag{2.11}$$

$$\mathrm{J} = -D\nabla P(\mathrm{r}|\mathbf{r}', t), \tag{2.12}$$

with P(r|$\mathbf{r}'$,t) the conditional probability that a particle starting at r at time zero will move to $\mathbf{r}'$ after a time t. P(r|$\mathbf{r}'$,t) has several names in the literature as diffusion propagator, diffusion function, or displacement distribution function.

In an unrestricted medium, the distribution of the displacements becomes Gaussian with a distribution width expanding over time. The Gaussian nature of the conditional probability for self-diffusion, represented by [equation 2.11] leads to Einstein's equation:

$$< (\mathrm{r} - \mathbf{r}')^2 >= 6Dt, \tag{2.13}$$

where <...> stands for an average over spins ensemble. We can define the diffusion coefficient as:

$$D = \frac{1}{3}\int_0^\infty < \nu(\tau)\nu(0) > d\tau, \tag{2.14}$$

where $< \nu(\tau)\nu(0) >$ is known as the auto-correlation function of the molecular velocity ν. The correlation time of this function is defined as :

$$\tau_c = \int_0^\infty \frac{< \nu(\tau)\nu(0) >}{\nu^2} d\tau, \tag{2.15}$$

where τ_c gives a measure of the timescale over which the fluctuating molecular velocity becomes decorrelated and the mean-squared velocity $<\nu^2>$ is a constant for the ensemble, which is taken to be stationary.

Furthermore, if we substitute [equation (2.11) (2.12)] we obtain the diffusion propagator equation:

$$P(\mathrm{r}|\mathbf{r}', t)\left[\frac{\partial}{\partial t} - \nabla D\right] = \delta(\mathrm{r} - \mathbf{r}')\delta(t). \tag{2.16}$$

In the case of free diffusion without restriction in a homogeneous medium, the solution to this equation is given by the Gaussian propagator P_G:

$$P_G(\mathrm{r}|\mathbf{r}', t) = ((4\pi Dt)^3)^{-\frac{1}{2}} exp(-|\mathrm{r} - \mathbf{r}'|^2/(4Dt)) \tag{2.17}$$

In anisotropic media without Gaussian approximation, we shall find it necessary to define a diffusion tensor and rewrite differential equation [equation 2.1] by replacing D with D. D is known as the diffusion tensor and describes how the particle flux in any direction is related to the probability gradient in any direction. A principal Cartesian axis frame may be found where D is diagonal, with elements Dxx, Dyy, and Dzz. These elements may differ for anisotropic diffusion, whereas for isotropic diffusion, they are equal. We will describe the diffusion tensor later in detail.

Flow and dispersion - Until now, we have described self-diffusion from the motion of molecules. Nevertheless, the molecular motion is also governed by flow and dispersion [Jones 2010], especially in blood vessels or tissue through which fluids perfuse. Dispersion is the process whereby molecules that start together in the same neighborhood become separated due to translational motions. Like diffusion, dispersion involves stochastic processes that necessitate the language of statistical physics. In thermal equilibrium, and in the absence of fluid flow, Brownian motion alone will be sufficient to induce molecular separation; thus, in the rest of the thesis, we will rely only on Brownian motion from water molecules, thus consciously neglecting the other phenomena.

2.3.2 Sensitizing MRI to the diffusion process

Until now, we have described the essence of the diffusion phenomenon independently from its effect on the MRI signal. The following section explains how the NMR signal can be sensitized to

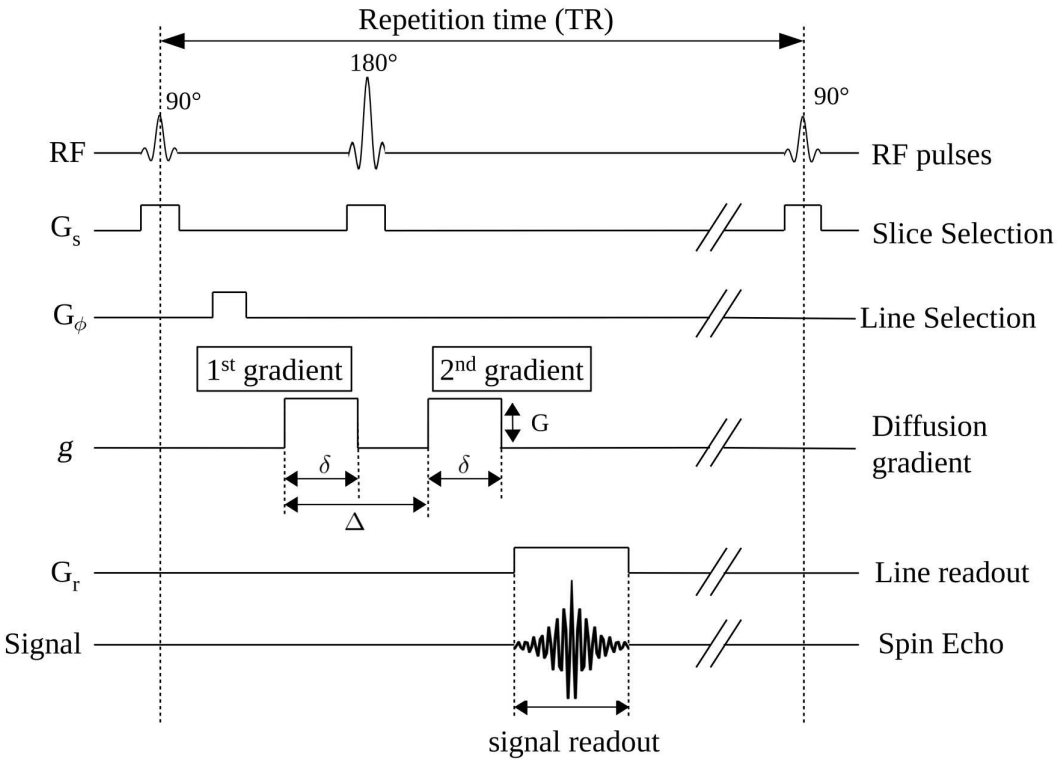


Figure 2.8: *Chronogram of a PGSE sequence. Adapted from [Beaujoin 2018].*

diffusion.

2.3.2.1 Pulsed gradient spin echo sequence

In 1963, McCall, Douglass, and Anderson suggested that a gradient might be used in the form of rectangular pulses inserted respectively in the dephasing and rephasing parts of the MRI echo sequence [McCall, Douglass, and Anderson 1963]. This pulsed gradient spin echo (PGSE) sequence was first demonstrated by Stejskal and Tanner (1965) [Stejskal and Tanner 1965]. PGSE modifies a spin echo sequence by applying two identical linear gradients of amplitude G and duration δ before and after the refocusing pulse. The two gradients are separated by a time interval Δ as shown in [figure 2.8]. The role of the first gradient is to tag each spin with a phase depending on its position. The second gradient does precisely the same but with an opposite sign. Suppose spins move between the application of the first gradient and the application of the second gradient. In that case, the second phase tag doesn't compensate for the first one, thus resulting at the voxel level to a signal decay [figure 2.9].

The expression of the diffusion attenuation can be written as follows:

$$S \propto S_0 e^{-bADC}, \tag{2.18}$$

with S the received signal, S_0 the signal value without diffusion sensitization (but including T2 relaxation), b-value the degree of sensitization to diffusion (in s/mm^2), and ADC is the apparent diffusion coefficient in the direction of the applied diffusion gradient. In the case of a free isotropic medium, the ADC remains the same whatever the diffusion direction. If the diffusion process is hindered or restricted in a given direction, then ADC will decrease along the direction.
In the case of rectangular gradient pulses, which is the ideal case, the b-value is defined as follows:

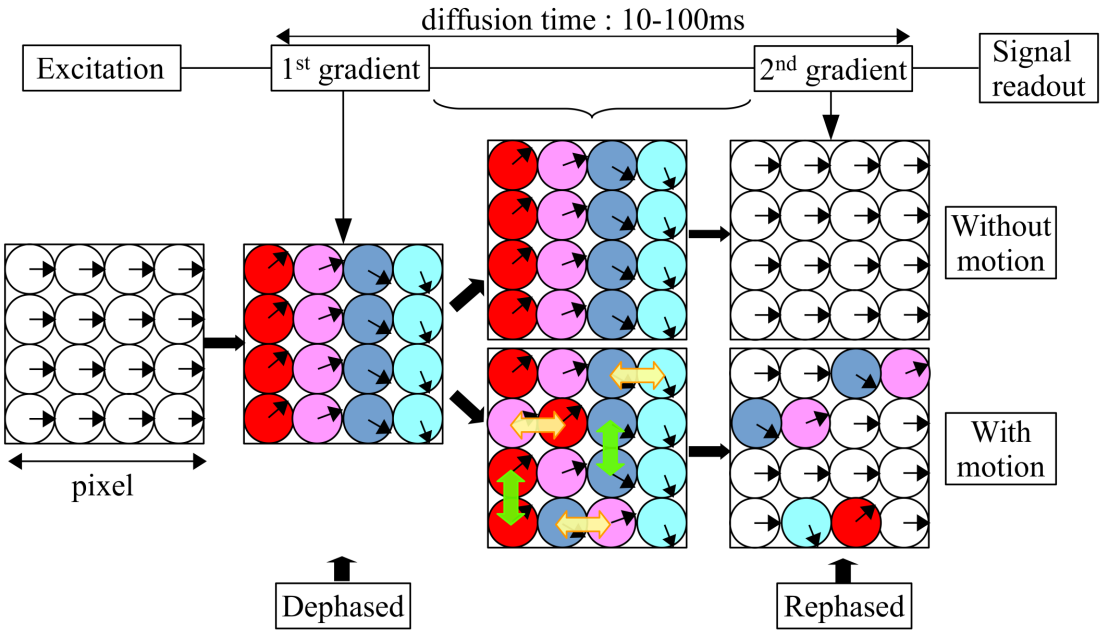


Figure 2.9: *The spin dephasing during the PGSE experiment: in the presence of a diffusive motion, spins are not rephased by the second diffusion gradient. This phase incoherence at the readout leads to a signal loss at the origin of the dMRI contrast. Adapted from [Beaujoin 2018].*

$$b = (\mathrm{G}\gamma\delta)^2(\Delta - \delta/3) = \mathrm{q}^2\tau \tag{2.19}$$

γ being the gyromagnetic ratio and q is called wave vector and is equal to $(\mathrm{G}.\gamma.\delta)$. If we consider the hypothesis of a Gaussian diffusion process, the signal follows the following equation driving attenuation in any direction:

$$S(\mathrm{q},\tau) = S_0 e^{-\tau(\mathrm{q}^T \mathrm{D}\mathrm{q})}, \tag{2.20}$$

with D a real, symmetric, and positive definite diffusion tensor that holds the information of the ADC value in any spatial direction.

2.3.2.2 Limitations

As mention earlier, [equation (2.19)] stands only in the case of Gaussian diffusion. The previous formalism cannot be used to characterize the non-monoexponential signal attenuation that is found at high b-values ($> 3000 s/mm^2$) for water in neuronal tissues [Beaulieu 2011]. [Figure 2.10] depicts different displacement distribution profiles of tert-butanol and water in neuronal tissue at different diffusion times, showing the drift of certain distributions from the Gaussian function.

Therefore, other formalisms must be used to represent a non-Gaussian displacement distribution function. For example, diffusion kurtosis imaging (DKI) analyses non-Gaussian water diffusivity using a higher-order polynomial model according to the following equation:

$$S = S_0 e^{-b*D_{app}+\frac{1}{6}*b^2*D_{app}^2*K_{app}}. \tag{2.21}$$

In comparison with the monoexponential formalism [equation 2.18], this equation yields two variables (D_{app} and K_{app}). The parameter K_{app} represents the apparent diffusional kurtosis (unitless), and

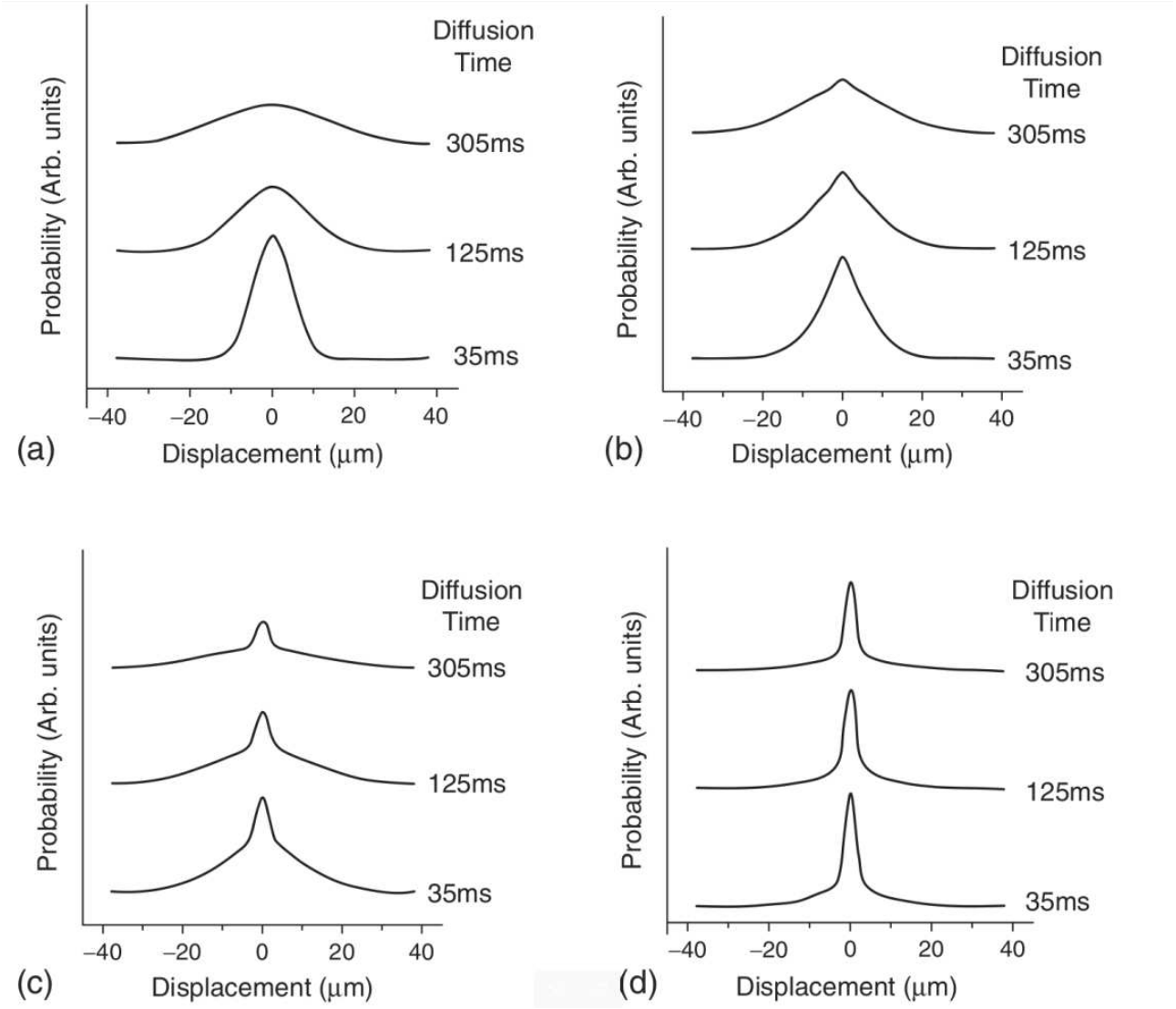


Figure 2.10: *Displacement distribution profiles for tert-butanol and water in neuronal tissue at different diffusion times. (a) A free diffusing reference sample of tert-butanol depicting the broadening of the displacement profile with increased diffusion time. Displacement profiles for excised rat brain (b) and an excised bovine optic nerve measured parallel (c) and perpendicular (d) to the long axis of the nerve show a mixed behavior of two components: a broad component that follows the tert-butanol behavior as a function of the diffusion time, and a narrow component that does not change its width with the increase in the diffusion time. Reproduced from [Assaf and Y. Cohen 2000].*

D_{app} is the diffusion coefficient that is corrected to account for the observed non-Gaussian behavior [Rosenkrantz et al. 2015].

Other formalisms exist; however, their analytical base form is restricted to simple geometries such as spheres, cylinders, or ellipsoids. Alternatively, as shown later, one can use the Fourier relation between the MR diffusion signal decay and the three-dimensional (3D) displacement distribution function to propose a non-Gaussian distribution formalism.

2.3.2.3 Introduction to the q-space

In [equation 2.19] we presented the wave vector q. We can draw here an analogy with the k-space (see section 2.2.2.1). Indeed, just like Fourier frequency K-space is fundamental for the final reconstruction of the image, the wave vector q is fundamental to draw a q-space that will allow the reconstruction of

a diffusion-weighted image. Completing the q-space using a certain model is called q-space sampling; the different q-space samplings are presented in [figure 2.15]. The different construction models of this q-space regarding diffusion signal acquisition will be detailed later.

2.3.2.4 Oscillating Gradients Spin Echo

Most diffusion MRI applications rely on PGSE, nevertheless, other diffusion MRI sequence schemes have been introduced for instance depending on "Oscillating Gradients Spin Echo" (OGSE). In their paper, [Gore et al. 2010] replaced the mono-polar trapezoidal pulses with oscillating diffusion gradients, whose frequency can be related to the adequate diffusion time in the experiment. Such sequences enable to access shortened diffusion times in comparison with PGSE sequences, thus increasing their sensitivity to smaller pore sizes. The ADC curve as a function of the oscillation frequency of the OGSE gradients varies as the pore diameter varies [figure 2.11]. In time, measuring this function may make it possible to identify the average diameter of the axons. The major drawback of OGSE sequences is that only small b-values can be reached on the current NMR apparatus due to gradient limitations.

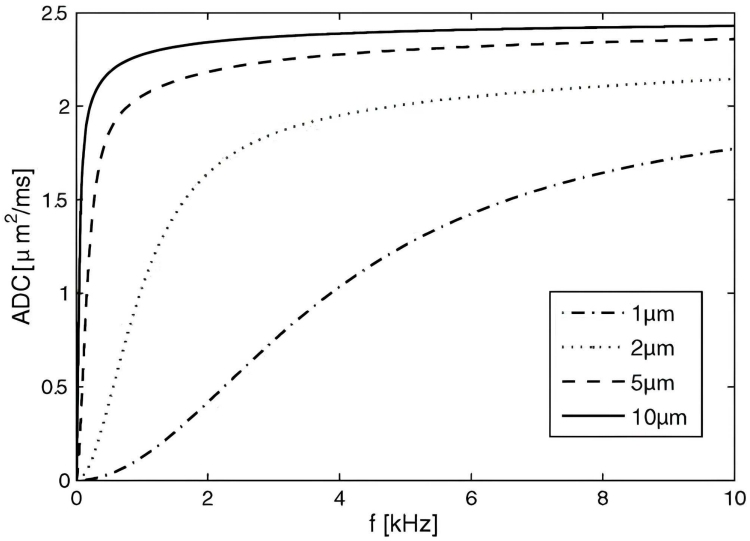


Figure 2.11: *Calculated values of the apparent diffusion coefficient (ADC) as a function of frequency for diffusion between two infinite impermeable planes for different spacings. The dispersion with increasing frequency is initially much more rapid for larger separations. Adapted from [Gore et al. 2010].*

Trapezoidal OGSE waveforms [figure 2.12] are usually preferred in practice since they offer stronger b-values while preserving the frequency selectivity [Van Nguyen et al. 2014], which is essential to measure the diffusion spectrum. Cosine-like waveforms enable getting rid of the null frequency peak of sine-like waveforms and are thus the sequence of choice for temporal diffusion spectroscopy.

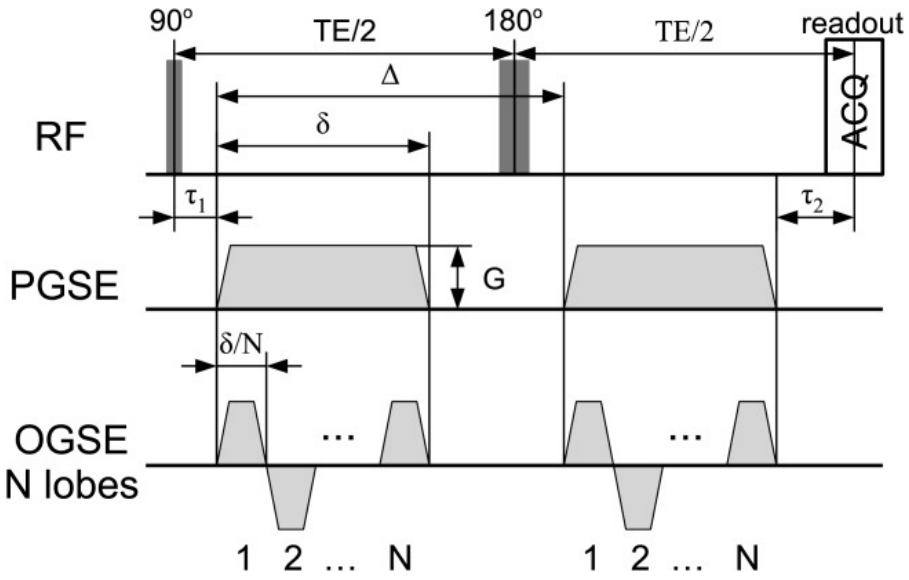


Figure 2.12: *An illustration of a PGSE (top) and an OGSE (bottom) sequence showing all the variables. The presented OGSE sequences have a trapezoidal shape with minimum achievable rise time to maximize diffusion weighting. The PGSE sequence is a special case of OGSE for N=1. Adapted from [Drobnjak, H. Zhang, et al. 2016].*

2.3.3 Models of the diffusion process

The development of models is motivated by the desire to synthesize the information contained in the many diffusion-weighted volumes acquired. Moreover, models can provide access to parameters characterizing the microstructure. Several representations of diffusion-weighted data exist. Two main groups can be drawn: model-based techniques that make physical assumptions about diffusion or tissue properties and model-free techniques that do not make these hypotheses.

2.3.3.1 Model-based techniques

Thanks to the pioneering work from Denis Le Bihan [Le Bihan et al. 1986] who first acquired diffusion-weighted MRI, helping the characterization of the tissues microstructure thanks to images obtained from healthy subjects and patients analyzed in terms of a mean diffusivity. This work on diffusion anisotropy took off with the introduction of the more rigorous formalism of the diffusion tensor [Basser, Mattiello, and LeBihan 1994]. With diffusion tensor imaging (DTI), diffusion anisotropy effects in diffusion MRI data can be fully extracted, characterized, and exploited, providing even more exquisite details of tissue microstructure.

Mathematically, following [equation (2.20)], the signal attenuation can be modeled using a symmetric, positive, and definite diffusion tensor:

$$\mathrm{D} = \begin{bmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{xy} & D_{yy} & D_{yz} \\ D_{xz} & D_{xy} & D_{zz} \end{bmatrix} \tag{2.22}$$

Since D is symmetric, only six parameters are unknown. Thus, if we add the acquisition to obtain the S_0 signal [equation (2.20)] a total of 7 acquisitions are required to solve [equation (2.22)] even if in practice, a few dozens of isotropically distributed gradient directions are used to become robust to the MRI data corruption. Each of these acquisitions, with a different gradient orientation, can count as a filling of the q-space mentioned earlier. The tensor can be visualized as an ellipsoid [figure 2.13]. Once D is estimated, its eigenvalues λ_1, λ_2, λ_3 with $\lambda_1 \geq \lambda_2 \geq \lambda_3$ and eigenvectors $\mathrm{e}_1, \mathrm{e}_2, \mathrm{e}_3$ can be computed, providing a series of rotation-invariant characteristics of the tissue microstructure:

- Mean Diffusivity or Apparent Diffusion Coefficient (ADC) - It represents the average diffusivity of water within the voxel and can be computed from the trace of the diffusion tensor D:

$$ADC = Tr(\mathrm{D}) = \frac{\lambda_1 + \lambda_2 + \lambda_3}{3} \quad in \quad m^2 s^{-1} \tag{2.23}$$

- Fractional anisotropy (FA) - It quantifies the degree of anisotropy of the diffusion process within the voxel ranging from 0 (isotropic case) to 1 (anisotropic case):

$$FA = \sqrt{\frac{3}{2}} \sqrt{\frac{(\lambda_1 - ADC)^2 + (\lambda_2 - ADC)^2 + (\lambda_3 - ADC)^2}{\lambda_1 + \lambda_2 + \lambda_3}} \tag{2.24}$$

- Radial and longitudinal diffusivities ($D_\perp / D_{//}$) - Radial diffusivity represents the diffusivity perpendicular to the mean axis of the tensor, whereas longitudinal diffusivity highlights the diffusivity along it:

$$D_\perp = \frac{\lambda_1 + \lambda_2}{2} \qquad D_{//} = \lambda_1 \qquad both \ \ in \ \ m^2 s^{-1} \tag{2.25}$$

Despite DTI being one of the first models developed in dMRI, DTI remains widely used for clinical applications thanks to its simplicity (dozen acquisitions) and robustness. Nevertheless, DTI suffers from severe limitations:

- Due to the mathematical representation of the tensor, only one principal direction can be decoded [T. E. J. Behrens et al. 2007]. As a result, DTI cannot account for multiple fiber populations crossing within the voxel, a typical case in white matter tissue, as seen in chapter one (see 1.2.3).
- DTI relies on a Gaussian hypothesis that cannot account for the multiple exponential decay observed when exploring higher b-values.
- Finally, DTI features lack specificity regarding the underlying microstructural changes in the tissues; for example, FA can decrease because of damaged myelin or decreased axonal density [Curran, Emsell, and Leemans 2016].

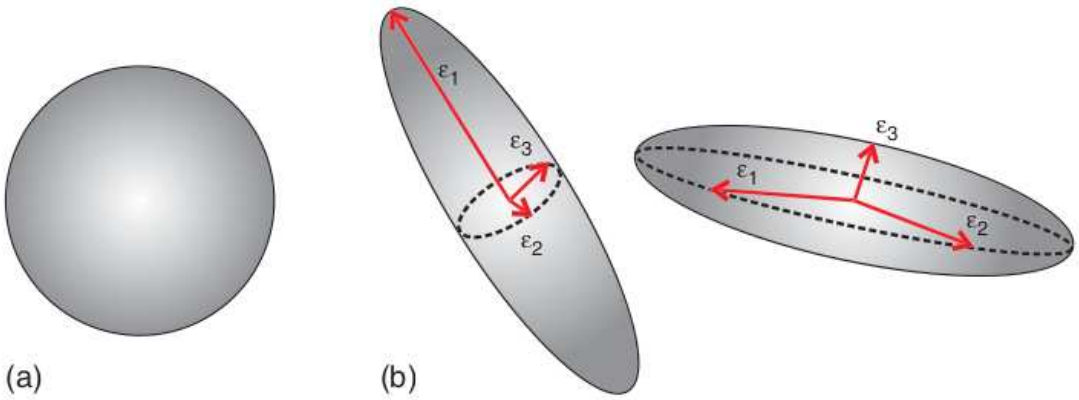


Figure 2.13: *Representation of the diffusion tensor as a sphere in the case of isotropic diffusion (a) and as an ellipsoid (b) for anisotropic diffusion behavior. The λ_1, λ_2, λ_3 and $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$ correspond to the eigenvalues and eigenvectors of the tensor respectively. Adapted from [Jones 2010].*

A plethora of alternative models have been proposed relying on the representation of the signal using specific distributions such as Wisharts, Bingham, Watson, or Von Mises-Fisher distributions [McGraw et al. 2006; Jian and Vemuri 2007]; or using higher order tensors like the Multi-Gaussian tensor model from Tuch [Tuch et al. 2002].

DTI's incapacity to solve fiber crossings led to the emergence of new model-free reconstruction techniques, based on the Callaghan fundamental Fourier equation linking the q-space and the displacement space.

2.3.3.2 Model free techniques

In the case of a model-free approach, we need to avoid the Gaussian diffusion distribution hypothesis. Several approximations can be made to describe diffusion signal attenuation in the case of non-Gaussian diffusion such as "Short Pulse Gradient" (SPG), "Gaussian Phase Distribution" (GPD) [Neuman 1974] or "Multiple Correlation Function" (MCF) [Grebenkov 2007]. SPG approximation is explained hereafter.

Short Pulse Gradient (SPG) - This approximation, also known as "Narrow gradient approximation" assumes that in [equation (2.19)] δ is infinitely smaller than Δ ($\delta << \Delta$), thus allowing to write the effect of the two gradients on a spin moving from r to r$'$ as a phase shift (γ δ G) * (r$'$ - r) contributing to the signal decay. We are defining E(G,Δ) the NMR normalized signal as follow:

$$\mathrm{E}(\mathrm{G},\Delta) = \frac{\mathrm{S}(\mathrm{G},\Delta)}{\mathrm{S}(0,\Delta)}, \tag{2.26}$$

where S(G,Δ) and S(0,Δ) are respectively, the $T2$ weighted MR signals with and without diffusion sensitivity.

E(G,Δ) is an ensemble average in which each phase term $\exp(i\gamma\delta \mathrm{G} * (\mathrm{r}' - \mathrm{r}))$ is weighted by the probability P(r)P(r|r′, Δ) for a spin to move from r to r′. Using this SPG approximation, Stejskal and Tanner [Stejskal and Tanner 1965] come up with the following equation:

$$E(\mathrm{G},\Delta) = \int P(\mathrm{r}) \int P(\mathrm{r}|\mathbf{r}',\Delta) exp(-i\gamma\delta \mathrm{G}.[\mathrm{r} - \mathbf{r}'])d\mathbf{r}'d\mathrm{r} \tag{2.27}$$

The phase shifts appearing in [equation (2.27)] depends only on the dynamic displacement, $\mathrm{R} = \mathbf{r}' - \mathbf{r}$. We can now introduce P(R,t) the probability that a particle moves by R over a time t called "average propagator" [Kärger and Heink 1983] thus yielding :

$$P(\mathrm{R},t) = \int P(\mathbf{r})P(\mathbf{r}|\mathbf{r}+\mathrm{R},t)dr \tag{2.28}$$

We can rewrite equation (2.20) as follow:

$$E(\mathrm{G},\Delta) = \int P(\mathrm{R},\Delta) exp(-i\gamma\delta \mathrm{G}.\mathrm{R})d\mathrm{R} \tag{2.29}$$

Using the q wave vector definition, [equation 2.29] can be modified using Fourier transform of E(G,Δ) and P(R,Δ):

$$E_\Delta(q) = \int P(\mathrm{R},\Delta) exp(-i2\pi \mathrm{q}.\mathrm{R})d\mathrm{r} \tag{2.30}$$

The SPG is a convenient hypothesis, nevertheless it is not valid in practice due to hardware limitations in gradient amplitudes, which impose a certain duration of the gradient application to reach a sufficient diffusion sensitization. For the in vivo case, the SPG approximation requires DW gradient pulse duration to be 1 ms or less [King et al. 1994]. This condition cannot be achieved for common clinical scanners, so the displacement distribution profile retrieved with SPG is relative rather than absolute numbers. However, violating this assumption may not be as problematic as might appear because even if quantitative measurements of displacement will be underestimated [Bar-Shir et al. 2008], the orientations themselves are unlikely to be affected [Wedeen et al. 2005].

SPG and q-space - From [equation 2.28], we can obtain the displacement distribution function by applying an inverse Fourier transform to the normalized diffusion-weighted MRI signal. To reconstruct a complete displacement distribution function, the diffusion-weighted normalized signal must be collected for a large set of q wave vectors, which corresponds to the sampling of the q space introduced by Callaghan [P. T. Callaghan et al. 1991]. This simple, model-free approach allows characterization and quantification of the displacement distribution function without the use of complicated mathematical procedures or hypotheses on the diffusion response of tissues diffusion. Once the displacement distribution function is reconstructed, a few quantitative parameters can be extracted: the mean displacement (measured as the width at half maximum of this function), the probability for zero displacement (P0, measured as the peak intensity of this function), and the kurtosis (measured as the deviation from the Gaussian distribution) [Assaf and Y. Cohen 2000][figure 2.14]).

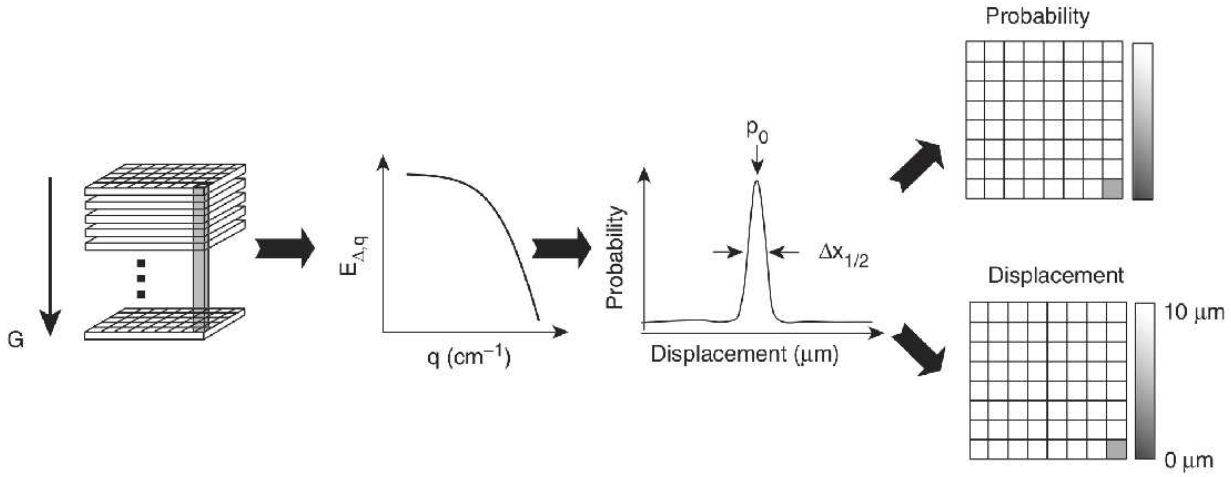


Figure 2.14: *Cartoon depicting the steps involved in obtaining the displacement and probability maps from a diffusion-weighted imaging (DWI) data set. The DWI data set is organized in a three-dimensional (3D) array. The normalized signal decay (EΔ,q) as a function of q is calculated for each pixel. The signal decay calculates the displacement distribution profiles by Fourier transform. Then, the two parameters characterizing the displacement distribution profile of each pixel are computed and collected into two sub-images. Reproduced from [Assaf and Y. Cohen 2000].*

In theory, propagator imaging reacquires an exhaustive sampling of the q-space:

- q magnitude must be sampled from 0 to an infinite value in theory.
- The diffusion time t_d must be large enough for the water molecules to explore their surrounding media, thus implying Δ to be larger than $\frac{l^2}{2D}$ with l being the size of the compartment and D being it apparent diffusion coefficient.
- The gradient pulse duration must be chosen such that the displacement of molecules remains much smaller than the compartment size l ($\delta << l$) [P. T. Callaghan et al. 1991].
- Finally, as mentioned earlier, SPG approximation is required, so that diffusion during the pulse is minimal, otherwise it will influence the extraction of the displacement distribution function [Mitra and Halperin 1995].

The q-space formalism described above can be expanded to q-space imaging (QSI), whose key motivation was to infer the compartment structural connectivity. The diffusion community introduced the concept of orientation distribution function (ODF) [Tuch et al. 2002], relying on the angular profile of the propagator. The ODF function ψ is described as follow:

$$\psi(\mathrm{u}) = \int_0^\infty P(\mathrm{ru})d\mathrm{r} \tag{2.31}$$

Mapping the ODF provides invaluable information about the tissue microstructure [Tuch et al. 2003]. A direct application of QSI is proposed with the diffusion spectrum imaging (DSI) method, described below.

Diffusion spectrum imaging (DSI) - Diffusion spectrum imaging is the direct three-dimensional (3D) imaging extension of q-space spectroscopy described until now, providing an estimate of the q-space

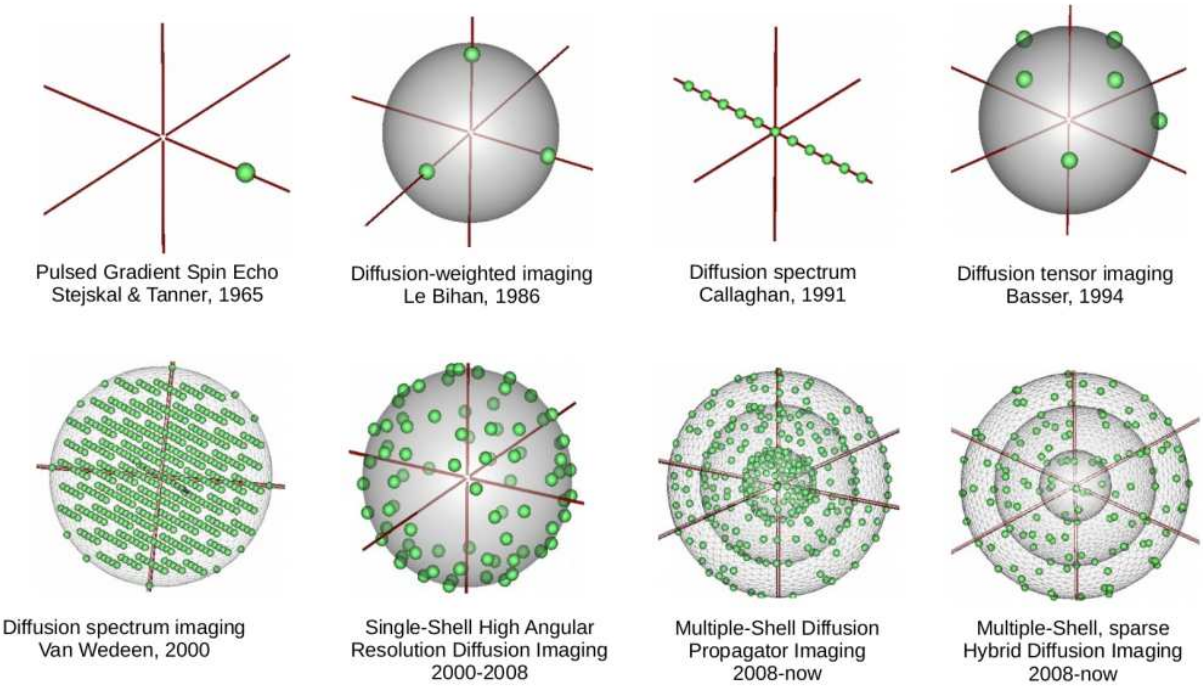


Figure 2.15: *Evolution of the q-space sampling schemes with the refinement of local models of diffusion MRI data. Courtesy of Cyril Poupon.*

spin propagator within each imaging voxel [Wedeen et al. 2005]. It requires a Cartesian q-space sampling acquisition, whereby a large number of DW images are collected. These q-vectors are arranged in a regular 3D Cartesian grid to obtain an adequate sampling of the DW signal as a function of position in q-space [figure 2.15]. After performing the requisite 3D Fourier transform, we obtain an estimate of the spin propagator within each imaging voxel. The fiber orientations are assumed to coincide with distinct peaks in the ODF, representing the directions of maximum diffusion. An illustration of DSI q-space sampling is presented in [figure 2.15].

DSI is considered the ultimate q-space sampling as it covers the full q-space 2.15. However, the most significant obstacle to its routine use is the large amount of required data that is incompatible with an acceptable scan duration. DSI has an inefficient sampling strategy: a complete 3D data set is acquired for each voxel, only to be projected down to a 2D ODF.
High angular-resolution diffusion imaging (HARDI) To tackle the limitations of the DSI method, the diffusion community has developed various approaches aiming at mapping the ODF from a simplified q-space sampling corresponding to a single sphere (or shell) of the q-space with a predefined radius corresponding to a single b-value ($b = \mathrm{q}^2\tau$, [equation 2.23]), introducing the concept of high angular resolution diffusion imaging (HARDI). Although the HARDI approach provides no information about the propagator, it does give the most concise characterization of its angular profile. With HARDI, the acquisition can be tailored to target a specific region of the q-space. With a complete 3D cartesian q-space acquisition like in DSI, a significant proportion of data samples will be located in low-SNR regions (high q-values) or low-contrast areas (low q-values), while with HARDI acquisition, it is possible to focus on an optimal b-value, resulting in an improvement of the contrast-to-noise ratio per unit of scan time.

The most popular HARDI reconstruction techniques are listed below:

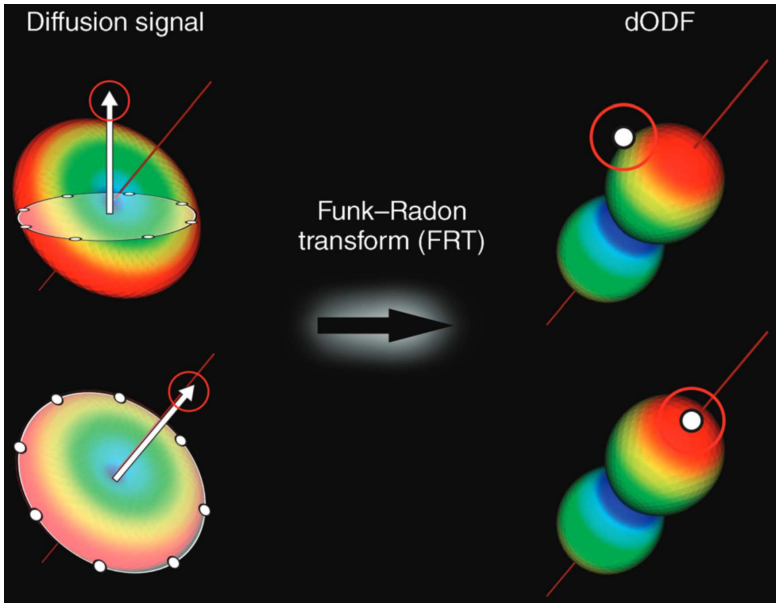


Figure 2.16: *Illustration of the Funk–Radon transform (FRT) at the heart of q-ball imaging and ODF reconstruction. Great circle integrals are computed from the measured signal to obtain the ODF. Adapted from [Descoteaux 2015].*

- Q-Ball Imaging (QBI) - Q-ball imaging was proposed as a way of directly obtaining an estimate of the diffusion ODF using HARDI acquisitions [D. S. Tuch 2004]. It relies on the Funk-Radon transform [Funk 1915] providing, in certain conditions, a good approximation of the projection of the propagator onto the orientation space [figure 2.16]. Fiber orientations can be extracted from QBI by finding the peaks in ODF and used to track through crossing-fiber regions [Campbell et al. 2005; Perrin et al. 2005; Berman et al. 2008].

QBI is fast and capable of producing results similar to DSI, with substantially reduced acquisition times. As with all q-space methods, large b/q-values are required to obtain adequate separation of the peaks in the ODF.

- Persistent Angular Structure MRI (PAS-MRI) - It provides an estimate of the persistent angular structure (PAS) of the diffusion propagator using HARDI data [Jansons and Daniel C Alexander 2003]. The PAS is defined as the angular dependence of the spin propagator, assuming its radial dependence is a function at a fixed radius r. In other words, spins are considered to diffuse at a fixed distance r, with an angular distribution given by the PAS.

- The Diffusion Orientation Transform (DOT) - The diffusion orientation transform also operates on HARDI data and provides an estimate of the spin propagator evaluated at a given radius R0 [Özarslan, Shepherd, et al. 2005]. To make the 3D Fourier transform tractable, the radial dependence of the DW signal is assumed to be monoexponential. With this assumption, the DOT can be performed efficiently, with the additional benefit that the spin propagator can be evaluated at any radius R0.

- Analytical Q-ball imaging (aQBI) - aQBI is an improvement of the QBI method relying on the decomposition of the signal acquired on a sphere into a spherical harmonics base [figure 2.17]. Whereas QBI relies on a Funk-Radon transform, aQBI relies on the Funk-Hecke transform [Descoteaux et al. 2007], adapted to the signal's spherical harmonic decomposition. aQbi has been proven to be 15 times faster than QBI, robust to noise, and more accurate in ODF maxima detection at the cost of slightly reducing angular resolution [Descoteaux et al. 2007].

- Constrained Spherical Deconvolution (CSD) - This method computes fiber ODF relying on the representation of the diffusion-weighted signal as the spherical deconvolution of a single fiber response function [Tournier et al. 2008]. CSD cannot be considered as a pure model-free technique because it relies on a preconception of a single fiber response, thus CSD is at the frontiers of model-free and model-based techniques.

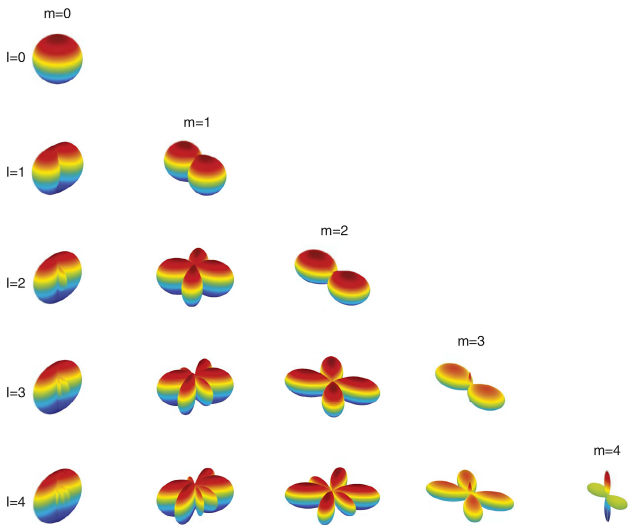


Figure 2.17: *Plots of the absolute value of the low-order spherical harmonic functions, with l the order value and m the index. Adapted from [Jones 2010] on page 462.*

2.4 Conclusion

In this chapter, we introduced the principles of magnetic resonance imaging with a focus on diffusion-weighted MRI being at the heart of this work. We tried as much as possible to avoid assumptions on the type of tissues concerned by the imaging. In fact, because of the historical link between MRI and neuro-imaging, the improvements of the models targeted white and grey matter brain tissue characterization. The next chapter describes the most common models established in the diffusion community, focusing on their assumptions to understand their limitations fully.

Chapter 3

Brain biophysical multi-compartment models

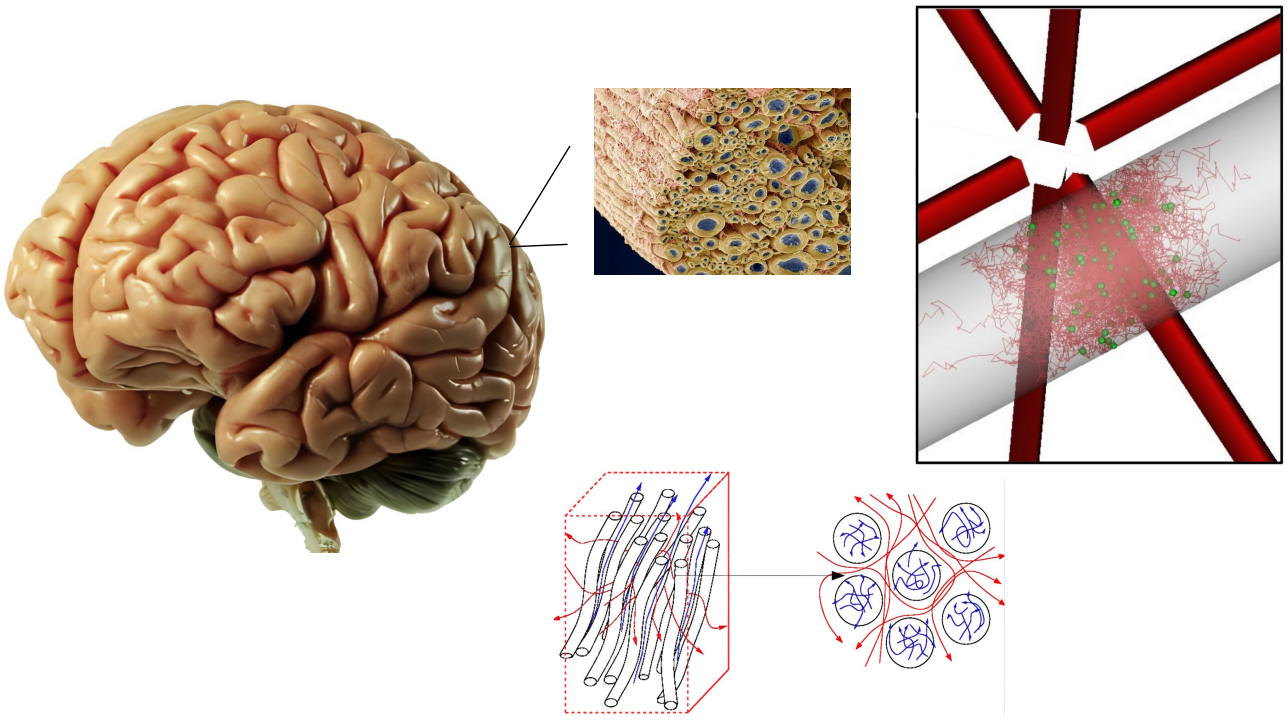


Figure 3.1: *Illustration of the hindered diffusion process within brain white matter tissues.*

Chapter outline

3.1 Brain tissues as restricted environment 58

3.2 Design of hardware microstructural phantoms 60

3.3 Multi compartmental analytical models 61

3.4 Computational models and dMRI simulator 67

3.4.1 Simulation of cell population geometries 67

3.4.2 Simulate the diffusion process using Monte-Carlo simulations 75

3.4.3 Synthesis of the dMRI signal 80

3.4.4 Computational model 81

3.5 Conclusion 81

In this chapter, we will present the state of the art of dMRI techniques developed to characterize the microstructure of brain tissues. Two main applications exist: first are local dMRI models computed at the voxel level to allow for specific reconstruction whole-brain tractograms [Mori et al. 1999; Poupon 1999]. In the last chapter, we saw some key elements concerning this application, notably through QBI and aQBI. The second application is the creation of biophysical models based on a geometrical model of the underlying tissue to solve the inverse problem of microstructure parameter estimation. Here, we will tackle only the second application as this thesis relies exclusively on it.

3.1 Brain tissues as restricted environment

Brain tissue microstructure affects the displacement of water molecules as demonstrated by [Chenevert, Brunberg, and Pipe 1990; Beaulieu and Allen 1994; Pierpaoli and Basser 1996]. The axonal fibers populating white matter bundles cause the observed average displacement to be smaller perpendicular to the fibers than parallel to the fibers [figure 3.2]. It was first thought that this observation does not contradict the Gaussian hypothesis of water diffusion in neuronal tissue, and the Stejskal-Tanner formalism was used to develop new frameworks to characterize the anisotropy of the diffusion process using the DTI model.

White matter (WM) tissue has a complex microstructure. It consists mainly of long axonal extensions of neurons, generally arranged into bundles, often represented as parallel tubular structures [figure 3.2]. WM also embeds glial cells surrounding and supporting the axonal fibers and microvasculature (see 1.2.3).
Biological membranes, such as cell membranes, are permeable to water molecules at various degrees, but it is expected to assume the existence of distinct water compartments to represent WM, each associated with a specific cell population or cellular space. Since the behavior of the diffusion process can be very different among these various compartments due to their geometrical and biophysical characteristics, diffusion MRI might be an adequate modality to probe the key features characterizing them, like size, density, or permeability.

Non-Gaussian displacement of water in brain tissue was reported in the mid-1990s [Beaulieu and Allen

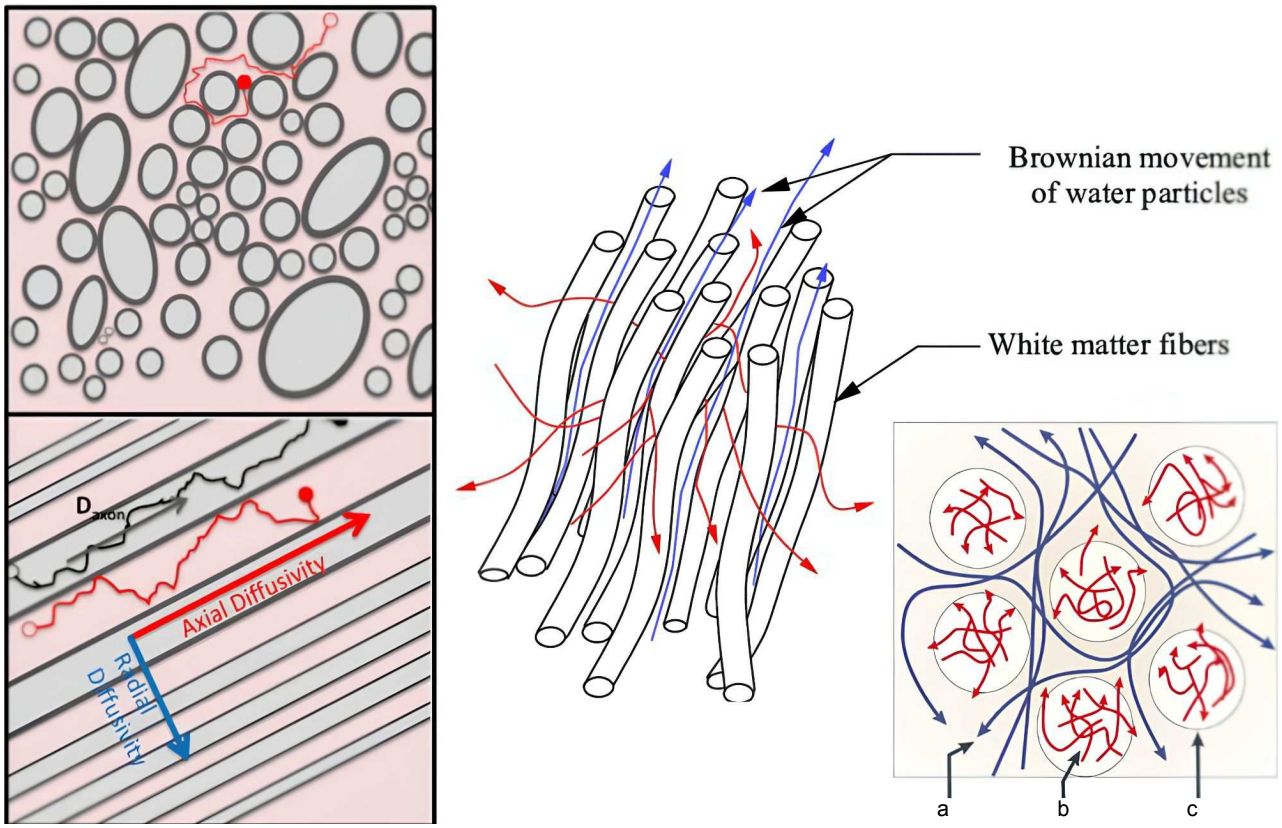


Figure 3.2: *Hindered water diffusion in white matter fibers. On the left, adapted from [Lazar et al. 2014]: Water diffusion in white matter is modeled using a two-compartment approach that assumes that the diffusion signal arises from intra-axonal (light gray) and extra-axonal (light pink) water. On the right, adapted from [Poupon 1999]: The restricted nature of diffusion inside white matter fibers or axons: Myelin, the axonal membrane, microtubules, and neurofilaments are all longitudinally oriented structures that could hinder water diffusion perpendicular to the length of the axon. On the bottom right from [Le Bihan 2003]: diffusion might be hindered by obstacles that result in tortuous pathways (a). Diffusion of molecules can be restricted in closed spaces (b), such as cells. Exchange between compartments also slows down molecular displacements (c).*

1994; King et al. 1994]. These experiments showed that water diffusion is at least partially restricted. The primary supporting experimental evidence was the finding that the relation between signal decay and diffusion tagging (b-value) is not mono-exponential when exploring a wide range of diffusion sensitivity, in other words, when high gradient amplitude and long diffusion times are reached. The non-mono-exponential signal decay may imply the existence of several diffusing components in the tissue and the effects of restricted diffusion. It was assumed that bi-exponential functions [Niendorf et al. 1996] then multi-exponential functions [Assaf and Y. Cohen 1998; Mulkern et al. 1999] were leading to a better signal decay representation.

Usually, the bi-exponential signal decay is associated with two components: a fast and a slow diffusing component. The fast-diffusing component is linked with water molecules in the extracellular space, whereas the slow-diffusing component may reflect water in the intracellular space. The fast component appears to be characterized by Gaussian diffusion, thus showing unrestricted or slightly hindered diffusion, whereas the slow component appears to be strongly restricted [Assaf and Y. Cohen 2000]. The main supporting experimental evidence comes from a multi-diffusion time experiment, which indicated that the mean square displacement grows linearly with the diffusion time (Einstein's relation) for the fast component, whereas the slow component's mean square displacement reaches a plateau and does not change when the diffusion time is increased.

Basically, by increasing the diffusion time, the weighting of the component exhibiting restricted diffusion increases, and most of the restricted diffusion in white matter originates from the axonal compartment. In addition, it was found that the magnitude of restricted diffusion reduces when demyelination and axonal loss occur [Assaf, Kafri, et al. 2002]. It has been speculated that the multiple lamellae of myelin membranes significantly reduce the exchange rates of water molecules between intra-axonal and extra-axonal spaces, causing the intra-axonal molecules to be restricted. Some studies have calculated that the intra- to extra-axonal exchange time may reach 500 ms [Assaf and Y. Cohen 2000; Meier, Dreher, and Leibfritz 2003].

Within brain tissue, water diffusion is regulated by the different cellular compartments. Different models were developed to decode the microstructural properties of these cellular compartments and will be described later in the section dedicated to compartmental analytical models (see section 3.3), and their accuracy on WM microstructure estimation will be evaluated thanks to the use of physical phantoms, which is the topic of the following section.

3.2 Design of hardware microstructural phantoms

To challenge and evaluate the accuracy of the microstructural information obtained from q-space diffusion MRI under different experimental conditions, it is important to devise and study hardware phantoms that share similar microstructural characteristics. Thus, microstructural hardware phantoms of different sizes and mixtures have been studied [Drobnjak, Neher, et al. 2021] [figure 3.3]. Originally, glass and plastic capillaries were proposed to design artificial crossing fibers [C.-P. Lin et al. 2003; Tournier et al. 2008] with diameters ranging from 50 to 350 μm, thus remaining out of the range from real axons. Then, some hardware phantoms proposed hydrophobic and resistant synthetic fibers of diameters ranging from 5 to 50 μm [Hagen and Henkelman 2002; Fieremans, De Deene, et al. 2008; Poupon et al. 2008; Moussavi-Biugui et al. 2011; Burcaw, Fieremans, and Novikov 2015].

Finally, innovative phantoms proposed to create more realistic hollow fibers using electro-spinning or melt-spinning techniques [Zhang 2012; Hubbard et al. 2015; Guise et al. 2016].
The phantoms define geometries of known size for which there is an empirical solution to the signal decay dependency on the q-value. [Avram, Assaf, and Y. Cohen 2004] conducted a comprehensive validation study of q-space diffusion MR of cylindrical micro-capillaries of different sizes under different experimental conditions using different pulsed gradient stimulated echo sequences. The test leading to signals shown in [figure 3.4] is a standard for validation and is systematically used to confirm the relevance of a model.

3.3 Multi compartmental analytical models

Since the observation of the bi-exponential nature of the diffusion MRI signal in WM tissue, multi-compartmental models have been proposed by the state of the art. Most of them rely on the modeling of compartments assumed to be impermeable, which remains acceptable for myelinated axons, enabling to write the acquired diffusion-weighted MRI signal as a sum of signal contributions stemming from the various compartments. Below, we present an exhaustive list of these models:

Restricted diffusion in Bovine Optic Nerve - Stanisz first introduced a model to explore the non-gaussian behavior of the dMRI signal in the bovine optical nerve [Stanisz et al. 1997]. It relies on an analytical multi-compartment model of the bovine optic nerve associated with 3 compartments and yielding [equation 3.1]:

$$S(b) = M_E(b) + M_S(b) + M_T(b), \tag{3.1}$$

with S(b) the total PGSE signal value and $M_E(b)/M_S(b)/M_T(b)/$ respectively the PGSE signals associated to the extracellular space, the glial cells and the axons. Geometrically, axons were represented as ellipsoid and glial cells as spheres [figure 3.5]. Parameters in this model were then adjusted to fit data collected from PGSE experiments, including intra- and extracellular diffusivities, axons and glial cell average lengths, volume fractions, and permeabilities. The extended set of parameters to be estimated requires high-quality diffusion MRI data, leading to very long acquisition times. Such an approach cannot be translated to clinical applications; simpler models with fewer parameters have thus been developed.

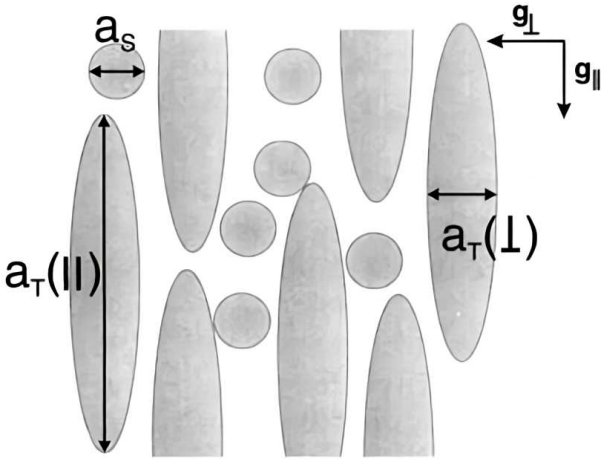
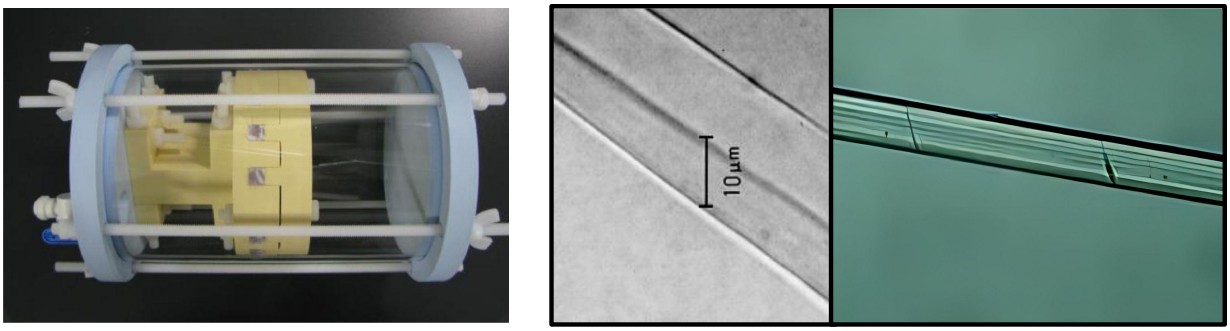
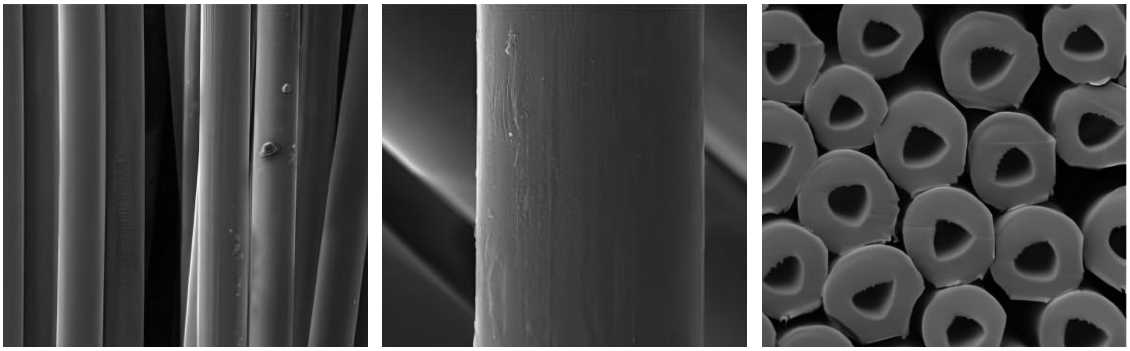


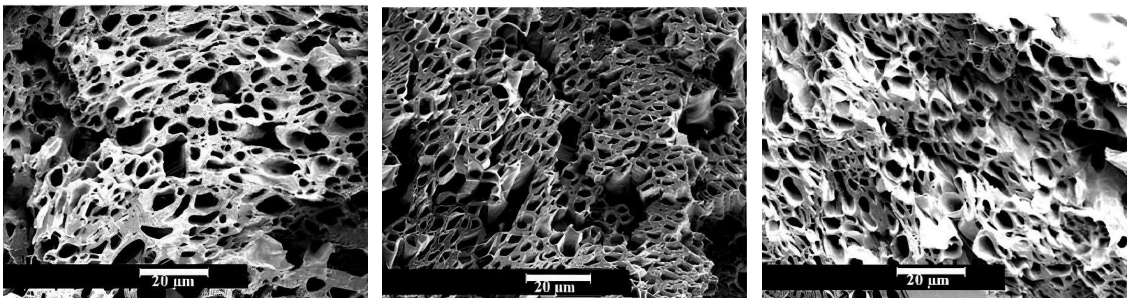
Figure 3.5: *Postulated tissue model as an approximation of bovine optic nerve structure. The axon cells are represented by prolate ellipsoids with short dimension $a_T(\perp)$ and long dimension $a_T(\parallel)$, glial cells are represented by spheres with diameter a_S, adapted from [Stanisz et al. 1997].*



a) Fiber Cup dMRI phantom *b) Synthetic fibers*



c) Electro-spinning hollow fibers



d) Melt-spinning hollow fibers

Figure 3.3: *Hardware phantoms of the microstructure. a) Phantom filled with distilled water inside its plexiglass container, adapted from [Fillard et al. 2011]. b) Example of synthetic fibers: (left) acrylic fiber of diameter 20 µm; (right) Dyneema® fiber of diameter 10 µm. Adapted from [IRB n.d.]. c) Melt-spinning: Hollow Polypropylene Yarns of inner diameter 12 µm and outer diameter 34 µm obtained by melt-spinning extrusion. Reproduced from [Guise et al. 2016]. d) Electro-spinning: Example of electrospun hollow fibers with different inner diameters according to the process flow. Reproduced from [Hubbard et al. 2015].*

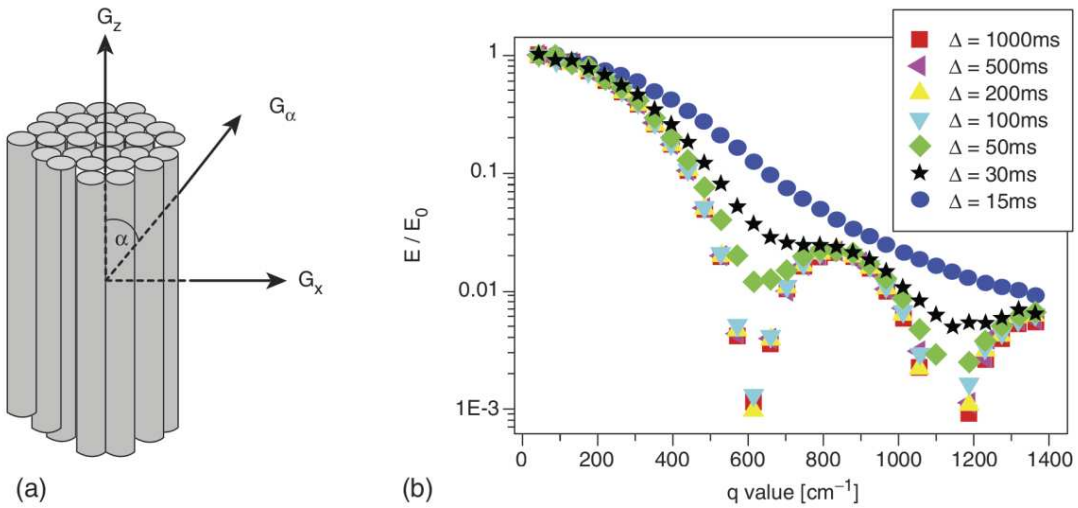


Figure 3.4: *a) Illustration of a phantom comprising microcapillaries. Gx and Gz are the gradient directions perpendicular and parallel to the main axis of the microtubes, respectively, and α is the rotational angle between the applied diffusion gradient, Ga, and the main axis of the cylinder, thus, the z-axis. b) Normalized signal decay (E/E0) for the 20 µm cylinders at a rotational angle α of 90°, as a function of q-values, for different diffusion times (Δ). Reproduced from [Avram, Assaf, and Y. Cohen 2004].*

Ball and Stick - [T. E. Behrens et al. 2003] proposed two compartments: one extra-axonal and one intra-axonal space, respectively, described as balls and sticks. It relies on the following equation:

$$S/S_0 = fe^{-bD(\mathrm{n.G})} + (1-f)e^{-bD}, \tag{3.2}$$

with f being the volume fraction of the anisotropic compartment (the sticks), n the mean orientation of axons and G the diffusion gradient direction. In this model, diffusion is assumed to occur only parallel to the axons inside the sticks, while it diffuses isotropically in the extra-axonal space (the balls). The simplification brought by this approach allows a clinical application, but it shows some limits as well in the capacity to reflect the real ground truth notably related to the assumption of isotropic diffusion in the extra-axonal space.

Composite Hindered and Restricted model of diffusion (CHARMED) - CHARMED is a two-compartment model of white matter describing a more realistic representation of the intra- axonal space [Assaf and Basser 2005]. Indeed, axons are represented as a distribution of packed parallel and impermeable cylinders forming one or two populations of fibers. The extra-axonal space is modeled by hindered Gaussian diffusion. CHARMED estimates the axonal fiber orientations, the parallel intra-axonal diffusivity, the extracellular diffusivity, and the respective volume fractions.

AxCaliber - AxCaliber [Assaf, Blumenfeld-Katzir, et al. 2008] is an extension of the CHARMED model, which assumes a known fiber direction and aims to estimate the distribution of axon diameters. Based on the histological work of Aboitiz [Aboitiz, R.S. Fisher, and E.Zaidel 1992], it assumes that axon diameters follow a Gamma distribution. The AxCaliber framework is well suited to the estimation of axon diameter in sciatic and optic nerves, but is not appropriate for clinical applications since it requires many measurements perpendicular to the studied bundle and assumes prior knowledge of fiber orientation.

ActiveAx - The purpose of the ActiveAx [Alexander 2008; Alexander 2010] model was to tackle the limitations of the AxCaliber framework to make the estimation of axon diameter feasible in a clinical routine. It uses rotative invariant dMRI acquisitions to infer the fiber orientation. The Gamma distribution model of axon diameter in AxCaliber is replaced by an "axon diameter index" corresponding to a single diameter value deduced from the mean volume-weighted diameter of the distribution. The extra-axonal space model follows the CHARMED intra-axonal space model but a third compartment featuring isotropic free diffusion is added to the signal model, describing partial volume effects stemming from CSF within tissues. The ActiveAx model relies on the following equation:

$$S = f_{intra}S_{intra} + f_{extra}S_{extra} + f_{iso}S_{iso}, \tag{3.3}$$

with f the volume fraction of each compartment such as $f_{intra}+f_{extra}+f_{iso} = 1$. Intra- and extracellular diffusivities are fixed to a constant. Despite the reduction of the number of parameters compares to AxCaliber, the ActiveAx framework still requires high-quality data and strong gradient strengths for the parameter estimation. In particular, estimating the axon diameter on a clinical setup remains challenging and controversial [Markus Nilsson et al. 2017].

One of the limitations of previous models was to consider axon populations as parallel structures, which is a strong approximation. Indeed, even in the corpus callosum, considered to be a region with strong axonal parallelism, some orientation dispersion can be observed [Ronen et al. 2014; Mollink et al. 2017].

Neurite Orientation Dispersion and Density Imaging (NODDI) - The next improvement regarding analytical models came from the NODDI approach [Zhang 2011; Zhang 2012]. It assumes that a parametric Watson distribution can represent fiber dispersion. The model is composed of three water pools: intra-cellular, extracellular, and CSF:

$$S = (1 - v_{iso})(v_{ic}S_{ic} + (1 - v_{ic})S_{ec}) + v_{iso}S_{iso}, \tag{3.4}$$

with S the full normalized signal, S_{ic} and v_{ic} the normalized signal and volume fraction of the intra-cellular compartment, S_{iso} and v_{iso} the normalized signal and volume fraction of CSF and S_{ec} the normalized signal of the extra-cellular compartment.

The intra-neuritic signal contribution S_{ic} adopts the following orientation dispersed cylinder model simplified for sticks:

$$S_{ic} = \int_{S^2} f(\mathrm{n})e^{bd_{||}(\mathrm{qn})^2}d\mathrm{n}, \tag{3.5}$$

with q the gradient direction, f(n)dn the probability of finding sticks along orientation n, $e^{bd_{||}(\mathrm{qn})^2}$ the signal attenuation due to unhindered diffusion along sticks with intrinsic diffusivity $d_{||}$ and orientation n. f(n) is modeling the orientation distribution function with a Watson distribution such as:

$$f(\mathrm{n}) = M\left(\frac{1}{2}, \frac{3}{2}, \kappa\right)^{-1} e^{\kappa(\boldsymbol{\mu}.\mathrm{n})^2}, \tag{3.6}$$

with M the confluent hypergeometric function, $\boldsymbol{\mu}$ the mean orientation and κ a concentration parameter that measures the extent of orientation dispersion $\boldsymbol{\mu}$. A Watson distribution is the simplest orientation distribution that can capture the dispersion in orientations [Mardia and Jupp 2009].

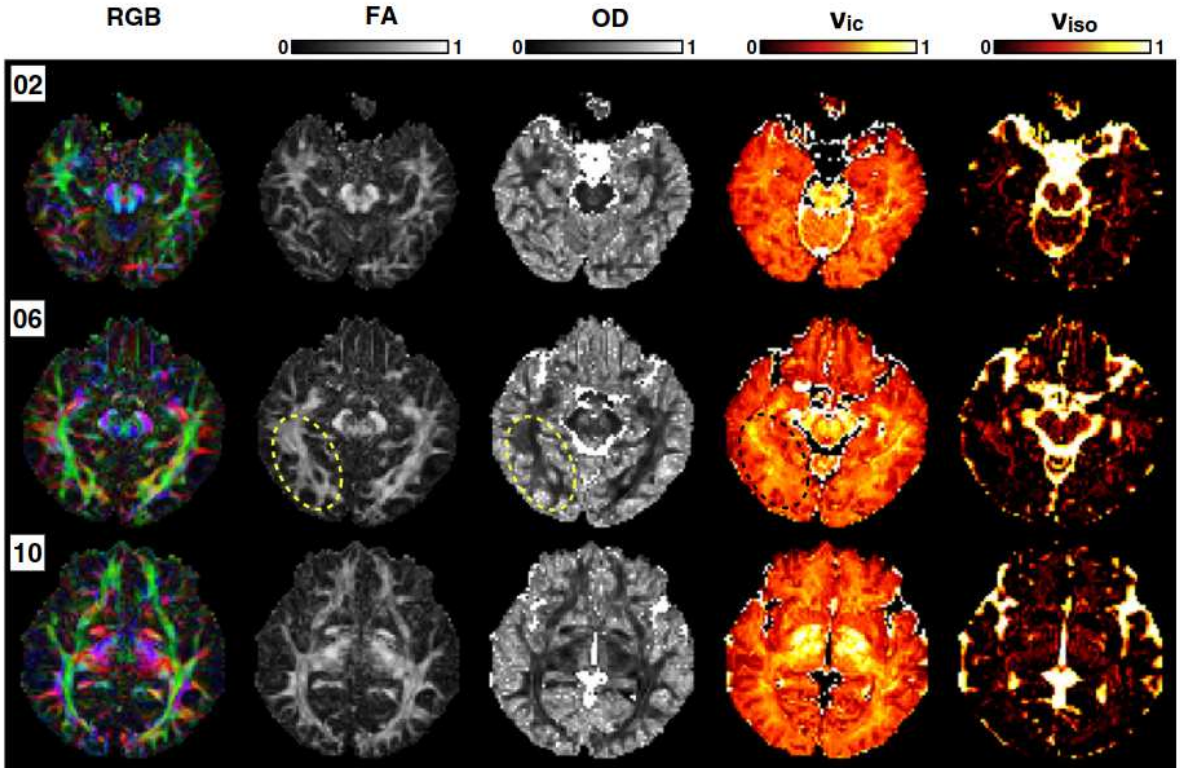


Figure 3.6: *Maps of RGB-encoded principal direction $\boldsymbol{\mu}$, FA, orientation dispersion index OD, intra-cellular volume fraction vic, and isotropic (CSF) volume fraction viso computed using the four-shell data, showing every 4th slice of the inferior half of the brain. Adapted from [Zhang 2012].*

Same formalism can be applied for S_{ec}, such as :

$$log(S_{ec}) = -b\mathrm{q}^T \left(\int_{S^2} f(\mathrm{n}) D(\mathrm{n}) d\mathrm{n} \right) \mathrm{q}, \tag{3.7}$$

NODDI adopts a classic isotropic Gaussian diffusion distribution for the CSF compartment.

The 2012 version of NODDI has the benefit of estimating all model parameters with only 2 HARDI shells (two b-values), thus feasible in a clinical routine [figure 3.6], making the NODDI model popular, as shown by the studies using it to characterize healthy and pathological brain states [Kodiweera et al. 2016; Colgan et al. 2016].

Extensions to NODDI were made notably by proposing different orientation dispersion models distribution such as the Bingham distribution [Tariq et al. 2016] or including spherical decomposition [Zucchelli, Descoteaux, and Menegaz 2017]. The NODDI-DTI model [Edwards et al. 2017] is ignoring the CSF compartment, thus allowing extraction of biophysical parameters from DTI data. NODDIDA model [Jelescu et al. 2016] ignores CFS contribution, and all diffusivities are independent of each other. This approach removes some incorrect assumptions from NODDI, but the problem remains ill-posed: multiple parameter sets can describe the dMRI signal equally well, and hence, the solution is not unique. Also, it reduces NODDI from a three-compartment model to a two-compartment model, by eliminating the isotropic compartment, the NODDIDA model is only applicable in brain regions with minimal CSF occupancy [Mozumder et al. 2019]. The LEMONADE(t) [Novikov, Veraart, et al. 2018] model aims at overcoming some of the issues of NODDIDA by accounting for the degeneracy in model parameter estimation, LEMONADE enables one to choose between two parameter branches, selecting the one corresponding to biophysical ground truth.

Soma and neurite density imaging (SANDI) - While previous models successfully describe diffusion data in white and grey matter, it has been shown to fail in the grey matter at high b values ($>$ 3.000 s/mm^2). SANDI [Palombo 2020] hypothesized that the unmodelled soma compartment also embedding neurogli and neurons may be responsible for this failure. Thus, SANDI proposes a three-compartment model considering a signal contribution from cell bodies assumed to arise from a pool of diffusion water molecules restricted in a sphere:

$$S = f_{ic}(f_{in}S_{in} + f_{is}S_{is}) + f_{ec}S_{ec}, \tag{3.8}$$

with S the resulting T2 normalized signal, f_{ic} and f_{ec} the intra and extracellular relative fractions satisfying $f_{ic} + f_{ec} = 1$, S_{in} and S_{is} the normalized signals for restricted diffusion within neurites and soma and S_{ec} the normalized signal stemming from the extracellular space. Notably, the model uses GPD approximation [Neuman 1974] for normalized signal representations. SANDI also involves Monte-Carlo simulation of spin-diffusion in realistic digital models of dendritic structures to investigate the regime of validity of the assumption of non-exchanging intra-cellular compartments and investigate the sensitivity to soma size and density within that regime. Monte-Carlo simulations will be further discussed in the next section. Compared to NODDI, SANDI is not an easy model to apply to clinical routines as high b values are required, relying on gradient hardware to deliver substantial gradient magnitude.

Cellular Exchange Imaging (CEXI) - CEXI is a model tailored for permeable spherical cells [Gardier et al. 2023]. As SANDI, CEXI model results were compared with Monte-Carlo simulations and show stable estimates of cell size and intracellular volume fraction. CEXI accurately estimates the exchange time for low to moderate permeability levels. This study highlights the importance of accurately modeling the exchange time to quantify microstructure properties in permeable cellular substrates.

This non-exhaustive list of analytical multi-compartmental models of WM dMRI signal points to a general trend that can be highlighted by this quote extracted from [Zhang 2012]:

> Our approach is first to choose a model that is sufficiently simple yet complex enough to capture the key features of neurite morphology, then identify the optimal acquisition protocol for such a model under scanner hardware and acquisition time constraints typical in a clinic setting."

Indeed, as we saw in the previous chronological list, the analytical model approach constantly struggles to balance model accuracy and clinical feasibility. However, since the first analytical model from Stanisz [Stanisz et al. 1997], a computational approach was already considered, based on satisfying results of Monte-Carlo simulations [Szafer, Zhong, and Gore 1995] but not developed further for technical limitations on computational times. The following paragraph introduces biophysical models based on a computational approach.

3.4 Computational models and dMRI simulator

Analytical models have multiple limitations, starting from the oversimplistic geometry representation that doesn't reflect authentic biological architecture. Moreover, the mathematical formalism doesn't always bring unique solutions, as several local minima can exist in the parameter space, which may induce degeneracy [Jelescu et al. 2016]. These issues are linked to the lack of solutions to the diffusion equations for complex geometries, which is, unfortunately, an intrinsic problem of dMRI.

The rise of High-Performance Computing (HPC) over the last decades supports the idea of computational models as an alternative to analytical models. The idea is to train a machine learning algorithm to decode microstructural parameters from brain tissues based on their dMRI signal footprint. To train the algorithm, it must be provided with a database including the information from the microstructure geometry and its dMRI footprint. Creating a database stemming from dMRI acquisitions of histological samples of brain matter is unrealistic, given the number of samples required and, consequently, the potential costs in terms of time and resources. An alternative is to implement a computational simulator to synthesize digital phantoms representative of the biological tissue microstructure, employ these phantoms to simulate the diffusion of water process within it, and then use a discrete model of the dMRI signal attenuation resulting from the integration of all water molecule trajectories.

The geometrical parameters describing the microstructure and its dMRI signature are the two sides of the targetted database we want to synthesize and from which the machine learning algorithm is assembling a regression function. These simulations are limitless concerning cellular microstructure realism, diffusion process hypothesis, or simulated MRI hardware if the framework is well designed and the computational resources to perform the simulations are sufficient. The simulator must address the following key issues:

- A high level of realism for each simulated sample must be reached: the geometry must be simulated accurately with respect to the ground-truth anatomy at cellular scales, the simulated diffusion process must match real phenomenon, and the diffusion sequences generated must correspond to what can be acquired with real MRI hardware.
- A computational efficiency must be found to generate large collections of samples, which is essential to robustly train machine learning algorithms.

With these two challenges in mind, a review of the literature is proposed hereafter about the three key components required to meet them: simulation of brain tissue geometry, simulation of the diffusion process, and simulation of the dMRI signal attenuation. This review will be followed by a description of the first computational models established by the community for decoding the brain tissue microstructure.

3.4.1 Simulation of cell population geometries

The membrane geometry of cell populations present in gray and white matter tissue can be very different from one species to another. Their cell shapes are drastically different, the grey matter being mostly populated by neurons and the white matter being mostly populated by their axonal prolongations assembled into large bundles. Because of the key application of dMRI to infer brain structural connectivity, first geometry simulations have addressed the representation of white matter

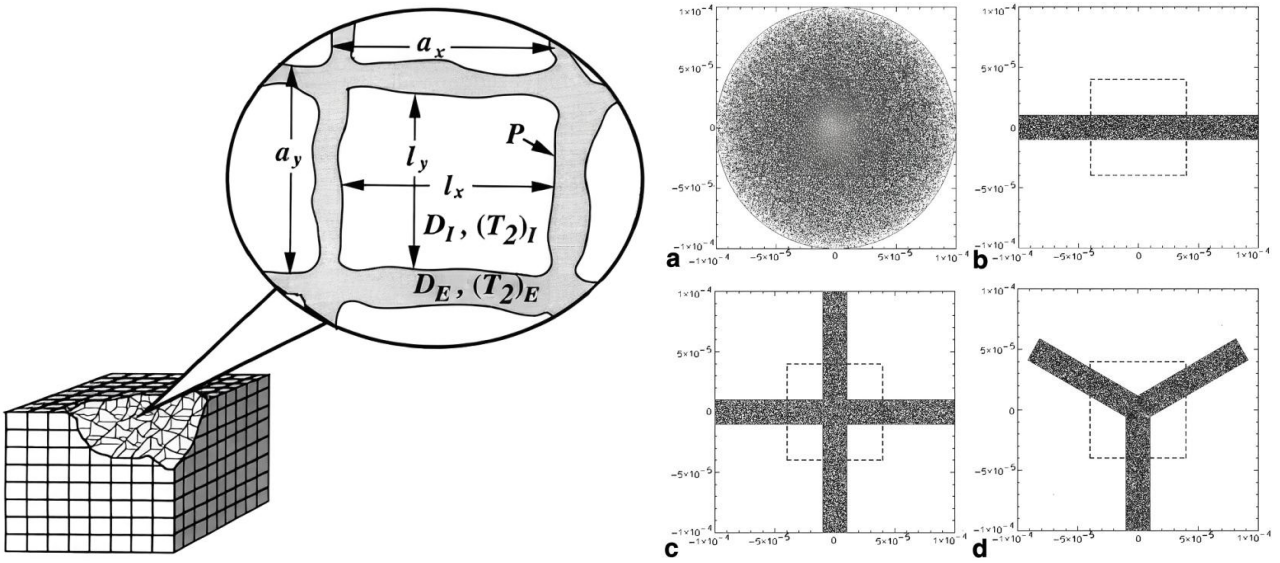


Figure 3.7: *Pioneering cellular membranes simulators. On the left side, adapted from [Szafer, Zhong, and Gore 1995], a cell represented by a box surrounded by a membrane of permeability. On the right side, adapted from [Liu et al. 2004], we can observe a cross-sectional view in the XY plane of 3D synthetic phantoms. The black dots are uniformly distributed in the initial positions of the spins. a: Phantom 1, isotropic sphere. b: Phantom 2, single tube. c: Phantom 3, crossing tube. d: Phantom 4, Y-shaped tube. Tubes have a square cross-section with a width of 20 μ, and the boundaries of the tubes were assumed to be impermeable. Spins can diffuse freely inside the phantom and are reflected elastically at the boundary indicated by the solid lines. The box in the middle, drawn with a dotted line, is the voxel of interest.*

samples [Matt G. Hall and Daniel C. Alexander 2009; Balls and L. R. Frank 2009; Fieremans, Novikov, et al. 2010; Harkins and Does 2016; Villarreal-Haro et al. 2023; K. Ginsburger et al. 2018]. Later were proposed geometry simulation of grey matter [Yeh et al. 2013; Ginsburger et al. 2019; Kerkelä et al. 2020; J.-R. Li, V.-D. Nguyen, et al. 2019; Fang et al. 2023]. The following section presents these different methods to simulate brain tissue microstructure.

3.4.1.1 Synthetic geometries

Various geometry representations can be considered depending on the objectives with respect to bio-realism, computational efficiency, and IT maturity.

Grid representation - [Szafer, Zhong, and Gore 1995] first proposed a cellular general tissue model represented by a periodic 3D grid where the individual square element represents a cell surrounded by a membrane with specific permeability [figure 3.7]. The edges of the structure are assimilated to the extracellular space, and both intra and extracellular spaces share the same bulk diffusivity. This attempt to represent the space as a 3D cartesian grid had the benefit of being computationally efficient but lacked realism regarding the ground-truth microstructure complexity. More adaptative representations were then developed, allowing more complex representations.

Generative model - The cells composing any simulated sample can be associated with a geometric pattern described by a reduced set of parameters describing the key geometrical features of its microstructure. As first approximations, white matter fibers were represented by cylinders [Balls and L. R. Frank 2009; Matt G. Hall and Daniel C. Alexander 2009] and neuron somas by spheres [Yeh et al. 2013; Ginsburger et al. 2019]. Despite being oversimplistic, these representations are very useful for

assessing the accuracy of diffusion simulations since the analytical solutions of the diffusion equation exist only for cylinders, spheres, ellipsoids, and plans. Accordingly, [Liu et al. 2004] synthesized four different 3D phantoms presented in [figure 3.7] and [Kuchel, Lennon, and Durrant 1996; Duh, Mohorič, and Stepišnik 2001] respectively proposing to simulate phantoms of a sphere and two plates.

To enhance the realism of cell representation, another approach consists of using 3D surface meshes to represent the geometry of each cell [Balls and L. R. Frank 2009; Matt G. Hall and Daniel C. Alexander 2009; Yeh et al. 2013; D. V. Nguyen et al. 2014; M. Lin et al. 2016]. Triangular or quadrilateral 3D meshes are the most common representations of 3D surfaces. The vertex density greatly impacts the accuracy of the geometrical representation and the computational efficiency. The following quote from [Balls and L. R. Frank 2009] confirms this statement:

> "Generating meshes representing true fiber dimensions from histology is a painstaking process, so perfect realism in our simulations may be too much to expect."

As an alternative to meshes, MEDUSA simulator [K. Ginsburger et al. 2018] proposes to decompose cells into simple geometric elements called "atoms", which allow the compression of the representation of cells. The next chapter will further develop the MEDUSA simulator approach.

Experimentations have shown that random packing of axons in their transverse plane plays a crucial role in the time-dependence of the diffusion coefficient in white matter [Burcaw, Fieremans, and Novikov 2015]; thus, generative simulators that represent white matter as an arrangement of regularly packed parallel cylinders miss the actual complexity of the diffusion process in white matter.

To improve the realism of the representations of WM microstructure, alternative simulators [Balls and L. R. Frank 2009; Matt G. Hall and Daniel C. Alexander 2009; Yeh et al. 2013; Ginsburger et al. 2019; Villarreal-Haro et al. 2023] propose to define fibers with more complex shapes and configurations (straight, bent, crossing, kissing, twisting, and fanning configurations) with an adjustable volume fraction. The realism can be enhanced by varying the radius of axonal cylinders within the scene using an adequate model of the axon diameter distribution (commonly a Gamma distribution). Dispersing the orientations of fibers is also crucial for realism; the CAMINO simulator [Matt G. Hall and Daniel C. Alexander 2009] was the first to propose a WM geometry simulator able to create crossing fiber bundles. The DMS simulator [Yeh et al. 2013] was going further by offering more than two different populations of fibers and a global angular dispersion parameter, allowing fibers from the same population to drift slightly from the mean population orientation, thus creating a more organic organization of the bundle. Furthermore, the DMS simulator proposes additional geometries, like ellipsoids or spheres, that are ideal to mimic the geometry of glial cells or somas. [figure 3.8] presents 2D and 3D samples of simulated WM microstructure with mixed sophistication.

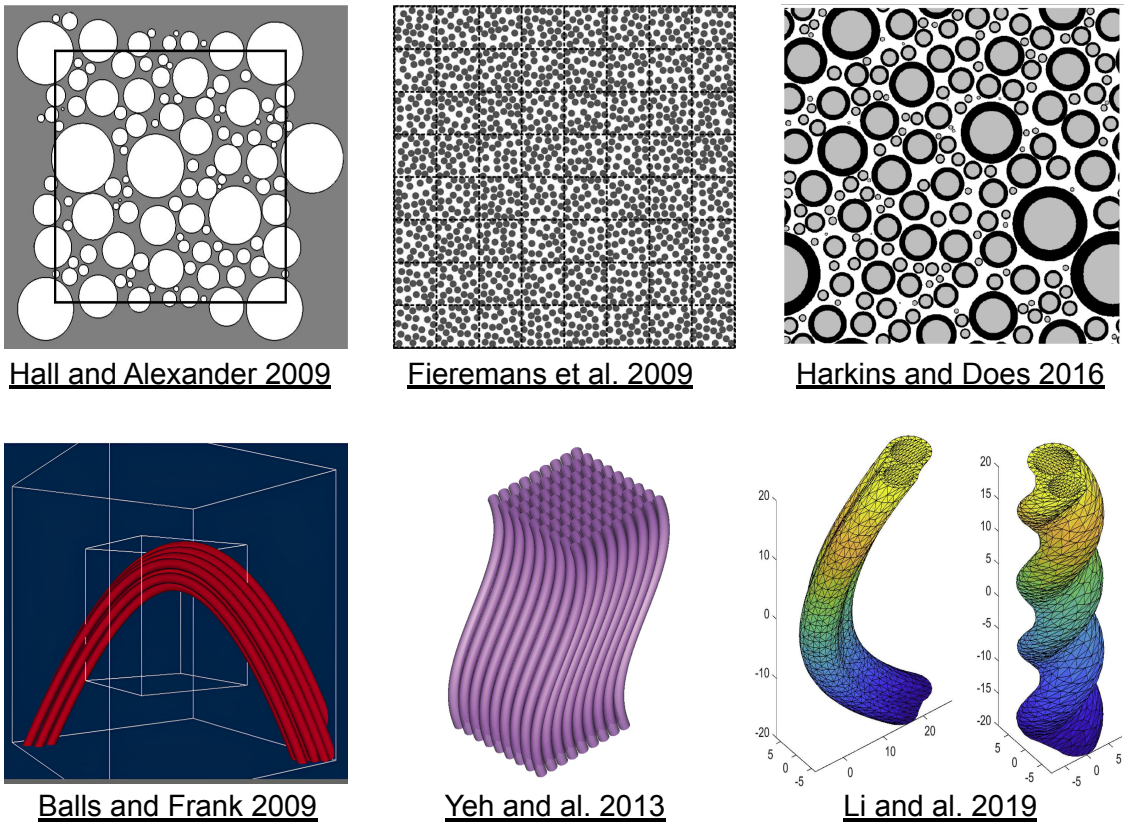


Figure 3.8: *Examples of packing methods developed to create white matter numerical phantoms. On the top are illustrated cross sections of three different simulated geometries, and on the bottom are illustrated three 3D phantoms of simulated geometries relying on meshes. Figures adapted from [Matt G. Hall and Daniel C. Alexander 2009; Fieremans, Novikov, et al. 2010; Harkins and Does 2016; Balls and L. R. Frank 2009; Yeh et al. 2013; J.-R. Li, V.-D. Nguyen, et al. 2019].*

The improvements in brain tissue microstructure representation techniques have made it possible to depict more than healthy tissues; consequently, different studies [Matt G. Hall and Daniel C. Alexander 2009; Budde and J. A. Frank 2010; Landman et al. 2010; Yeh et al. 2013; M. Lin et al. 2016; K. Ginsburger 2019] have proposed to simulate damaged neurite structures [figure 3.9]. Beading of axons, notably, is known to be pathologically associated with ischemic stroke and is an appropriate application of computational models [Lee, Jespersen, et al. 2020] since analytical models cannot easily be developed for axonal beading.

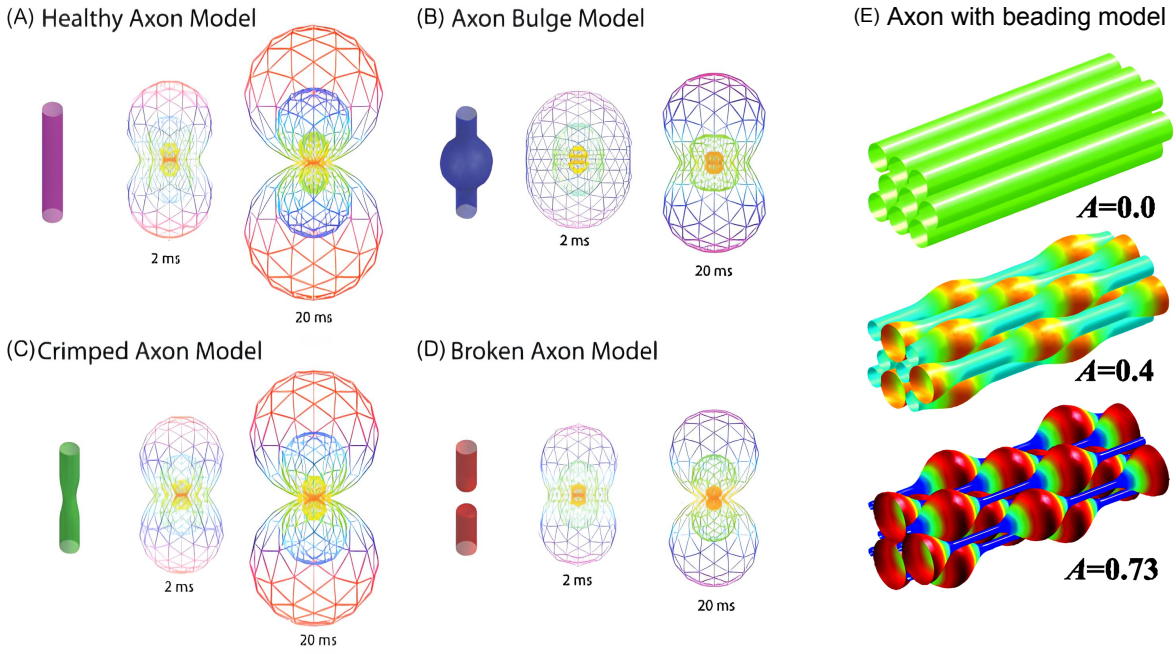
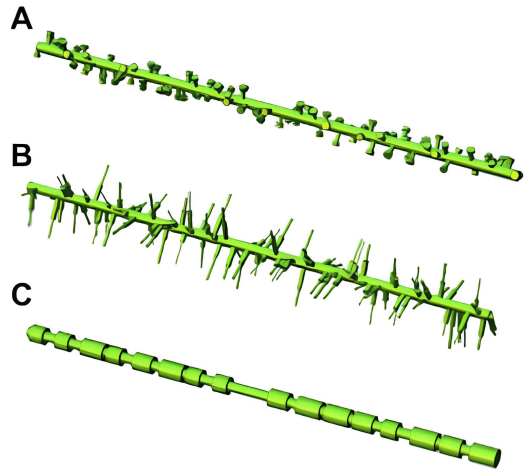


Figure 3.9: *Damaged neurites simulations : (A to D) plot the 10th, 50th, and 90th percentile isosurfaces of the motion-probability propagators for different axonal structures, adapted from [Landman et al. 2010]. (E) Simulator of packed beading neurites. Beading cylinders were packed in a hexagonal pattern, adapted from [Budde and J. A. Frank 2010].*

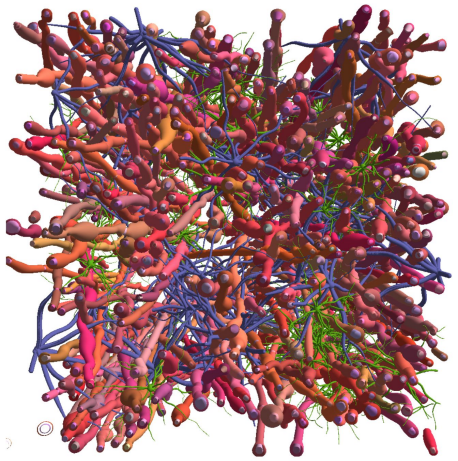
The development of ultra-realistic generated geometries has become pivotal these few last years, as shown by [M. Palombo et al. 2018] work focusing on the synthesis of substrates mimicking one cellular branch presenting microscopic details such as spines, leaflets, or beads. The MEDUSA simulator [Ginsburger et al. 2019] was one of the first to simulate with a high degree of realism WM fiber bundles, including extra microstructural details such as fibers tortuosity, beading, Ranvier nodes, or myelin sheath. Moreover, MEDUSA also simulates realistic astrocytes and oligodendrocytes. The Config simulator [Marco Palombo 2019; R. Callaghan et al. 2020] proposed to mimic the natural fiber growth relying on a set of rules motivated by biological axonal guidance mechanisms. Recently, the CACTUS simulator [Villarreal-Haro et al. 2023] implemented a "fiber radial growth" method, allowing the fibers to deform radially and thus modify their cylindric shape to become more bio-realistic. These ultra-realistic generative simulators are presented in [figure 3.10].

3.4.1.2 Acquired geometries with microscopic imaging modalities

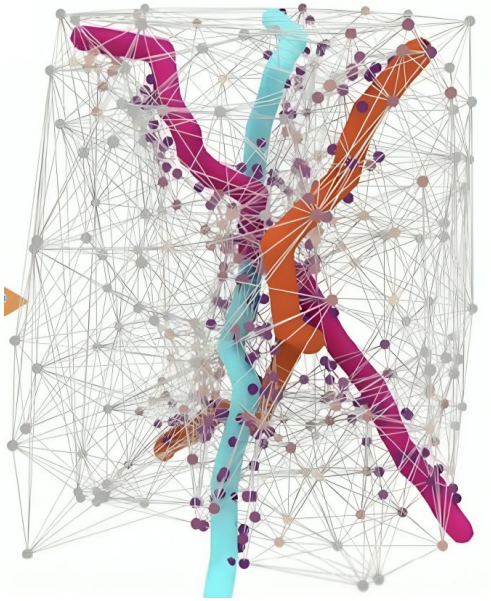
To fully catch the complexity of the brain microstructure, the generative approach must integrate multiple parameters, and methods within the simulator to mimic the anatomical ground truth, which can be complex and a never-ending challenge regarding the complexity of the brain tissue microstructure. Therefore, an alternative solution would be to extract tissue ground-truth membrane geometries



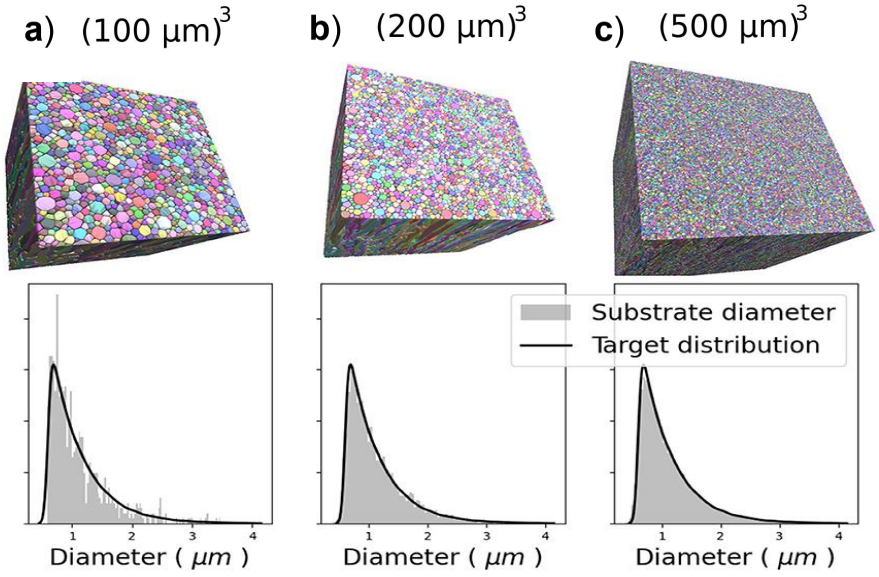
a) (Palombo et al. , 2018)
Synthetic substrates of cellular branches



b) (Ginsburger et al. , 2019)
MEDUSA: Microstructure Environment Designer with Unified Sphere Atoms simulator



c) (Callaghan et al. , 2020)
Config: Contextual Fibre Growth simulator



d) (Villarreal-Haro et al , 2023)
Cactus: high axonal volume fraction simulator

Figure 3.10: *Ultra realistic generative model of WM microstructure. a) Synthetic substrates mimicking three specific finer morphological characteristics of cellular branches: spines (A), leaflets (B), and beads (C). Adapted from [M. Palombo et al. 2018]. b) MEDUSA simulated samples of WM presenting axonal fibers, astrocytes, and oligodendrocytes. Adapted from [K. Ginsburger 2019]. C) Overview of the basic growth algorithm in ConFiG. Three fibers are shown with a growth network. Adapted from [R. Callaghan et al. 2020]. D) Cactus simulator: three 3D substrates of varying FOV sizes with 1,316, 4,859, and 33,478 fibers, respectively. The empirical and target radii distributions are displayed on the bottom of each substrate. Adapted from [Villarreal-Haro et al. 2023].*

directly by meshing their surfaces. Various biological or chemical materials sharing similar microstructure with the human brain were scanned using high-resolution imaging methods; examples of such materials are gel foam [Baete et al. 2008] or asparagus steam [Panagiotaki et al. 2010] scanned respectively with micro CT-scan and high-resolution confocal laser scanning microscopy and then reconstruct using 3D surface representations [figure 3.11].

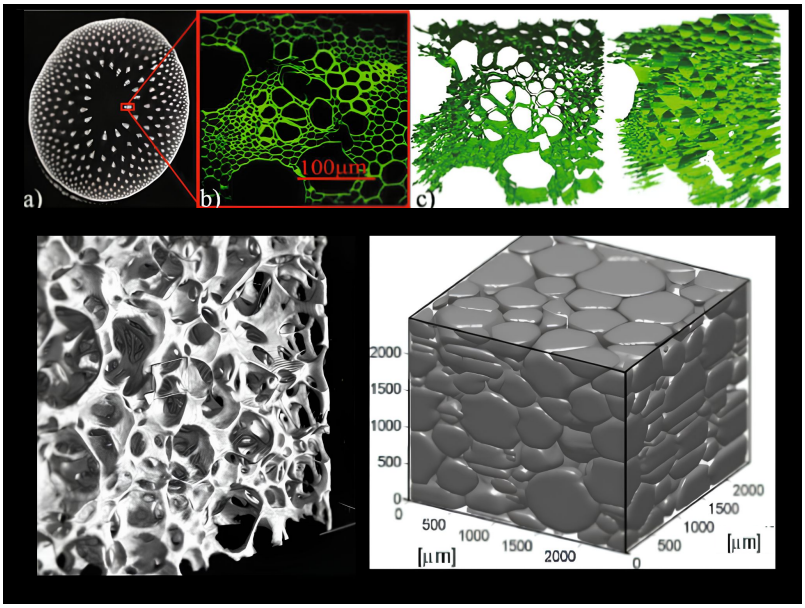
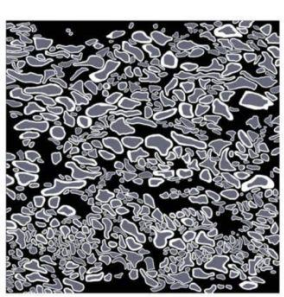


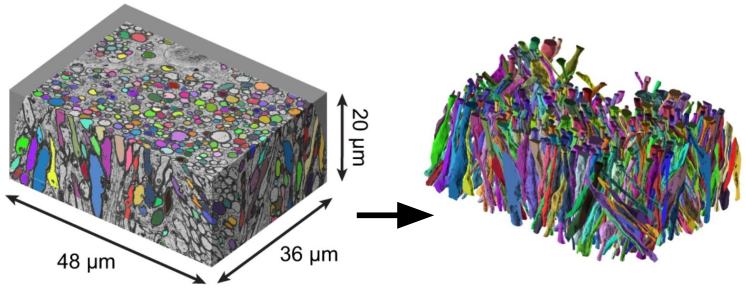
Figure 3.11: *Biological and chemical materials as white matter phantoms. On top of the illustration: a) DW-MRI image of a transverse section of the asparagus stem. The red square indicates the ROI, containing one of the vascular bundles. b) Scanned image of the same ROI. c) Three-dimensional mesh model shown from two different angles. Adapted from [Panagiotaki et al. 2010]. At the bottom, on the left, a 3D micro CT image of gel foam, and on the right, a three-dimensional view of one set of processed micro CT images of gel foam with each air bubble represented by a triangulated surface. Adapted from [Baete et al. 2008].*

Benefiting from the advances in microscopy, the approach was further pushed towards the scanning of brain tissue samples directly, enabling the reconstruction of ground-truth geometries stemming from 2D [Chin et al. 2002; T. Xu et al. 2018]and 3D scans [K.-V. Nguyen et al. 2018; Marco Palombo 2019; Lee, Jespersen, et al. 2020; Lee, Fieremans, and Novikov 2021] [figure 3.12]. To complete the full diffusion simulation, the segmented and meshed geometries need to be put as inputs of diffusion simulators like "SpinDoctor" [J.-R. Li, V.-D. Nguyen, et al. 2019] or "Disimpy" [Kerkelä et al. 2020]" that computes the dMRI footprint resulting from meshes. "Neuromorpho" [Ascoli, Donohue, and Halavi 2007] is an online website storing the different meshes of brain cells, whose database has been expended over the years and used to complete full diffusion simulations [Marco Palombo 2019; Fang et al. 2023].

It should be pointed out that the search for optimal realism of simulated microstructure may prove futile, as the simulated substrate is aiming to be decoded through the simulation of its dMRI footprint, and the diffusion sensitivity is capped such that the signal ignores small details from the simulated substrate.

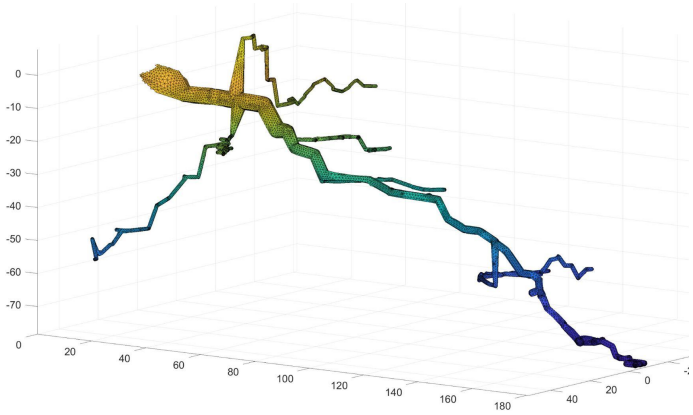


a) (Xu et al., 2018)
Mouse brain, 2D electron microscopy of segmented axons

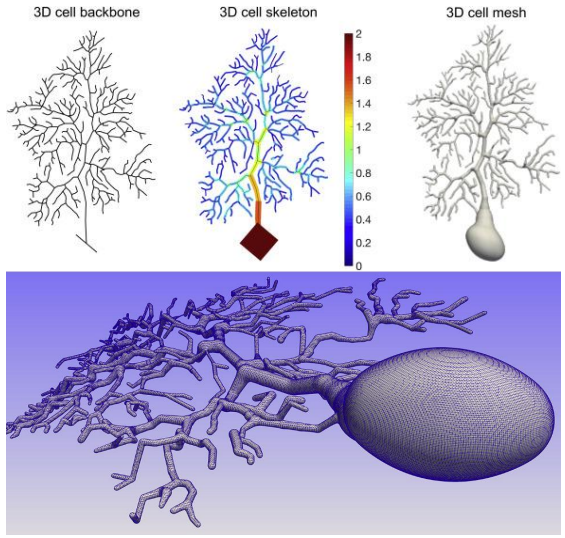


b) (Lee et al., 2020)
Mouse brain, 3D electron microscopy of segmented axons

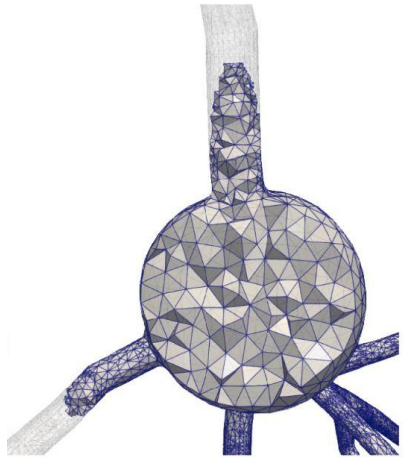
c) (Nguyen et al., 2018)
Mouse brain, confocal microscopy of astrocytes



d) (Jing-Rebecca Li et al., 2019)
Spin Doctor: extern neuron mesh geometry input



e) (Palombo et al., 2019)
Skeleton and mesh for a reconstructed ,cell from NEUROMORPHO



f) (Fang et al., 2023)
Tetrahedral mesh reconstructed from NEUROMORPHO

Figure 3.12: *a) Axons segmented from electron microscopy (EM) data of mouse WM. Adapted from [T. Xu et al. 2018]. b) Structural analysis of axons segmented from female mouse brain corpus callosum: left 3D EM image with segmented axons passing through the central slice, right 3D representation of the intra-axonal space (IAS) segmentation yielding 227 axons. Adapted from [Lee, Jespersen, et al. 2020]. c) View of two surface meshes, left: protoplasmic astrocytes, right: Fibrous astrocytes. Adapted from [K.-V. Nguyen et al. 2018]. d) SpinDoctor: Dendrite branch mesh. Adapted from [J.-R. Li, V.-D. Nguyen, et al. 2019]. e) Example of 3D cell backbone, skeleton, and mesh for a reconstructed Purkinje cell from the NEUROMORPHO database (neuromorpho.org). The color encodes cellular fibers and soma diameter (in µm).Adapted from [Marco Palombo 2019]. f). Cell's tetrahedral mesh imported from (neuromorpho.org). Adapted from [Fang et al. 2023].*

3.4.1.3 Filling the field of view (FOV)

The FOV is the dimension that encloses the scene comprising the simulated geometry. State-of-the-art simulators rely on cubic FOVs [Szafer, Zhong, and Gore 1995; Liu et al. 2004; Kuchel, Lennon, and Durrant 1996; Duh, Mohorič, and Stepišnik 2001] as the goal of the simulation is to be bio-realistic; nevertheless, most of the simulators propose a cross-section view of their geometry and therefore following a 2D formalism [Liu et al. 2004; Fieremans, Novikov, et al. 2010; Matt G. Hall and Daniel C. Alexander 2009] [figure 3.8].

Volume fraction and cell overlapping - Within a simulated substrate, the cell populations are distinct compartments with their own volume fractions within the FOV. This volume fraction is weighting the contribution of the compartment to the final dMRI signature of the sample. In the case of generative geometry, the synthesized cells might overlap during their creation process and thus reduce the realism of the simulated geometry. Various strategies emerged to fill digital phantoms with cells according to this potential overlapping. In MCell simulator [Balls and L. R. Frank 2009], each element (cylinder, sphere, ellipsoid) is added one after the other to reach the desired volume fraction. The fiber is discarded if an intersection is found, and another location and radius are randomly pushed up. In CAMINO [Matt G. Hall and Daniel C. Alexander 2009] and CACTUS [Villarreal-Haro et al. 2023], a deformation is applied to the cylinder representing the fiber when an intersection occurs, contributing to the elimination of any potential cell overlapping.

Overlapping makes high volume fractions difficult to reach and thus a challenge in the competition for realistic samples of the brain microstructure as the intracellular space can constitute up to 80% of the voxel volume [Bourne 2012; Voříšek et al. 2002]. Recent simulation frameworks, like Config [R. Callaghan et al. 2020] or CACTUS [Villarreal-Haro et al. 2023], reached respectively 80% 94.7% in the case of a single bundle simulation [figure 3.10]. CACTUS allows the generation of large fields of view and takes approximately 4 hours to create a substrate of $500\mu m$*$500\mu m$*$500\mu m$ on a desktop station equipped with 64 cores (2.4 GHz) and 400 MBytes of RAM.

3.4.2 Simulate the diffusion process using Monte-Carlo simulations

Most simulations of the diffusion process rely on Monte-Carlo (MC) simulations consisting of distributing a large number of particles (or random walkers) within a geometry and making them move according to iterative trajectories of length "r" and randomly oriented until reaching a final echo time (T_E). MC simulations are designed to mimic the Brownian motion of water observed empirically [R. Brown 1827] and formalized by an adjustment of Einstein's initial diffusion equation [Einstein 1905]:

$$r = \sqrt{6Dt_s}, \tag{3.9}$$

where t_s is the time step, D is the diffusion coefficient, and r is the step length. The factor 6 corresponds to the 6 degrees of freedom in a 3D space. In the case of a 2D diffusion process, one has to consider a factor 2. t_s and r are respectively the time and displacement step of the global diffusion phenomenon discretized into $N_{IterationCount}$ steps with $N_{IterationCount} = T_E/t_s$.

Whatever the MC implementations, first introduced by [Lipinski 1990], remain similar and consist of the following iterative process:

- At time t = 0, a set of N_p random walkers is created within the field of view of the virtual

sample at spatial positions randomly drawn from a 3D uniform distribution.

- After each time step, displacements are drawn randomly for all particles in order to replicate the real Brownian motion of water molecules.
  - If the particle stays in the same compartment after moving, its new position is kept.
  - If the particle collides with a cell membrane along its displacement, a new position has to be computed with respect to a particle-to-membrane membrane interaction model. Various interaction models exist that will be described later.
- The displacement of particles induces phase shifts related to the diffusion encoding schemes. Each particle's resulting phase shift contributes to the dMRI signal decay.
- The previous steps are repeated iteratively until reaching the echo time T_E of the MRI chosen sequence.

Particle-to-membrane interaction - Here is a list of different "particle-to-membrane" interaction strategies described in [Xing et al. 2013] and illustrated in [figure 3.13]:

- Elastic boundary reflection (EBR) - EBR interaction consists of moving the particle to a symmetric position with respect to the direction normal to the point of the collision on the membrane. The collision of the reflected step is checked, until no further membrane barrier cells are detected.
- Non-elastic boundary reflection (NBR) - Another approach is to keep the reflection effect from EBR but not consider it elastic. Thus, NBR interaction keeps the particle reflected but along another random direction.
- Equal-step-length random leap (ERL) - This approach considers that steps share the same length; thus, when a particle faces a membrane, the resulting trajectory is canceled and replaced with a new one, integrating a new orientation. This process is iteratively applied to the particle until it stops colliding with a membrane or until it reaches a maximum interaction count.

Computationally, ERL can easily exclude the computational cost linked with the reflection position. Indeed, considering that millions of collisions will be computed during the MC simulation, each reduction of computational cost for one particle actually impacts the simulation overall. When the step number is sufficiently large, and the step size is much smaller than the restriction size, the non-crossing random leap agrees with elastic collision, at least in the case of impermeable membranes [Xing et al. 2013].

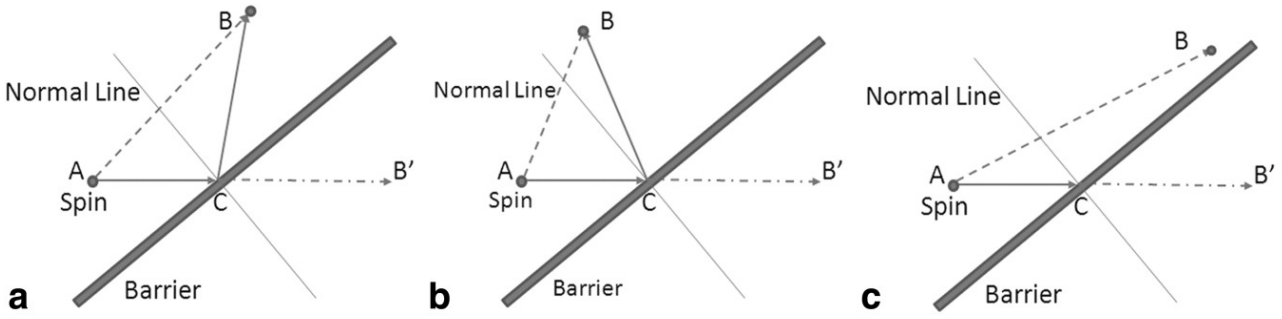


Figure 3.13: *Three particle-to-membrane interaction models. A is the starting point, and B is the ending point. B' is the supposed endpoint without barrier; (a) EBR. The length of the reflected step is the sum of AC and CB, which is equal to AB'; (b) NBR. CB returns to the original side along a random direction. The sum of AC and CB is also equal to AB'; (c) ERL. AB is sampled randomly to replace AB', and AB should not cross the barrier. Adapted from [Xing et al. 2013].*

Boundary conditions - The diffusion process is simulated within the FOV of the simulated geometry, thus establishing a bounding box for the distributed particles. A finite bounding box may influence the simulation result [Fieremans and Lee 2018], as all the particles crossing its edge see their contribution erased. This effect can be reduced at moderate diffusion times by calculating the impact of particles exclusively located in the central portion of the phantom as presented in [figure 3.7]. Still, it becomes essential to consider the impacts of the bounding box for long diffusion times, during which many particles are likely to cross these boundaries. A First solution, called "periodic boundary condition," used in CAMINO [Matt G. Hall and Daniel C. Alexander 2009], consists of reintroducing the particle at the opposite edge of the phantom. However, if not appropriately treated while building the phantom, geometrical discontinuities over the boundary could introduce an additional restricted diffusion effect corresponding to the box size. Another solution, the "mirroring boundary condition", simulates the diffusion outside the phantom by mirroring the particle position concerning the phantom edge it crosses. This mirroring method can artificially increase the fiber orientation dispersion of the phantom [Fieremans and Lee 2018] but can be used on a broader range of phantoms since it does not require periodicity.

Constant steps - Whereas first simulation frameworks relied on a Gaussian distribution [Lipinski 1990; Liu et al. 2004] of step lengths, the CAMINO simulator [Matt G. Hall and Daniel C. Alexander 2009] first proposed to switch to a constant step length. This choice corresponds to a numerical optimization mathematically justified by the statistical properties of random walks. Indeed, although a Gaussian distribution of velocities occurs in a real physical system, generating steps with Gaussian-distributed lengths is unnecessary to approximate them as the central limit theorem ensures that values obtained summing repeatedly over random values drawn from any distribution will converge to a Gaussian distribution. Using variable step size methods also exists [Grebenkov 2011], but these methods are harder to implement; in particular, they lead to a more complex computation of particle phase shifts, which prevents the code from being optimally parallelized.

Convergence and parameter choice for Monte-Carlo Simulation - Although the overall implementation of Monte Carlo simulations remained the same over the years, the realism of the simulated diffusion process increased jointly with the computational capacities. The quality of an MC diffusion simulation is correlated to its ability to catch the infinitesimal behavior of the water Brownian motion, which requires establishing a trade-off between computational resources and the degree of realism. The two parameters driving this balance are the particle count (N_p) and the step length (r).

If we visualize the phantom as a playground that needs to be investigated, the more particles there are, the greater the chance to cross all the space and capture the real diffusion signal decay. The same consideration stands for step length, which is linked to the time step [equation (3.9)]; the shorter it is, the closer the particles will envelop microstructural details from substrate elements. The remaining question is: "What's an optimal value for "N_p" and "r"? The answer will depend on the dimension of the microstructural elements and the targeted field of view. Based on studies from [Matt G. Hall and Daniel C. Alexander 2009; Xing et al. 2013] we can introduce two mathematical concepts helping to answer the question:

- Complexity (U) - The complexity U of a MC simulation is described as $U = N_p N_{IterationCount}$. With $N_{IterationCount}$, the number of updates during the simulation. In [Matt G. Hall and Daniel C. Alexander 2009], the duration of the simulation was 0.1s with a

time step dt = $\frac{0.1}{TE}$. In a simulation applied on simple geometries (packed cylinder), different values of U were considered to draw the graph of mean square error (MSE) of the signal obtained with simulation compared with its analytical ground truth [figure 3.14]. The minimum MSE observed on the graph corresponds to the optimal setting, although no dependence on the field of view or cell sizes was considered in this study.

- Relative Step Length (RSL) - It measures the ratio between the step length and the dimension of the structuring element. In Xing's paper [Xing et al. 2013], the geometry was composed of cylinders, with RSL representing 1%, 3%, 8%, 16%, 22%, or 28% of the radius. For different values of N_p, the signal acquired was compared with the corresponding analytical solutions (known for cylinder geometries), and a standard deviation and bias graph was established [figure 3.15]. The minimal value indicates the optimal value for the couple RSL/N_p. Xing's work focuses on the simple geometry of a cylinder; thus, there is no mention of the field of view size, which certainly also plays a role in the case of complex geometries.

The choice for N_p and step length r depends on the type of geometry. In their cookbook, [Fieremans and Lee 2018] wrote this recommendation:

> "A diffusion simulation without including enough particles effectively lowers the SNR (signal-to-noise ratio)[...] A clear cut-off for a minimally required number of particles has not been proposed so far, but a rule of thumb often used is that $10^5 - 10^6$ particles diffusing from random initial positions generally provide reliable simulation results ([Balls and L. R. Frank 2009; Matt G. Hall and Daniel C. Alexander 2009; Landman et al. 2010; Yeh et al. 2013].[...]thereby keeping in mind that simulations with a higher degree of freedom [...] require more particles to obtain similar precision."

They also mentioned the key role of the step size:

> "The step size is limited by the smallest microstructural length scales in a numerical phantom, [...] empirically, the minimally required step size was set to be smaller than a tenth of the cylinder radius [Fieremans, Novikov, et al. 2010]. In general, numerical simulations' accuracy and computational speed have to balance each other out. To better approach the continuous diffusion process, simulations with a small step size are required for accurate results, which is computationally expensive for a given diffusion time."

Look up tables (LUT) - Computational efficiency is crucial to creating a sizeable simulated phantom dictionary. One strategy regarding the Monte-Carlo algorithm is to use look-up table structures [Yeh et al. 2013; Matt G Hall, G. Nedjati-Gilani, and Daniel C Alexander 2017; Ginsburger et al. 2019] consisting of subdividing the global voxel field of view with a periodic 3D grid to reduce the number of collision checks of each particle to the neighboring membranes, thus reducing the overall computational time of Monte-Carlo simulations.

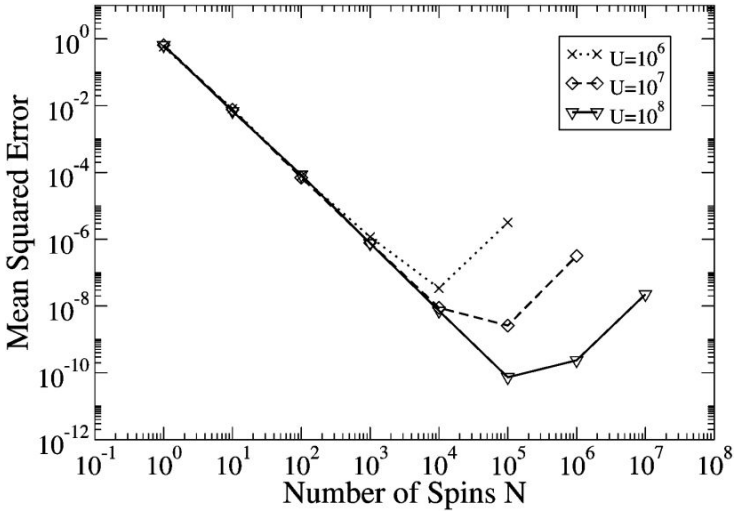


Figure 3.14: *Bias of synthetic diffusion-weighted signals. Worse-case, mean squared differences between signals from the Monte-Carlo and analytical models are shown for sets of 30 simulations with different pairs of N_p and T for several values of complexity U=NT. Results for $U = 10^6$, $U = 10^7$, and $U = 10^8$ for sets of hexagonally packed cylinders. Adapted from [Matt G. Hall and Daniel C. Alexander 2009].*

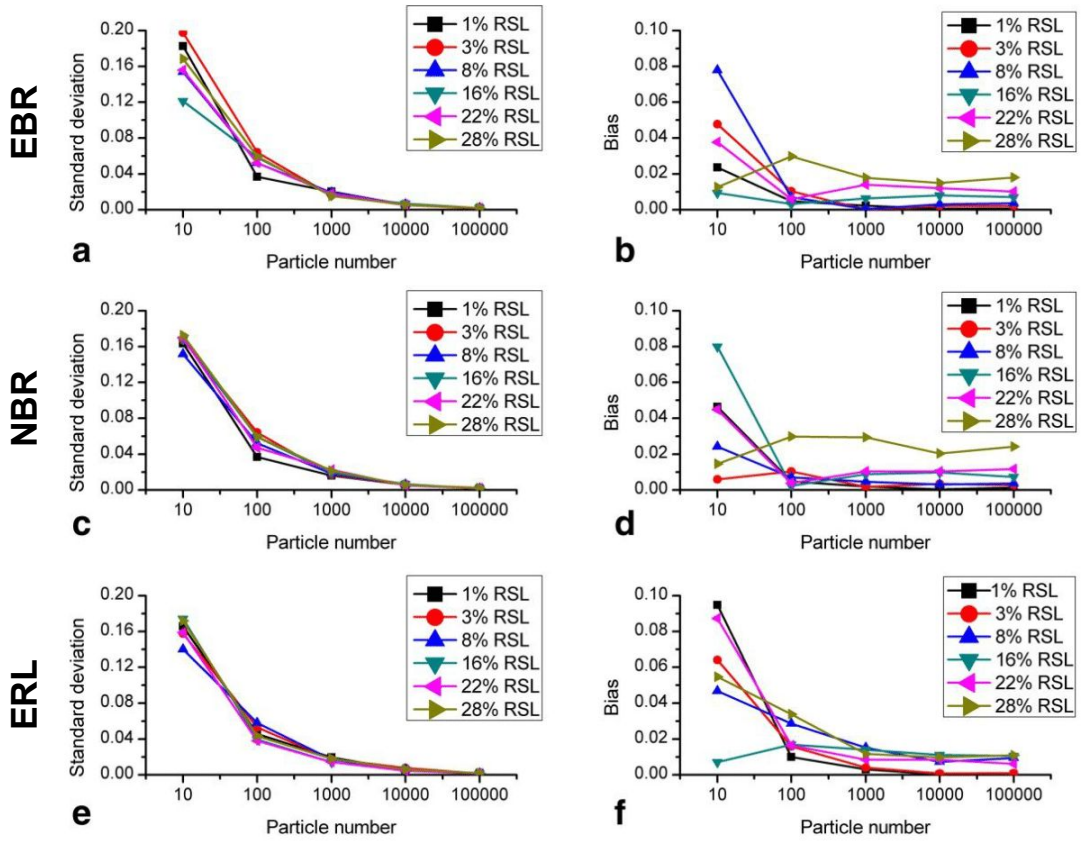


Figure 3.15: *Effects of relative step-length on convergences and biases of synthesized signals. From top to bottom, the figures correspond to EBR, NBR, and ERL, respectively. Different relative step lengths (RSL in legends) are denoted by different logos. Black lines with filled blocks (1%), red lines with filled circles (3%), blue lines with upward-facing triangles (8%), green lines with downward-facing triangles (16%), purple lines with leftward-facing triangles (22%), and brown lines with rightward-facing triangles (28%) represent the results at different RSL for the four methods respectively. Increasing values of RSL increases the bias; an increasing number of particle counts decreases the standard deviation and the bias. Adapted from [Xing et al. 2013].*

Permeability - Different simulators propose to take into consideration membrane permeability or absorption, resulting in more complex implementations requiring, in the case of permeability, to adjust the step size to maintain the particle-density balance between intra and extra-axonal compartments [Fieremans, Novikov, et al. 2010]. Water exchange due to the permeability of cell membranes can be modeled by a transmission probability for the particle to pass through the membrane [Fieremans, Novikov, et al. 2010; Landman et al. 2010; Matt G. Hall and Daniel C. Alexander 2009]. The exchange can happen on the whole membrane surface or at selected areas such as the Ranvier nodes for axons [M. Nilsson et al. 2010].

Alternative to Monte-Carlo simulations - Most diffusion process simulations rely on Monte-Carlo simulations; nevertheless, an alternative has been proposed relying on solutions to the Bloch-Torrey equation [Henry C Torrey 1956] in the case of SPG hypothesis [Neuman 1974]. From a computational point of view, this solution implies a discretization of the spatial domain with a Cartesian grid [J.-R. Li, Calhoun, et al. 2013] or finite elements [Van Nguyen et al. 2014; Fang et al. 2023]. The accuracy of the signal depends on the sharpness of the discretization, which means, in the case of finite elements, the need for a high resolution of meshes. The Spin Doctor simulator proposes synthesizing the dMRI footprint according to the Boch-Torrey formalism.

3.4.3 Synthesis of the dMRI signal

In the case of constant step sizes, the diffusion-weighted MR signal is computed by integrating the phase changes of each step along each particle's trajectory over time. The total phase shift Φ from a particle is given by the following equation:

$$\Phi = \sum_{j=0}^{N} \Phi_j \;\; = \;\; \sum_{j=0}^{N} \gamma < G(t_j, r(t_j) > t, \tag{3.10}$$

with$< \; . \; >$ the scalar product between vectors, N the total number of steps, Φ_j the particle phase shift at step j, $G(t_j)$ the magnetic field gradient at step j, $r(t_j)$ the particle position at step j, t the time step, and γ the gyromagnetic ratio. Summing Φ of all particles gives access to the global phase shift. In the absence of shift, the overall diffusion signal attenuation for M particles at the echo time is given for each gradient direction by:

$$\frac{S(TE)}{S_0} = \frac{1}{M} \sum_{M}^{i=1} cos(\Phi_i) \tag{3.11}$$

dMRI sequence schemes - As the computed phase shift depends on the diffusion sensitization scheme, a good simulator must propose a flexible framework to design various dMRI sequences. Accordingly, the Mcell simulator proposed to model PGSE sequences with tunable b-values and gradient orientations, whereas DMS simulator first added OGSE sequences. In addition, the DMS simulator distributes the direction of the diffusion gradient randomly from a pre-allocated look-up table, which includes uniform and symmetric orientations obtained using an electrostatic repulsion algorithm [Dubois et al. 2006]. To enable a comparison with hardware dMRI settings, synthesized dMRI sequences must keep reasonable values for b-values despite the fact they can artificially depict infinitely high gradient values or infinitely small δ values. To mimic the limitations of actual dMRI hardware, the MEDUSA simulation framework proposes a syntax that enables the design of any new diffusion MRI sequence [Ginsburger et al. 2019].

3.4.4 Computational model

The main application of dMRI simulators is the creation of large dictionaries of simulated tissues and of their dMRI signatures representing all possible microstructural configurations. Such dictionaries allow the training of machine-learning algorithms in a supervised manner to decode the estimation of the underlying microstructural parameters from a diffusion signal.

Nilson [M. Nilsson et al. 2010] proposed a database of 60.000 samples using Monte-Carlo simulations performed in a three-dimensional geometry mimicking myelinated axons with Ranvier nodes following a two-compartment model. The intra-axonal exchange time was investigated for exchange occurring exclusively at the nodes of Ranvier using 10^5 simulated particles. A nearest-neighbor algorithm was employed to estimate microstructure parameters. They concluded that water exchange in the white matter could be an essential biomarker in neurological disease and brain maturation. Another model was proposed by [Nedjati-Gilani 2017], extracting orientationally invariant features derived from the diffusion-weighted signals using a robust random-forest regressor based on reported 12.500 generated unique white matter substrates.

These two studies were successful proofs-of-concepts that need to be further developed, for instance, to enhance the realism of the geometrical representation of the tissues, or the precision of the machine learning algorithm, taking benefit from the development of high-performance computing and deep learning tools over the last decades.

In 2023 [Fang et al. 2023] implemented a neuron mesh generator that simulated grey matter subtracts based on digital neuronal reconstructions archived in (neuromorpho. org) [Ascoli, Donohue, and Halavi 2007]. They mention 1213 neuron meshes that can be randomly combined with a free diffusion compartment to mimic an artificial grey matter sample. Therefore The generation of 1.45 million artificial brain samples is reported. The model relies on the Bloch-Torrey equation solution for simulating diffusion signal decay over time. A Multilayer perceptron (MLP) was trained to estimate the volume fraction corresponding to the soma, neurites, and extracellular compartment. The trained neural network predictions are compared with the SANDI analytical model [Palombo 2020] from in silico neurons originating from the homemade dataset first, then from in vivo data sets stemming from the MGH "Connectome Diffusion Microstructure Dataset" (CDMD) [Tian et al. 2022]. Nevertheless, the imaging data required for the MLP described in this paper are not feasible for most practitioners. Computationally, the authors mention a 10-minutes simulation, for 1,000 simulated dMRI signals and a neuron mesh of around 80.000 vertices [figure 3.12.

To conclude on computational models, the lack of substrate can deter good machine-learning training. Thus, the sampling scheme can be further improved using smart sampling techniques where each new sample is optimally chosen to fine-tune the regression results of the learning algorithm [Pedergnana, García, et al. 2016].

3.5 Conclusion

During this chapter, we presented analytical and computational models aiming to characterize the microstructure of brain tissues. We insisted on the technical aspect of a simulation framework that should rely on a bio-realistic phantom generator, an efficient simulator of the diffusion process, and

an accurate simulator of the dMRI signal attenuation. This thesis relies on the MEDUSA simulator [K. Ginsburger et al. 2018], developed at Neurospin, whose functioning will be developed through the next chapter.

Chapter 4

Microstructure Environment Designer with Unified Sphere Atoms simulator

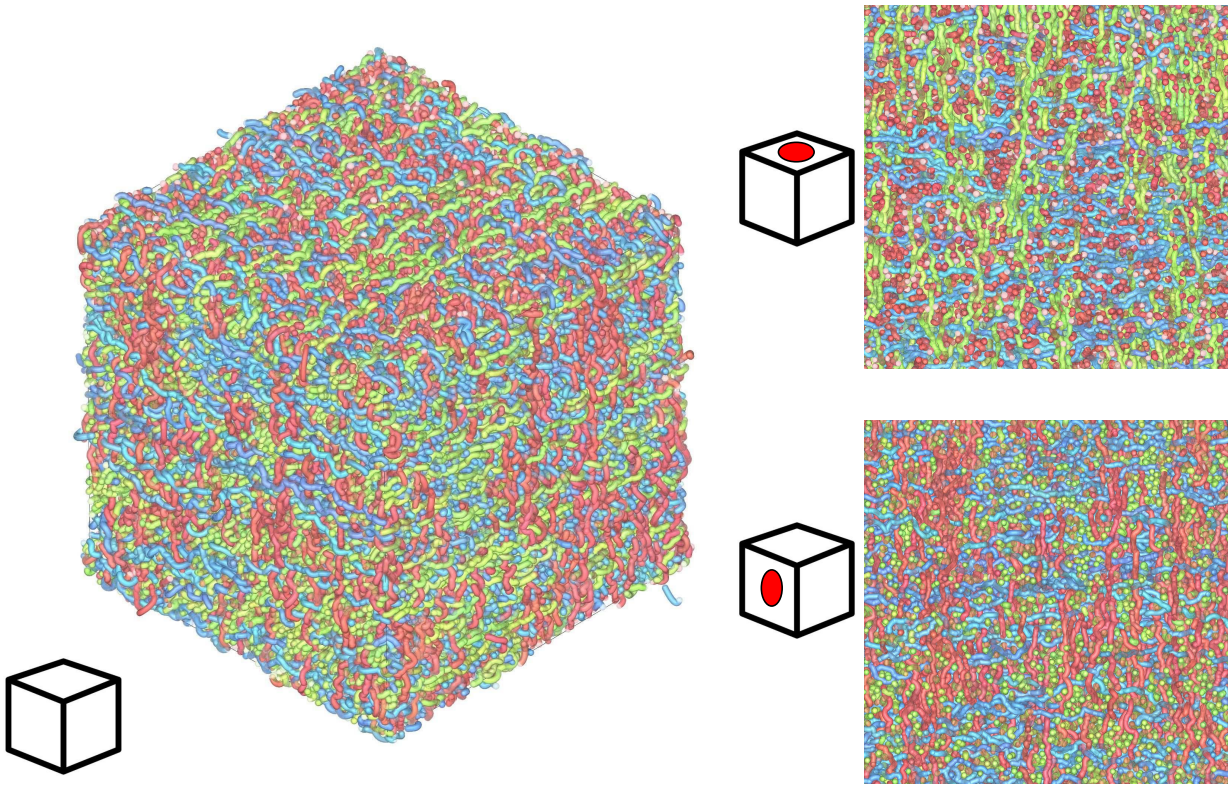


Figure 4.1: *Geometry voxel sample from a MEDUSA simulation of 3 fiber populations oriented respectively along X, Y, and Z axis. A view from the top corner of the voxel is presented (left), and views from the voxel's top and bottom are presented (right). The FOV is [50μm, 50μm, 50μm], the axonal diameter from three populations is 0.8μm and the volume fraction from each fiber population is 20%.*

Chapter outline

4.1 The MEDUSA simulator 85
4.1.1 State of the art 85
4.1.2 Object oriented software design 86
4.1.3 Geometry simulator 86
4.1.4 Diffusion process simulator 103
4.1.5 DMRI sequences and simulator of the dMRI signal attenuation 105
4.1.6 Conclusion 107
4.2 Driving large simulation campaigns 107
4.2.1 Simulation input dictionaries 108
4.2.2 Meta simulation dictionary configuration 112
4.3 Validation 114
4.3.1 MEDUSA reliability regarding analytical solutions 115
4.3.2 MEDUSA validation in the case of complex geometries 116
4.3.3 Conclusion 134

The Microstructure Environment Designer with Unified Sphere Atoms simulator (MEDUSA) [Ginsburger et al. 2019] is a three-component simulator built on some of the achievements of DMS [Yeh et al. 2013; K. Ginsburger et al. 2018] enriched with a new approach to generate the cellular compartmental geometries within the simulated sample through notably the use of a decomposition of individual cells into simple glyphs called ”atoms”. As a full dMRI simulator, MEDUSA’s first component relies on the simulation of a microscopic, bio-realistic virtual tissue geometry composed of white or grey matter cells. The second component replicates the Brownian motion of water, thanks to the use of a Monte-Carlo process [Fieremans and Lee 2018], and finally, the third component consists in the simulation of a dMRI sequence recording the attenuation of the diffusion signal from the movement of water within the geometry. MEDUSA shines through its computational efficiency brought by the use of atoms. Therefore MEDUSA is calibrated to be launched in large simulation campaigns to generate large dictionaries of tissue samples associated with their diffusion MRI footprints, essential to create computational models of the brain microstructure.

In the course of this chapter, we will first present the overall pioneering contribution from Kevin Ginsburger [Ginsburger et al. 2019] for MEDUSA, then describe MEDUSA’s functioning by showing how it successfully depicts one diffusion simulation within a microscopic digital phantom of brain tissue. Furthermore, we will present the pipeline, allowing us to launch large simulation campaigns, and finally, several results supporting the validity of MEDUSA simulations will be highlighted.

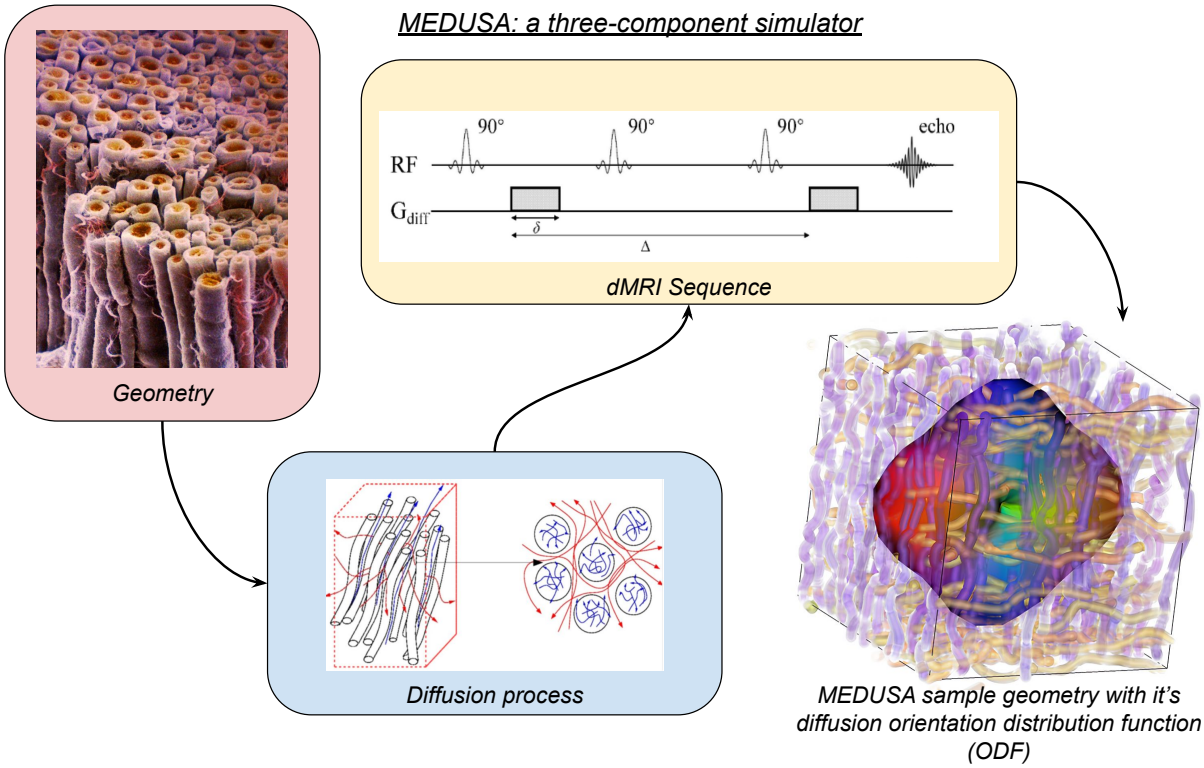


Figure 4.2: *MEDUSA overall structure. MEDUSA is a three-in-one simulator, embedding the simulation of cellular geometry, diffusion process and dMRI signal decay. The sample presented illustrates these simulation modalities by presenting both the geometry and the ODF stemming from it.*

4.1 The MEDUSA simulator

We will first list the milestones reached during the previous work [Ginsburger et al. 2019] on MEDUSA, then introduce the object-oriented design covering its code to finish with the presentation in detail of how the simulator executes a simulation through its three components: the geometry, the diffusion process, and the dMRI signal attenuation [figure 4.2].

4.1.1 State of the art

As mentioned in the introduction, MEDUSA was first presented by Ginsburger in 2019 [Ginsburger et al. 2019]. The article's pioneering work focuses on the design of brain tissue microarchitecture geometries: axonal fibers, astrocytes, and oligodendrocytes for white matter and neuronal somas for grey matter. The geometry procedure relies on the use of "atoms" which are elementary construction objects used as an alternative to meshes. Atom functioning and implementation will be developed later. The simulator proposes to create different complex cell populations crossing each other with additional details, including Ranvier nodes, tortuosities, and beadings.

The article focuses only on the geometry but MEDUSA actually implements the full simulation pipeline till the generation of the dMRI signal attenuation [K. Ginsburger 2019]. It's reported that for a typical white matter phantom characterized by an axonal packing density of 0.7, a mean diameter of 1.0µm, 10^5 random walkers, a diffusion time of 200ms and a time step of 1.0µs, the sample takes 10 min and 27 sec to be generated with a GPU implementation using a Nvidia® DGX station. Different sequences were implemented, such as PGSE or OGSE. Moreover, a machine learning algorithm based

on a random forest approach was developed to quantify the impact on the signal of an axon damaged by beading. The database used to feed the random forest was composed of 4902 samples. The current thesis aims to improve the design of the MEDUSA framework and to exploit its software design to establish a novel computational model of brain white matter microstructure.

4.1.2 Object oriented software design

Object-oriented programming (OOP) is a programming paradigm based on the concept of objects containing data in the form of fields (known as attributes or properties), and code in procedures (often methods). Objects can share features through interfaces thanks to polymorphism. Compared to its 2019 version, one of MEDUSA's new features relies on combining the rigor of OOP, through the C++ language in which it is written, to the anatomical reality of the elements that make up biological white matter tissue. In other words, the exact anatomical classification that allows the biologist to discriminate cells within a tissue is used by the simulator to differentiate objects. Curiously, the polymorphism concept is originally known to be inspired by biology [Oracle 2021].

Design patterns are the different writing schemes allowing the code to be articulated in a way to promote re-usability, flexibility, and readability, and OOP benefits from a good choice of design patterns [Gamma et al. 1999]. In the case of MEDUSA, the code benefits from a high degree of readability because the logic behind OOP is directly inspired by the reality it simulates. Moreover, OOP is optimized to work as quickly as possible and thus meets our big data creation requirements. Finally, OPP facilitates the evolution of the code over time. Indeed, MEDUSA can be improved both in terms of realism and computational optimization, thanks to its careful design that makes the slightest update (creating a new method or class) independent from the existing code and thus not disturb its initial paradigm. In practice, the addition of a new population of cells to the geometry can be simply done by implementing a new class inherited from parent classes.

4.1.3 Geometry simulator

In this section, we will describe the MEDUSA geometry simulator, being the first component of the entire simulation.

4.1.3.1 A generative procedure

The geometry component of MEDUSA has been designed to answer a simple question: what is a virtual tissue mimicking a microscopic brain sample? A virtual tissue is described as a voxel with a specific field of view (FOV) in which we can observe an assembly of different cell populations, filling the space each with a particular volume fraction. The cells can have different natures and are represented based on a decomposition using the packing of atoms. Virtual tissue, cell populations, cells, and atoms correspond all to individual C++ classes representing concepts. Their interface is an aggregation, which means that virtual tissue is made of cell populations themselves made of individual cells themselves made of atoms as shown in [figure 4.3].

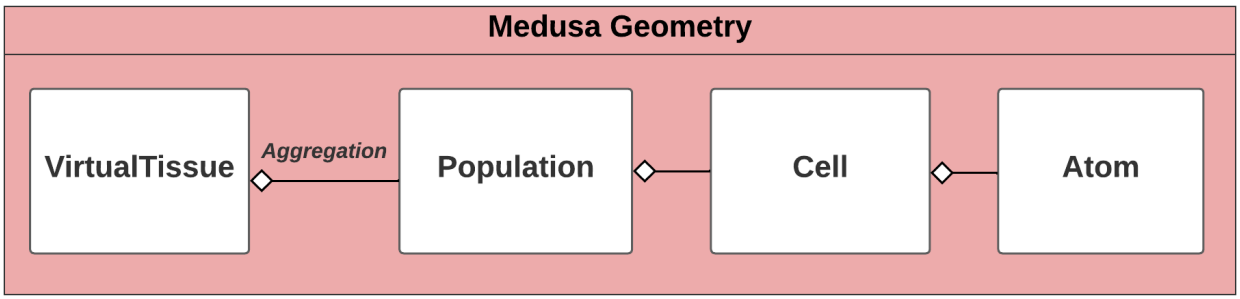


Figure 4.3: *The elementary classes composing MEDUSA geometry framework.*

We will describe the different geometry classes by starting from the smallest element: the atom.

4.1.3.2 Atom: a fundamental generative brick

In the previous chapter, we discussed the limitation of using meshes to represent the cell membranes (see 3.4.1.1) as they significantly impacted the computational efficiency of the simulation and required large amounts of memory for storage. MEDUSA proposes an alternative approach using elementary glyphs called atoms to represent cells. Atoms composing a cell are strongly overlapping each other to provide smooth representation. The MEDUSA simulator today implements two different atom classes:

- Spherical atoms - The name MEDUSA (Microstructure Environment Designer with Unified Sphere Atoms) comes originally from the exclusive use of such atoms [Ginsburger et al. 2019]. Their benefit is to be computationally optimal as each sphere only requires storing four floats in memory: one for its radius and three for its center [figure 4.4]. Interaction between sphere atoms within the 3D space is simple as it only requires comparing distances separating atoms with their respective radius.

- Ellipsoid atoms - Nothing prevents us from trying other atoms' shapes; indeed, in some scenarios, different geometry could help increase the realism of the sample or further compress the cell representation, which improves the computational efficiency of MEDUSA. With that in mind, a new MEDUSA approach was developed to replace sphere atoms with ellipsoid atoms. These new atoms are more complex and computationally heavy to implement as they require a matrix formalism with ten independent parameters to store [figure 4.4]. It's believed that they could help represent axon cylinders more efficiently than spheres. For more details about ellipsoid atoms, the reader should consider looking at my co-worker Anas Bachiri's thesis work, as he mainly developed the new ellipsoid interface within the MEDUSA simulator.

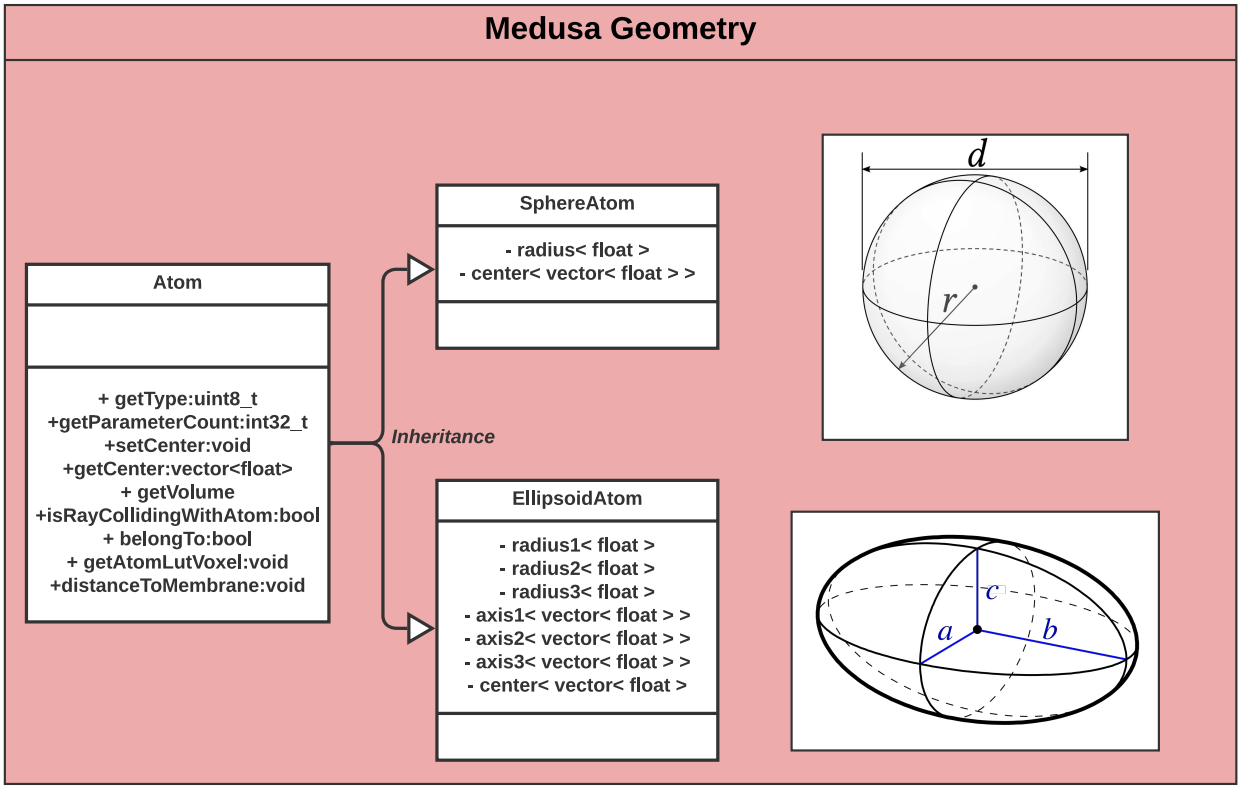


Figure 4.4: *UML structure of the Atom classes within MEDUSA. Sphere Atom is characterized by four independent parameters whereas ellipsoid is characterized by 10.*

In the following sections, all MEDUSA simulations presented will rely on the use of sphere atoms as optimization regarding ellipsoid atoms is not part of this PhD work.

4.1.3.3 Cells

Overlapped atoms contribute to the creation of larger structures: cells. Here is an exhaustive list of cells that can be created using the MEDUSA simulator:

- Axonal fibers - These cells are composed of axons wrapped with a myelin sheath. They mostly intervene in the structure of white matter tissue. As a first approximation, they can be compared to simple cylinders.

- Soma - They represent the cell body and, in particular, the neuron cell bodies. We depict them as a sphere (using a unique sphere atom). They populate gray matter tissues.

- Glial cells - MEDUSA proposes two types of glial cells: oligodendrocytes and astrocytes. They both share common structuring properties such as processes and soma, thus both inheriting from a parent class "Glial cell" [figure 4.5].

- Dendrites - Dendrites are extensions of the cell body of neurons represented as cylinders in MEDUSA.

- Neurons - They were developed in the frame of Anas Bachiri's thesis and stem from real neurons scanned using electron microscopy.

Each of these cells inherits from a virtual class called ”Cell”, thus sharing some common attributes as can be seen in [figure 4.5]. The different design stages of each cell will be further developed in the next section dedicated to ”cell populations,” as the generation of one cell is called at the level of the constructor of the concrete population's classes.

4.1.3.4 Cell populations

A cell population gathers cells of the same type, sharing the same generative distributions of parameters describing their geometries. The creation of a cell population relies on three consecutive steps:

- Initialization - The cells are created one after the other until the target volume fraction of the population within the simulated sample is reached. The cells are randomly created independently of each other, ignoring their respective positions and thus putatively overlapping each other.
- Remove overlap - The initial overlap of distinct cells is unwanted as it is not a realistic representation of the microstructure. The remove-overlap process consists of separating the atoms of the overlapped cells from each other by using repulsion forces. Within the MEDUSA geometry simulation, it's the most computationally expensive step; it will be described in detail later.
- Adding details - The cells are initially created with fewer microstructural details than their final representations which allows it to speed up the remove overlapping procedure. Once all cell populations see their overlaps removed, the representations of cells are improved to add further details, increasing their realism.

Let's now describe the inherited population classed implementing the various cell species:

Fiber population - When generated, fibers are first assimilated to parallel straight cylinders whose origins are controlled by a Watson distribution [Fisher, Lewis, and Embleton 1993] using a method consisting in spreading initialization points across a sphere, wrapping the FOV of the simulation [figure 4.6]. Like seeds for a tree, each of these points constitutes the center of the first atomic sphere of the fiber. All cylinders are then decomposed into overlapping sphere atoms and sphere atoms outside of the surrounding sphere are removed from the final FOV [figure 4.6a].

The direction in which the cylinder grows follows a mean orientation specific to the population and is provided as an input generative parameter. To give a more organic look to the fiber population, a random drift angle from the mean orientation is applied to each fiber iteratively until reaching a desired global angular dispersion within the fiber population [figure 4.6b].

The fiber relies on two inner and outer cylinders representing the axon and the myelin sheath respectively. If we consider the inner axonal sheath has a diameter ”d” and the outer myelinated has a diameter ”D”, then the two diameters are linked by the g-ratio parameter described as follows:

$$g - ratio = \frac{d}{D}. \tag{4.1}$$

To be more bio-realistic, the fiber's cylindrical shape can further be improved by proposing more microstructural details such as tortuosity, beading, or Ranvier nodes. MEDUSA allows to choose whether the target sample contains each of these details or not; thus, as these details are not computationally expensive to compute, the choice to add them or not to a simulation can be driven by the will to

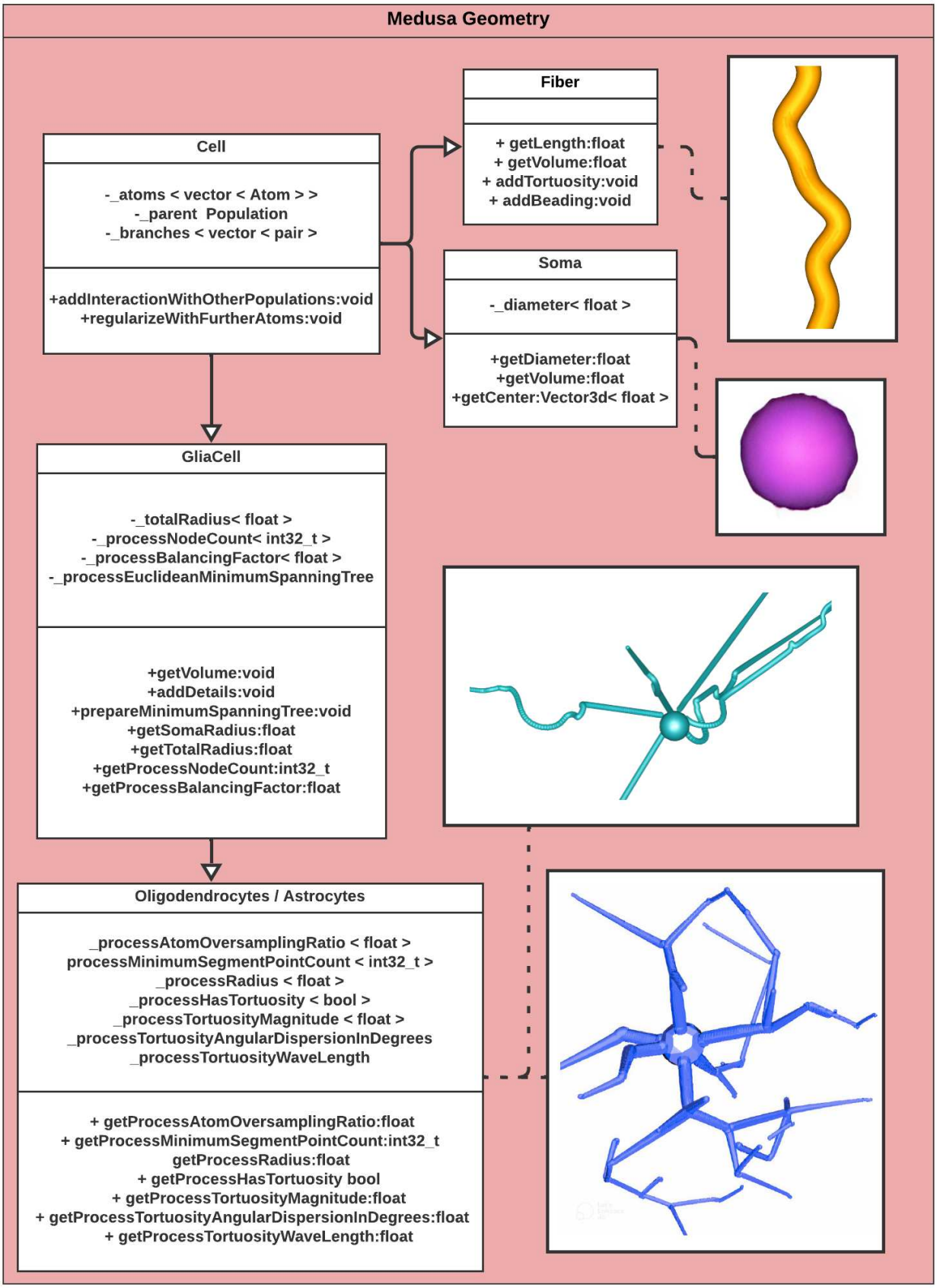


Figure 4.5: *UML structure of classes related to cells. Oligodendrocyte and astrocyte cell classes are merged in the same block for readability but actually represent separated classes.*

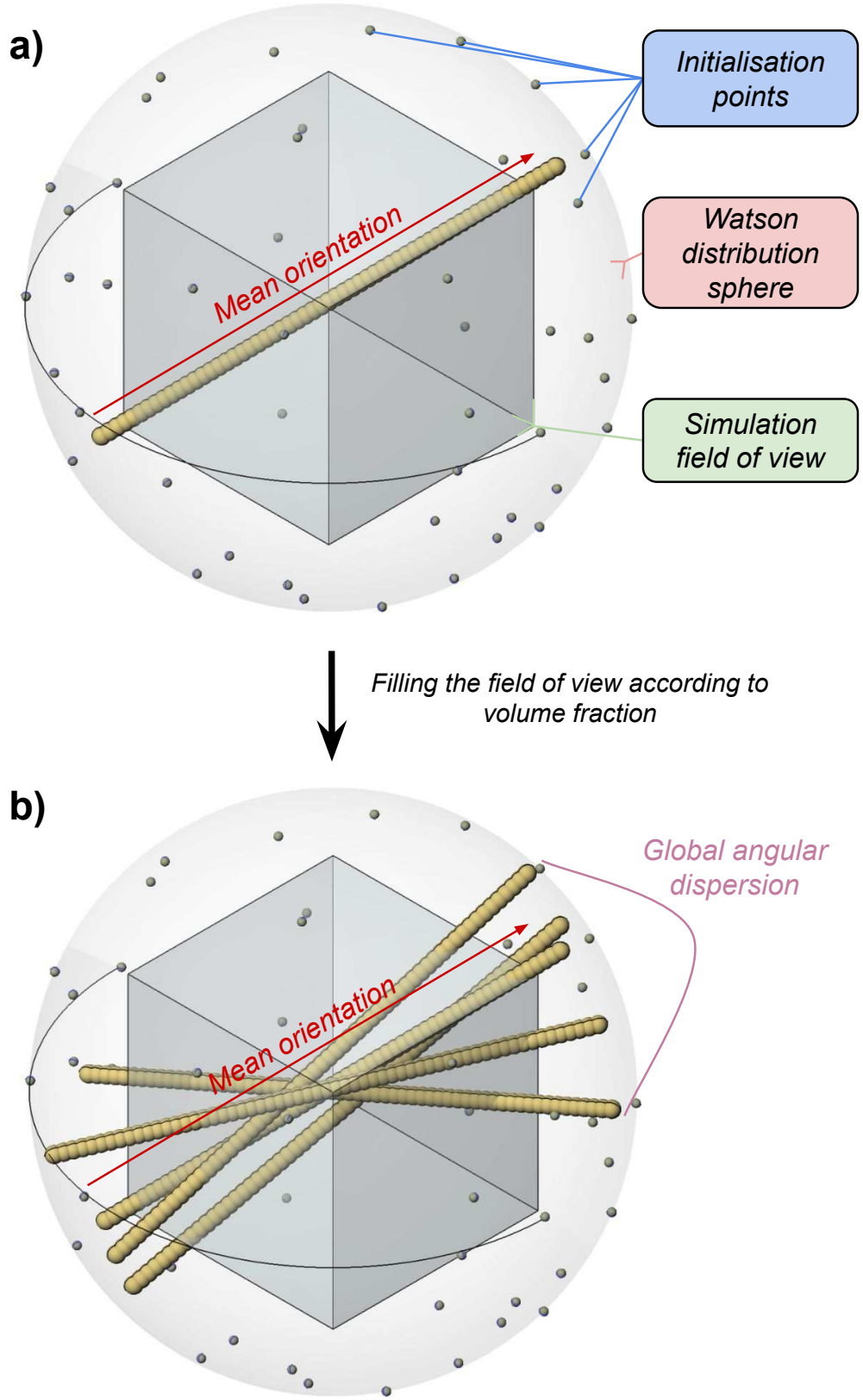


Figure 4.6: *Waston distribution pattern. a) Only one fiber is created, resulting from the successive overlapping of sphere atoms from one side of the Watson distribution sphere to the other. b) Multiple fibers are drawn; they are oriented in the direction of a mean orientation vector, but drift is allowed between fibers, creating a global angular dispersion around the main orientation.*

enhance or discard their contribution to the final MRI signal attenuation. [Figure 4.10] depicts the results of various simulations with different levels of detail for targeted fiber populations.

To be more specific, tortuosity is implemented iteratively as a local drift of spherical atoms belonging to fibers. For each iteration, random atoms from a random fiber are selected to apply a local deformation characterized by a magnitude and a waveform. The wavelength parameter drives the number of atoms the transformation concerns, whereas the deformation magnitude parameter drives its displacement. Wavelength and magnitude both tune a Gaussian deformation [figure 4.7]. The orientation of this deformation is randomly chosen in the plane perpendicular to the fiber at the position of the selected atom.

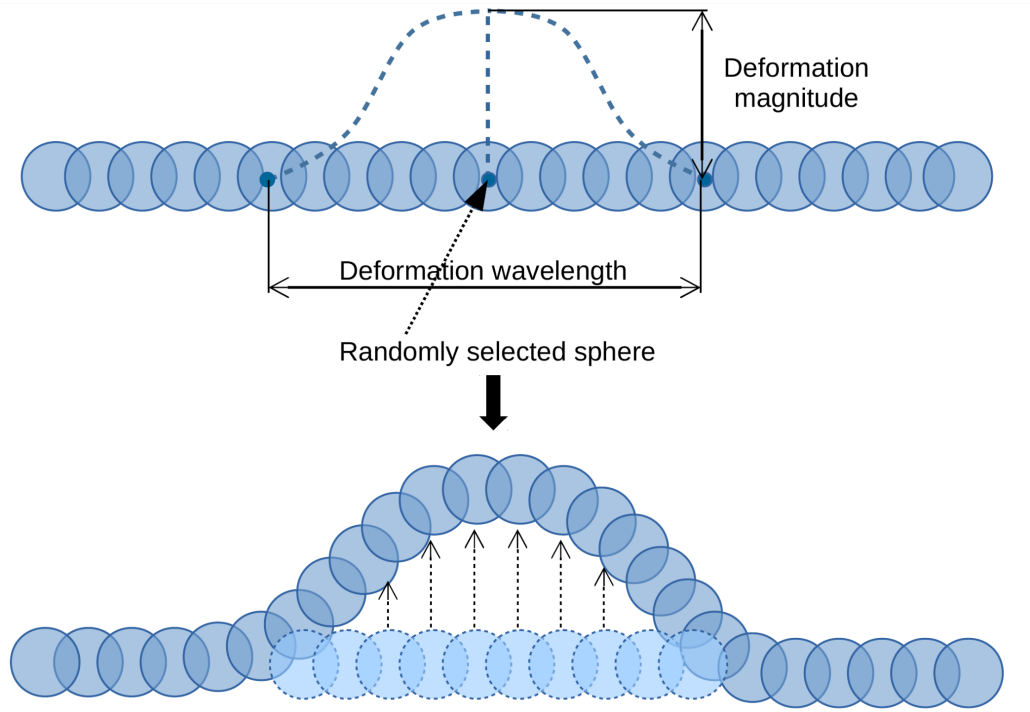


Figure 4.7: *Illustration of the induction of tortuosity within axonal fibers. A Gaussian deformation is applied at a randomly selected sphere in a random direction orthogonal to the main fiber orientation. Adapted from [Ginsburger et al. 2019].*

Inducing axonal beading consists of the application of a local variation of the axon diameters. This feature depends on three parameters: the beading magnitude, the beading width, and the inter-beading length, respectively, driving the radius increasing amplitude, the number of neighboring atoms concerned by the beading, and the space between two consecutive beading patterns in the same fiber. Beading is known to be found in damaged axons [Budde and J. A. Frank 2010] following an ischemic stroke, but it is also found in healthy white matter tissues to a lesser extent, as demonstrated in several studies [Lee, Yaros, et al. 2018; Abdollahzadeh et al. 2017].

Finally, Ranvier nodes can be computed by locally using the same radius value for sphere atoms corresponding to both fiber's outer and inner sheath. Ranvier nodes are distributed at regular intervals corresponding to the inter-nodal length, computed using the maximal conduction relationship 1.1, as illustrated in [figure 4.9].

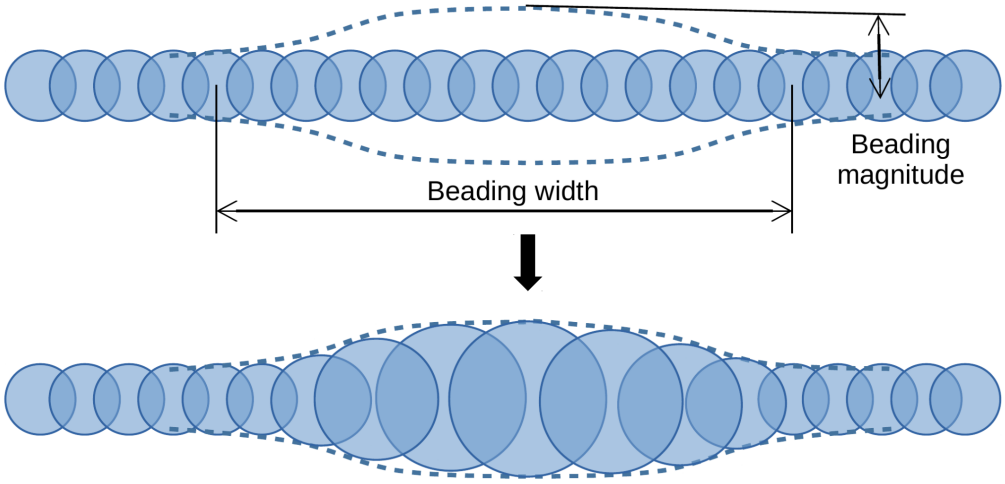


Figure 4.8: *Illustration of the induction of beading within axonal fibers. Fiber spheres are swollen according to a sinusoidal or bell-shaped function to locally increase axonal diameter. Adapted from [Ginsburger et al. 2019].*

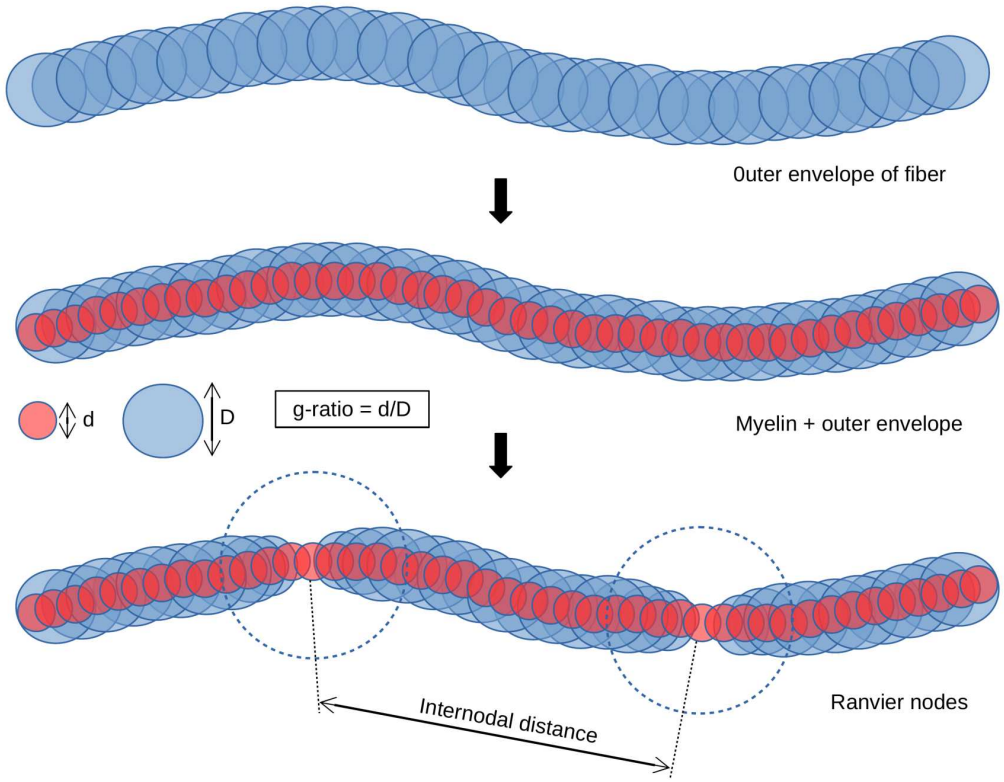


Figure 4.9: *Creation of a myelin sheath and Ranvier nodes on axonal fibers. The illustration presents the diameters d and D from the inner and the outer sheath (myelin) of the axons. The Ranvier node is purposely exaggerated in the illustration for visualization purposes.Adapted from [K. Ginsburger 2019].*

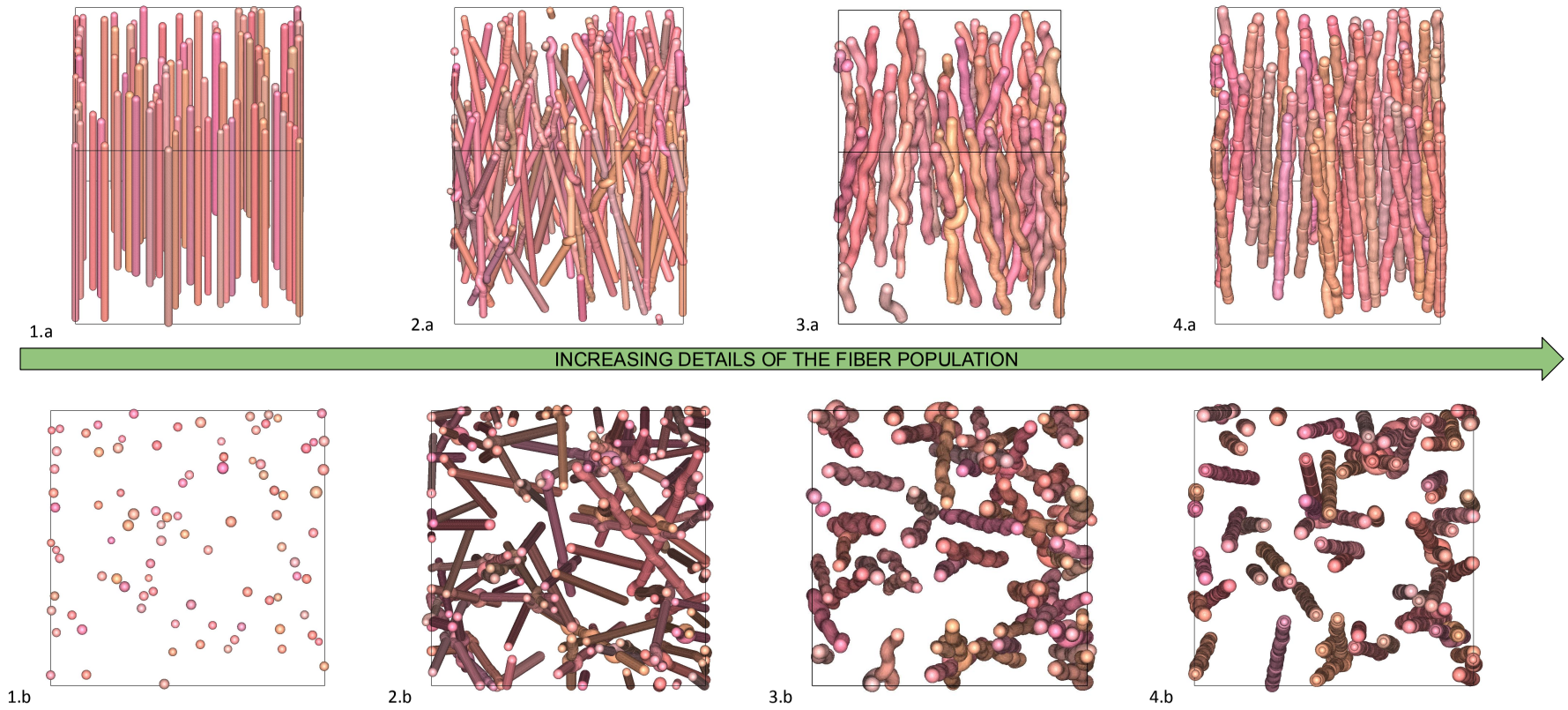


Figure 4.10: *3D rendering of single fiber population geometries with an increasing degree of realism. (1) A straight fiber population generation. (2) Adding global angular dispersion. (3) Adding tortuosity. (4) Adding myelin sheath and Ranvier nodes. (a) Front top view, (b) top view.*

Soma population - Various types of neurons exist (pyramidal, monopolar, multipolar) that are characterized by various shapes. In its original release, the MEDUSA framework proposed a very simple model to represent neurons with sphere atoms whose radii are distributed according to a Gamma distribution. [Figure 4.11] illustrates the laminar structure of the cortex that can be simulated with MEDUSA using for each sample mimicking a specific cortex layer 2 tangential and radial fiber-populations and a soma population. More realistic neuron representations have been proposed in the literature [Marco Palombo 2019; Lee, Fieremans, and Novikov 2021] that all rely on mesh surfaces extracted from electron microscopy data sets. My partner, Anas Bachiri, is currently developing a novel neuron population model in MEDUSA that is based on the use of real neuron models also stemming from electron microscopy scans performed on human brain samples but decomposed into a sphere or ellipsoid atoms to get compressed representations.

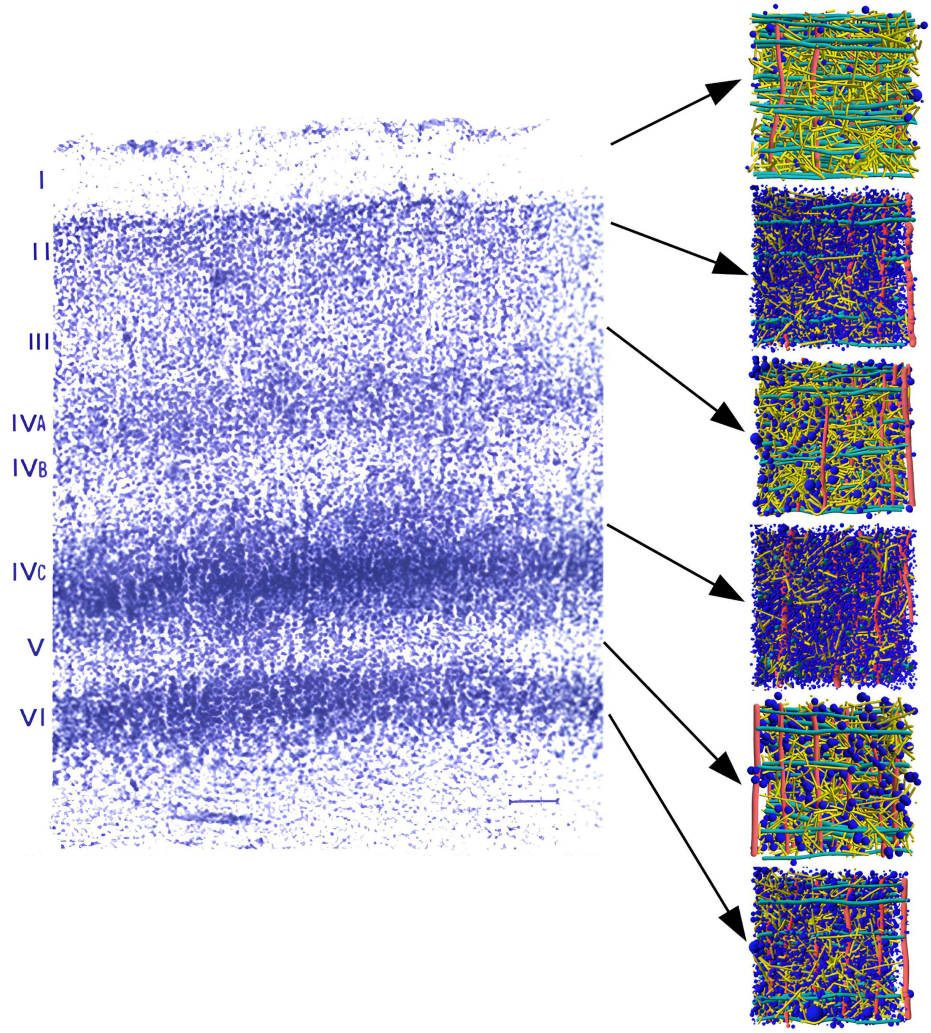


Figure 4.11: *Simplified grey matter phantoms with spherical neural cell soma in dark blue, tangential and radial axons in green and red, and isotropically oriented dendrites in yellow. The constructed phantoms are shown beside a Nissl stain of the visual cortex from [Schmolesky 2016]. Six different samples with distinct soma densities and diameter distributions are aligned vertically to show the possibility of mimicking cortical layers. Adapted from [K. Ginsburger 2019].*

Glial cell populations - Astrocytes and oligodendrocytes are star-shaped cells sharing a similar structure. They are composed of a soma (whose radius is drawn by a Gamma distribution) from which emerge processes giving them their star shape. While processes of the oligodendrocytes don't branch, those of the astrocytes can branch with various degrees of branching.

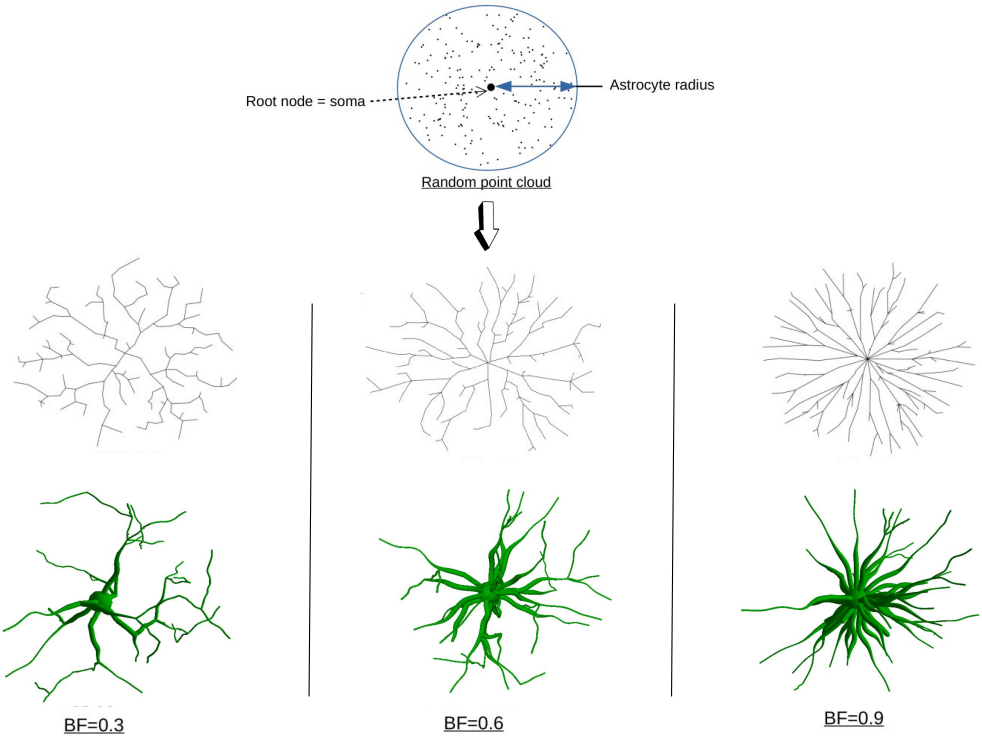


Figure 4.12: *Illustration of the astrocyte creation procedure. On top is the initial node distribution over a sphere centered on the root. Below are 2D and 3D representations of astrocytes for balancing factors equal to 0.3, 0.6, and 0.9. The 3D astrocytes are created with MEDUSA. Adapted from [K. Ginsburger 2019]*

To generate glial cells, the Medusa framework relies on the construction of a minimum spanning tree (MST), considering the soma center as the central node of the MST and randomly seeding points within a sphere centered on the soma and of radius corresponding to the extension of the glial cell. [Figure **??**] illustrates the MST obtained for various branching factors (bf) as well as the evolution of the process diameter progressively decreasing with the geodesic distance from the soma center. Oligodendrocytes are simply obtained with a branching factor set to bf = 1 as they do not depict any branching. The extension of the oligodendrocyte and astrocyte processes is limited by a threshold value. In the case of astrocytes, this threshold limits the radius of the MST point cloud, whereas, for oligodendrocytes, it limits the neighboring axon search radius.

Another difference concerns the number of processes known to be lower in oligodendrocytes than in astrocytes. Last, processes of the oligodendrocytes extend to the myelinated axons and surround their membrane to create a myelin sheath. Furthermore, one oligodendrocyte can be connected to neural axons. Therefore, a specific process has to be added to the generative process to create those connections to fiber populations. This process relies on a dedicated look-up table, which allows the efficient identification of each extremity of a process to the neighboring fibers to be attached to.

MEDUSA simulator proposes different features to increase the realism of glial cells. As for axons, tortuosity can also be added to the glial cells processes, and in addition, the radii of processes decrease when the distance d to the soma increases, according to the following equation:

$$r = r_0 \, e^{-\alpha \, \frac{d}{R}}, \tag{4.2}$$

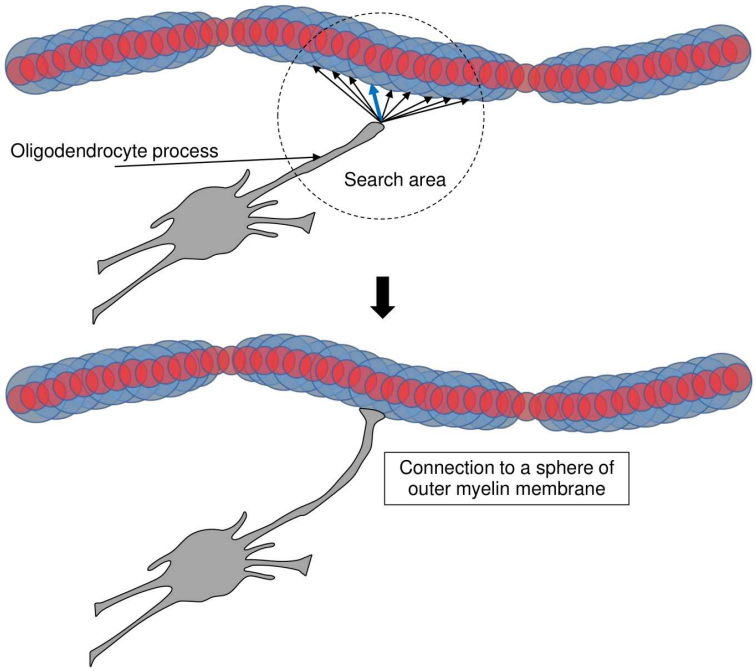


Figure 4.13: *Illustration of the oligodendrocyte creation procedure. Each oligodendrocyte searches for outer axonal membrane spheres within a search area and connects to one of these. The connection algorithm employs a lookup table to make the search procedure efficient. Adapted from [K. Ginsburger 2019].*

where α is an adjustable parameter (typically equal to 2), r the modified radius of the process, r_0 the original radius of the process, R the total radius of the astrocyte (process plus soma) and d the distance to the soma. This feature has been added regarding the observations over astrocytes made in [N. Oberheim et al. 2009].

To conclude on the generation of cell population geometries, we can state that the tremendous amount of details proposed by MEDUSA allows us to generate bio-realistic samples of white and grey matter. Nevertheless, one should remember that offering a high level of geometry details may prove futile. Indeed, any simulated geometry is generated to perform a specific dMRI simulation strongly depending on the hardware limitations of the simulated MRI instrument and some details represented within the voxel might be too small to be detected due to these hardware limitations. In other words, it can be useless to increase the level of realism to some details that will not be encoded in the dMRI signal attenuation. A good example is Ranvier nodes, which are approximately 2µm long and spaced 1mm apart (depending on the axon diameter, see 1.1), and thus constitute only 0.2 % of the axon surface [Giuliodori and DiCarlo 2004]. One could question the actual impact of Ranvier nodes on the final signal decay.

The UML structure dedicated to the population of cells in MEDUSA is presented in [figure 4.14].

4.1.3.5 Overlapped atom

Atom oversampling ratio - MEDUSA relies on overlapped atoms to form its cells. The atom oversampling ratio (AOR) parameter allows the tune of the distance d separating two atoms. In the case of spherical atoms of radius R_{Satom}, the AOR is defined by the following equation:

$$d = \frac{AOR}{R_{Satom}}. \tag{4.3}$$

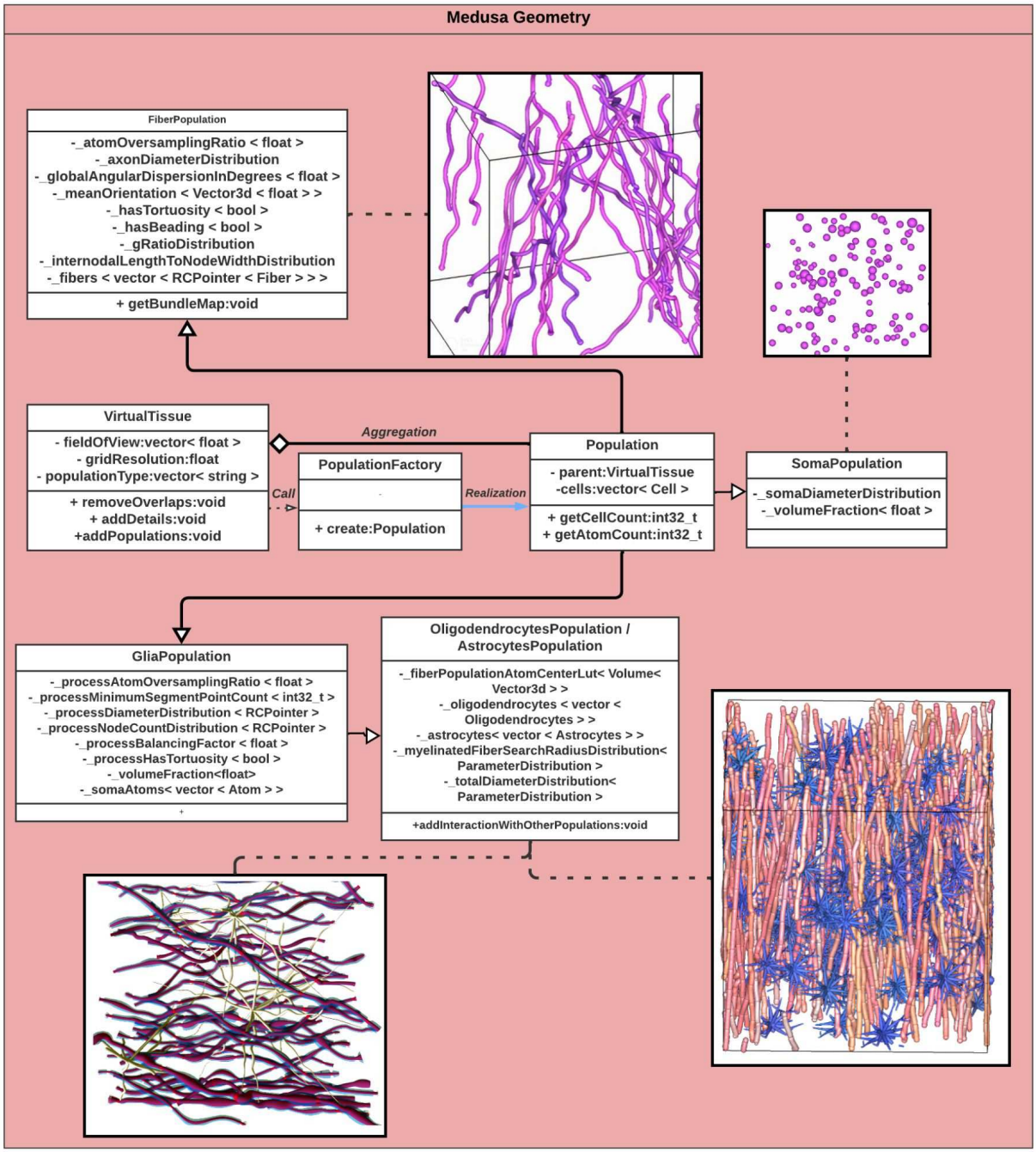
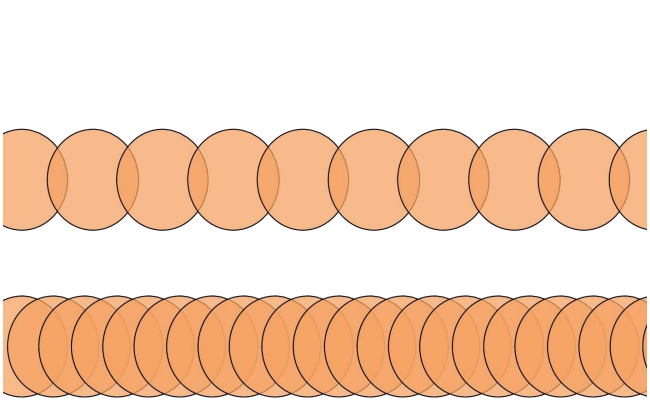
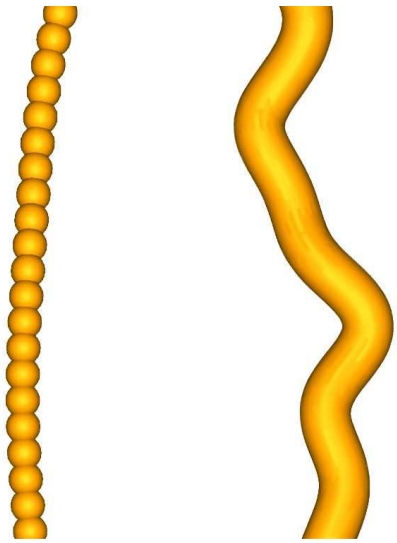


Figure 4.14: *UML structure of classes related to the population of cells. Astrocytes and oligodendrocytes population classes are merged for readability but are actually separated.*



a) 2D diagram of overlapped sphere atoms within a fiber



b) 3D MEDUSA simulation of overlapped sphere atoms within a fiber

Figure 4.15: *Overlapping spheres atoms from the same cell, 2D (a) and 3D (b) illustration. The fiber on the left depicts an oversampling ratio of 1; the fiber on the right depicts an oversampling ratio of 20. Figure visualized with the anatomist toolbox*

Increasing the AOR enhances the quality of the simulation by reducing inter-spherical space. Nevertheless, it increases the number of atoms necessary to create a cell, thus the overall computation time. Different values of the AOR case are presented in [figure 4.15] where we can notice the gaps between sphere atoms for fibers with a small value of AOR. These gaps can truncate the volume fraction obtained for each fiber as it is computed as the volume of the enveloping cylinder.

In [figure 4.16], we can observe three geometries of fiber populations with an AOR fixed to 1, 10, and 20; the wire-framed representation of the simulated substrates on top supports the idea that poorly overlapped atoms create a vacuum for the diffusion process, thus preventing a good simulation of the dMRI signal.

Remove overlap process - When different populations of cells are generated, they fill the FOV to reach a specific volume fraction, taking care not to exceed a total volume fraction of 1.0. However, each cell fills the voxel independently, ignoring the space filled by other cells; thus, the cells overlap. This leads to a not suitable geometric environment and must be tackled by using a remove overlapalgorithm. This algorithm takes benefits from the object-oriented design of the simulator and relies on the definition of lookup tables (LUT) to speed up the process. The overlap from atoms belonging to the same cell must be preserved, as it makes the cell a whole; only the overlap from different cells must be tackled. Thanks to OOP, each atom is associated with a cell ID, thus allowing the discrimination of atoms from different cells. Two sphere atoms are considered overlapped if the distance between their center is smaller than the sum of their radius. The following equation quantifies this overlap:

$$O(s_1, s_2) \;=\; r_1 + r_2 - d(c_1, c_2), \tag{4.4}$$

where s_1/s_2, r_1/r_2 and $d(c_1, c_2)$ respectively represent the sphere atoms, their radius and the distance between their center. When two sphere atoms from different cells are intersecting, a repulsion force is applied to remove the overlap:

$$F(s_1, s_2) = \frac{O(s_1, s2)}{2} \; \frac{c_1 - c_2}{|c_1 - c_2|} \tag{4.5}$$

This force is applied to every overlapped atom from different cells over multiple iterations. If the repulsion of one atom makes it leave the field of view, the repulsion is canceled. A total repulsion force norm (TRF) over all atoms for each iteration of the remove overlap procedure is computed. The repulsion of one atom might make it intersect with another atom, thus increasing the overall global overlap and TRF within the voxel. This phenomenon can be observed in TRF plots [figure 4.16]: at the beginning, TRF is decreasing fast, proof that very close overlapped cells are separating as expected, then the function reaches a minimum which can be explained by the fact that the atoms within the scene found a balance position, making hard for them to be moved without intersecting other elements.

In the case of densely populated geometries (high volume fraction), it's hard to remove the unwanted atom overlapping. Therefore, two mechanisms prevent the algorithm from endlessly applying repulsion forces over atoms: first, if a maximum iteration count is reached, the procedure stops. Second, a minimum total repulsion force threshold is set during the simulation; when the repulsion function reaches it, the remove overlap procedure stops. This threshold is calculated as a percentage of the initial total repulsion force. Nevertheless, as previously mentioned, the function decreases overall, but every repulsion can lead to more overlap, thus increasing the TRF locally. This causes the function to decrease in an undulatory way [figure 4.16]; therefore, when the minimum threshold is reached, the following steps might hide an increase in the TRF. A repulsion force moving average window is applied during the iteration to avoid this issue, ensuring that the minimum threshold has been reached continuously. The management of the TRF presented here is an improvement from [Ginsburger et al. 2019].

The remove overlap procedure is the most computationally expensive step of MEDUSA geometry simulation. Therefore, its code was parallelized using the Kokkos toolkit, which will be detailed in the next chapter. In addition, a Cartesian grid look-up table (LUT) of atoms was constructed to efficiently identify each atom and its neighboring atom [figure 4.17]. The resolution of the LUT grid must be chosen adequately, taking into account the cell dimensions, the local density, and the available memory to reach a compromise (see the validation section 4.3.2.3). In the latest version of MEDUSA, this grid resolution is directly linked with the dimension of the cells composing the geometric sample.

During this thesis, the implementation of the LUT was improved. Before, the total memory space allocated for LUT (LUT_{memory}) was implemented such as :

$$LUT_{memory} = N_{sub-voxels} \; . \; MaxAtomPerLutVoxel \; . \; 4floats, \tag{4.6}$$

where $N_{sub-voxels}$ is the total of sub-voxels and MaxAtomPerLutVoxel is the maximum of atoms found among sub-voxels. The "4 floats" correspond to the memory required to store a sphere atom (one for its radius and three for its center). This approach had the advantage of statically fixing an overall memory for the LUT but was overestimating the number of atoms in the LUT, thus increasing the computation times. The new approach considers the exact amount of atoms from each sub-voxel by dynamically considering the atom count offset brought by each sub-voxel. This feature helped to reduce the computation time overall.

The remove overlap procedure has a beneficial side effect regarding the realism of the geometry. Indeed, the repulsion of atoms from each other creates an additional tortuosity effect on the axonal fiber or

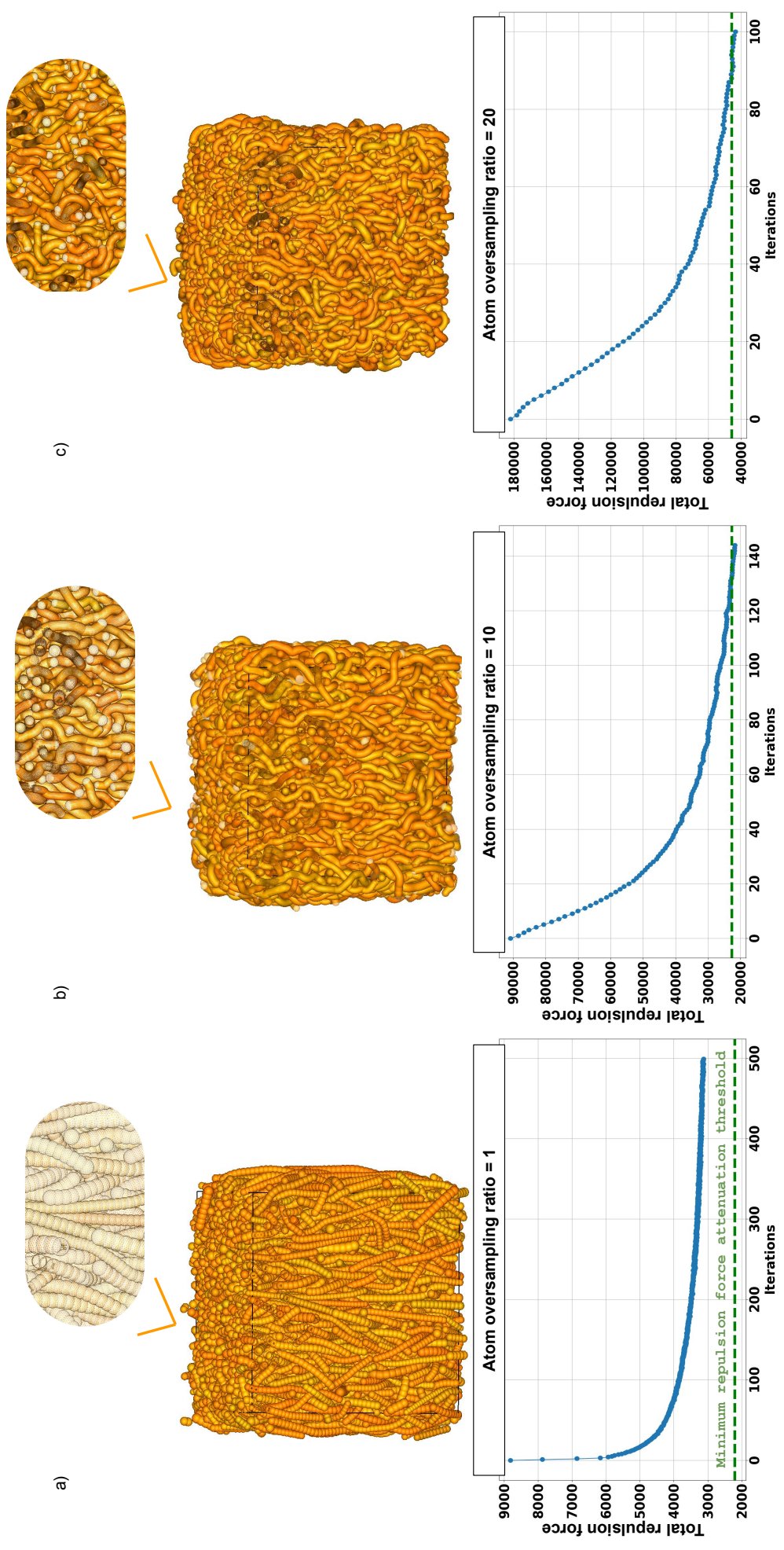


Figure 4.16: *Illustration of three simulated fiber populations and their associated total repulsion force norm function over iterations. Above the FOV, a zoom is applied to the sample and shows the "wire-frame" structure of the elements. a) The atom oversampling ratio is set to 1, and the total force norm computed is inferior to the other scenario. The function never reaches the threshold value, and the remove overlap procedure stops after reaching the maximum iteration set to 500. b) and c) The atom oversampling ratio is respectively set to 10 and 20, the function reaches the minimum threshold and stops.*

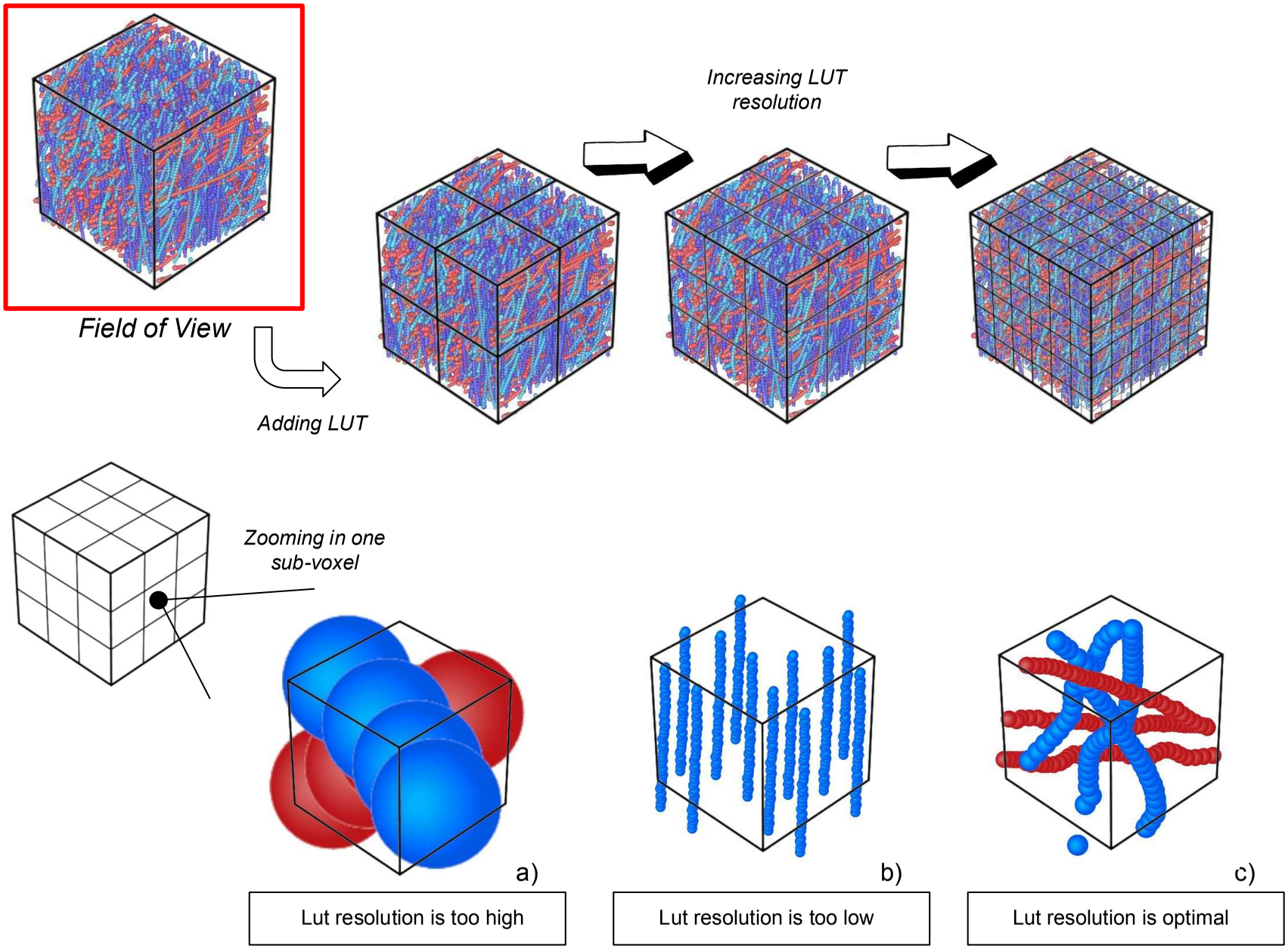


Figure 4.17: *Illustration of the lookup table in MEDUSA. At the top, we can see the different resolutions of the LUT cartesian grid. On the bottom, we can observe different scenarios where the resolution of the LUT faces different dimensions of the sphere atom. a) and b) choose the resolution grid poorly by setting it too small or too big. c) The grid is carefully implemented to consider an optimized number of atoms.*

glial cell processes, thus bringing a more bio-realistic effect to the geometry.
Sphere atom interpolation - During this thesis, an additional feature called ”smoothing” has been implemented for MEDUSA. After applying the remove overlap procedure, some cells see their original shape disturbed, creating, in some cases, gaps between consecutive atoms from the same cell. The interpolation algorithm fills these gaps as the space between atoms is completed with atoms following the atom oversampling ratio parameter. This step contributes in enhancing the realism of the geometry as the gap within the cell might bring errors during the diffusion process simulation, with particles crossing those gaps.

After detailing the MEDUSA geometry simulator, let's see how virtual tissue samples can be used to simulate the water diffusion process.

4.1.4 Diffusion process simulator

To simulate the Brownian motion of water, MEDUSA relies on a Monte-Carlo approach randomly distributing and moving particles within MEDUSA simulated geometries. The particles can be distributed within the geometry in three different ways: only in the intracellular space, only in the extracellular space, and everywhere (both intra- and extracellular space) [figure 4.18].

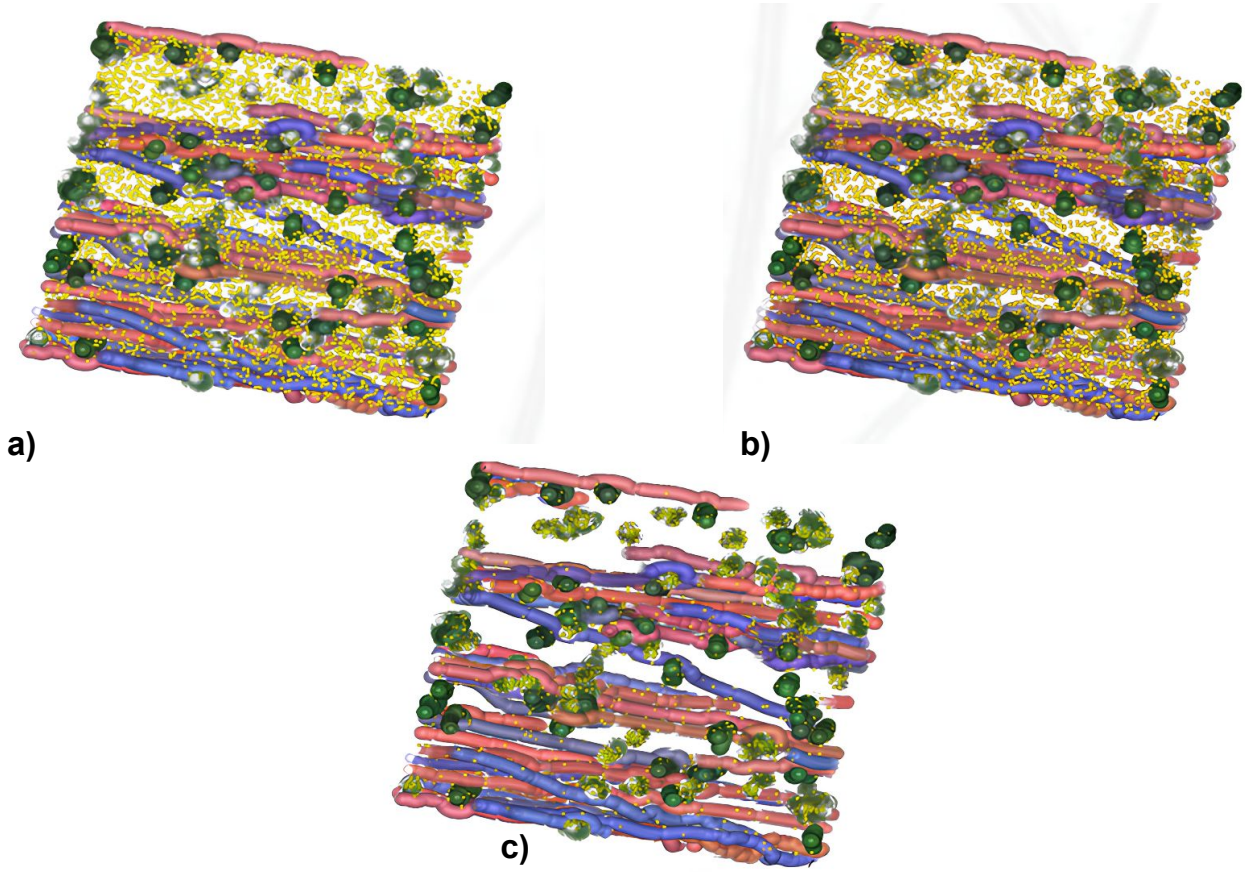


Figure 4.18: *MEDUSA samples of geometries associated with distributed particles cloud. a) The particles are distributed all over the FOV. b) The particles are only distributed in the extracellular space. c) The particles are only distributed in the intracellular space.*

The MEDUSA diffusion process simulator takes into consideration the permeability of membranes. During a collision, the particle has a probability of crossing the membrane according to a permeabil-

ity ratio that can be tuned for every population cell. If the permeability is set to 0.0, the cell is impermeable, whereas if it is set to 1.0, the cell becomes inexistent in terms of diffusion.

On the one hand, particles are associated with a specific ID corresponding to the compartment in which they are located. On the other hand, each atom has its own ID as well as a couple of labels corresponding to the cell and population to which it belongs (the extracellular compartment has its own identification index). With such a double particle/atom identification system, it's possible to track the path of particle trajectories during the Monte-Carlo simulation.

Once initialized, particles move iteratively with a constant step length in a randomly drawn direction until the echo time is reached. As introduced in the previous chapter 3.4.2, the choice for a constant step length is not exact regarding the physical diffusion phenomenon, but it remains a valuable approximation supported by the law of averages. At each step, a particle can face two distinct scenarios:

- In the first scenario, the particle is located in the extracellular space.
  - If the step yields a situation where the particle remains in the extracellular compartment, the particle step is accepted.
  - If the step yields a situation where the particle moves to the intracellular compartment, a collision with a membrane occurs, depending on the membrane permeability. If the permeability test allows it, the particle penetrates the membrane. Otherwise, the particle collides with the membrane.
- In the second scenario, the particle is initially located in the intracellular space.
  - If the step yields a situation where the particle remains in the same atom, the step is accepted.
  - If the step yields a situation where the particle moves to another atom belonging to the same cell, the step is accepted.
  - If the step yields a situation where the particle moves to the extracellular space, the particle undergoes a permeability test. If it's positive, the particle penetrates the membrane. Otherwise, the particle collides with the membrane.
  - If the step yields a situation where the particle moves to another atom belonging to another cell, the particle undergoes two successive permeability tests corresponding to the crossed membrane. Depending on the results, the particle collides with the first membrane, the second membrane, or none of them.

The MEDUSA diffusion simulator proposes two types of membrane collision: equal step length random leap (ERL) and elastic bouncing reflection (EBR), both introduced in the previous chapter (see 3.4.2). The use of atoms helps to track particles during the Monte-Carlo simulation and participate in the computational efficiency of MEDUSA. However, it involves one approximation: MEDUSA cells are composed of overlapped atoms. Thus, the particle may leave the cell and re-enter it as it moves through the gap created by the overlapped atoms, knowing that this gap size directly correlates with the AOR [figure 4.15]. Nevertheless, with a correct AOR value, this approximation has a minimal impact because the number of particles facing such a situation is considered to be negligible.

During Monte-Carlo simulations, only the last position of a particle is kept in memory, to avoid memory overload during the process. However, if the user wants to save particle trajectories, there still is a possibility to specify before simulating a percentage of particle trajectories to save, as shown in [figure 4.19]. Monte-Carlo diffusion simulation can involve large numbers of particles that could still lead to memory overload; to prevent this risk it's possible to split the simulation into several sessions, each session being responsible for the simulation of the pseudo-random trajectories of a subset of the total particle set.

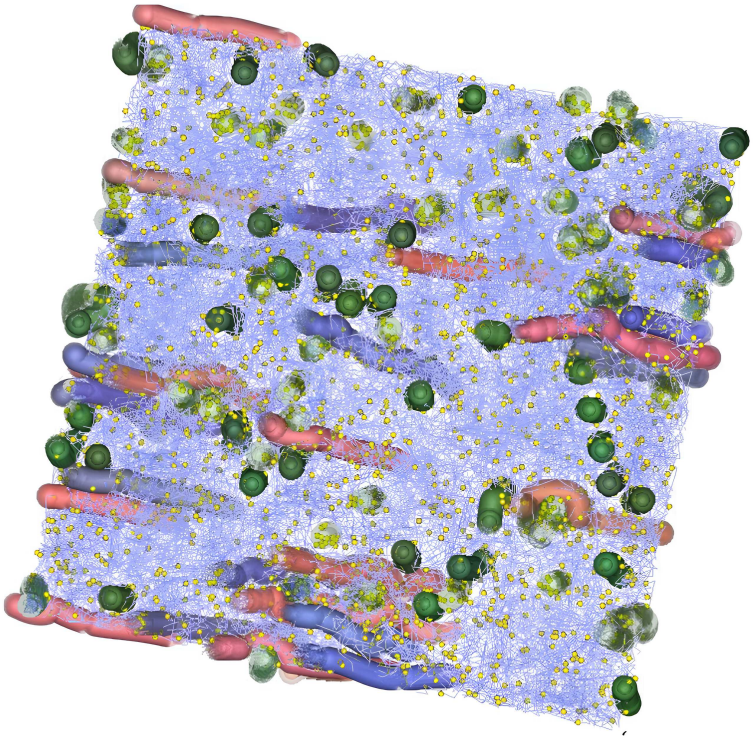


Figure 4.19: *Particle trajectories simulated using the MEDUSA diffusion process simulator. The simulation contains 200 000 particles, moving all over the FOV, during an experiment longing 130 000μs with a time step of 10μs.*

As the remove overlap procedure from the geometry simulation, the diffusion process benefits from a parallelized code based on Kokkos implementation and LUT. The LUT dedicated to diffusion was improved in the same way as the one dedicated to geometry, thus improving the diffusion simulation performances compared with the previous version of MEDUSA. An UML diagram of the MEDUSA diffusion simulator code is provided in the appendix of this thesis for more details concerning its Kokkos implementation.

In the next section, we will focus on the use of MEDUSA diffusion simulations of the dMRI signal attenuation corresponding to various dMRI sequence schemes.

4.1.5 DMRI sequences and simulator of the dMRI signal attenuation

The diffusion-weighted MRI signal attenuation is synthesized by computing the phase shift from every particle for each time step (see chapter 3 3.4.3) induced by the presence in the MRI sequence of diffusion pulses used specifically to sensitize the MRI signal to the diffusion process. Different schemes of MRI sequences exist, such as PGSE and OGSE. It is a major interest for a dMRI simulator to be able to address any of these sequences as for a given simulated geometry and diffusion process,

one would like to simulate the dMRI signal attenuation for various MRI hardware and for various dMRI sequences. These observations have pushed us to propose a generic design for dMRI sequences, inspired by the previous DMS development done in the team [Yeh et al. 2013] to be able to develop any sequence scheme generically.

Within MEDUSA, sequences, and their common features are built by the generic class MRI Sequence; these features include parameters such as echo time, gradient orientations/magnitudes, and methods such as phase shift and b-value computation. Magnitudes and orientations can be filled manually, nevertheless, an automatic approach is recommended, considering that a large number of different gradient orientations need to be specified during a dMRI acquisition. Therefore, taking inspiration from DMS [Yeh et al. 2013], MEDUSA proposes to automatically select the direction of diffusion gradient by randomly choosing it from a pre-allocated lookup table, which included uniform and symmetric orientations obtained using an electrostatic repulsion algorithm [Dubois et al. 2006].

To imitate the limitations of real MRI hardware, a slew rate is applied to sequences, which limits the gradient magnitude shift quickness. The existence of the slew rate implies gradients to be shaped as trapezoids rather than rectangles in the case of PGSE or trapezoid OGSE.

Inheriting from MRI Sequence class, PGSE Sequence and OGSE Sequence MEDUSA classes propose to describe PGSE and OGSE dMRI sequences. PGSE are built by assembling two trapezoids of length δ, of respective amplitude G and -G and separated by a time Δ, as depicted in [figure 4.20]. Due to their oscillating shape, OGSE can be built using a sine/cosine function or trapezoidal sine/cosine function [figure 4.20]. Δ value is still necessary to specify for OGSE to separate the first pulse gradient from the second; nevertheless, the δ value is replaced by the gradient period count, giving access to the frequency of the OGSE.

The expression for bValue $b = \gamma G^2 \delta^2 \ (\Delta - \frac{\delta}{3})$ (see chapter 2 **??**) is prohibited as it only applies in the case of perfectly rectangular shaped gradient, thus as its not our case, bValues are computed by following its more general integral expression derived from the Bloch-Torrey equations [J. Xu, Does, and Gore 2009; Henry C Torrey 1956].

The MEDUSA class MRI Sequence Pool allows us to acquire several sequences with different b-values and orientations during the Monte-Carlo simulation. Each gradient contribution to the phase shift is computed and stored in a specific gradient vector before the diffusion process. During the diffusion process, every particle's trajectories are computed with the gradient vector to update the global phase shift. Each b-value from each sequence allows a multiple diffusion contrast characterization. The gradient vector's size impacts the simulation's computational efficiency as it needs to integrate all particle's phase shifts for each step; thus, we observe an increase in simulation time for sequences with a high amount of orientations.

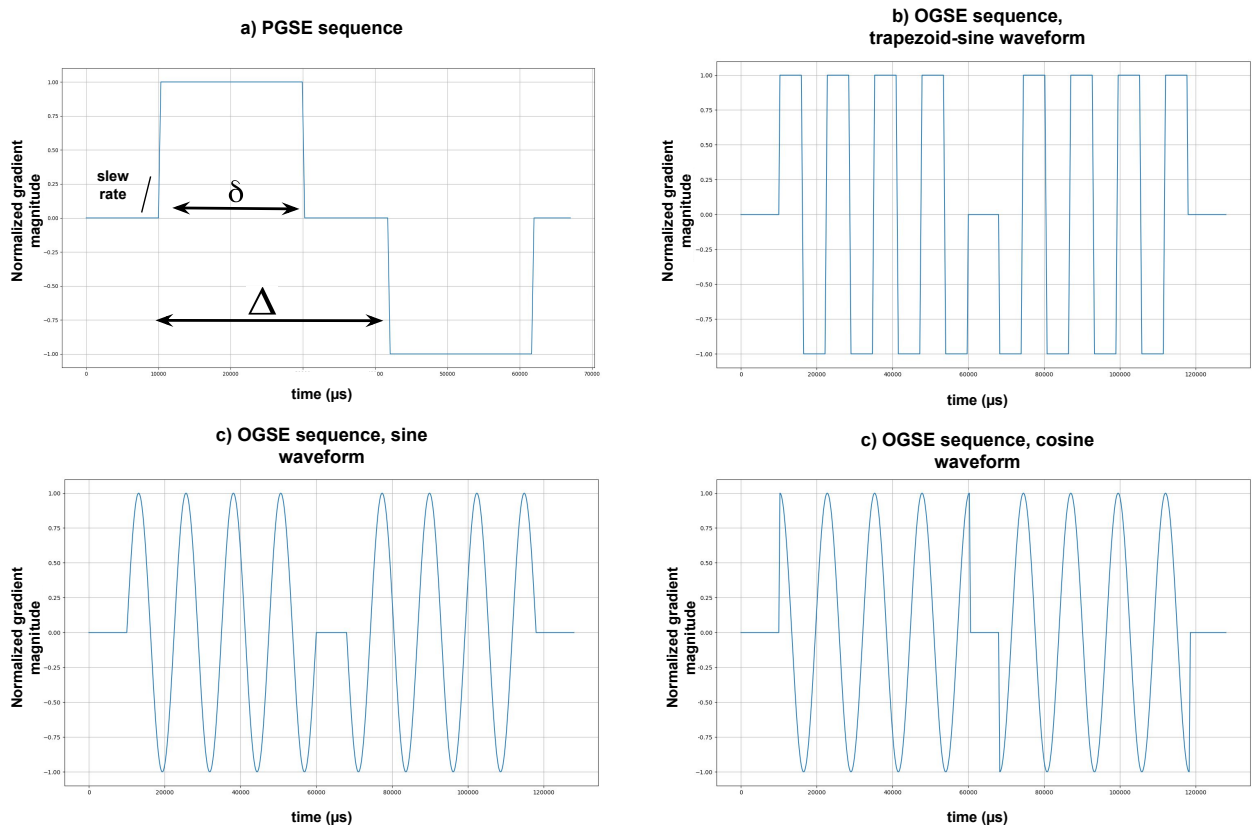


Figure 4.20: *Exemples of dMRI sequences simulated with MEDUSA. a) PGSE sequence, b) OGSE sequence with trapezoid-sine waveform, c) OGSE sequence with sine waveform, and d) OGSE sequence with cosine waveform.*

An UML diagram is provided in the appendix of this thesis for more details concerning the implementation of dMRI sequences classes within MEDUSA.

4.1.6 Conclusion

MEDUSA is an all-in-one simulator beneficiating from code optimization due to the massive use of OOP and the parallelization of the remove overlap procedure and diffusion process. The code from the MEDUSA simulator is open source and available on framagit.org [FRAMAGITMEDUSA 2024]. In the next section, we will introduce the implementation formalism surrounding the input parameters of a MEDUSA simulation.

4.2 Driving large simulation campaigns

MEDUSA aims to be an efficient and user-friendly simulator with a simple input formalism. However, the simulation parameters are numerous and diverse as we roughly count 26 parameters for a fiber population, 28 for an astrocyte population, 27 for an oligodendrocyte population, 9 for a diffusion process, and 9 for each implemented dMRI sequence, which gives an overview of the complexity of the simulator. For a single all-in-one MEDUSA simulation, 3 configuration files need to be provided:

- A geometry JSON file tuning the generative geometry model.
- A diffusion process JSON file tuning the Monte-Carlo simulation.

- A Mri sequence JSON file tuning the set of dMRI sequences.

The Python JSON format has been chosen for these configurations because of their simple structures, allowing a hierarchical description of the parameters with Python dictionaries. MEDUSA is embedded in the Ginkgo software toolbox, which offers the possibility to launch the simulator from the command line or an inherited Gingko Python module.

4.2.1 Simulation input dictionaries

In the following sections, the JSON inputs related to the simulation of geometry, diffusion, and dMRI sequences are introduced.

4.2.1.1 Sructure of the geometry input dictionnary

Within the JSON dictionary dedicated to geometry, some information concerning the whole sample parametrization is placed beforehand as follows:

```
Geometry_Dictionary =
{
  "apply_add_details": 1,  # Boolean indicating if details are integrated
                           # to the cell's geometry
  "apply_atom_regularization": 1,  # Boolean for interpolation procedure to
                                   # fill gaps with further atoms
  "apply_remove_overlaps": 1,   # Boolean indicating if the remove overlap
                                # procedure is integrated into the simulation
  "field_of_view": [-10.0,    # [ -x, +x, -y, +y, -z, +z ] dimensions of
                    -10.0,    # the field of view
                    -10.0,
                    +10.0,
                    +10.0,
                    +10.0],

  "grid_resolution": 0.25,  # The resolution of the look-up table grid
                            # associated with removing overlap procedure
  "maximum_atom_count": 5000000,  # The maximum number of atoms carried by the simulation
  "maximum_force_norm": 0.25,  # The maximum force threshold applied to
                               # atoms during the remove overlap procedure
  "minimum_repulsion_force_attenuation_percentage": 10.0,   # Threshold after which the
                                                            # remove overlap procedure stops
  "repulsion_force_moving_average_window_size": 10,
  "repulsion_force_stddev_percentage_threshold": 1.0,
  "atom_overlap_solver_maximum_iteration_count": 500,  # Number of iteration after which
                                                       # the remove overlap procedure stops
  "populations":   # Dictionnary listing all populations within the simulation
  {
    "fiber-population-01":
    {
      [ ... ]
    },
    "fiber-population-02":
    {
      [ ... ]
    },
    "astrocyte-population-01":
    {
      [ ... ]
```

```
    }
  }
}
```

Each cell population is described in individual sub-dictionaries. The example of a fiber population JSON dictionary is presented as follows:

```
"fiber-population":
{
  "type": "fiber-population",  # Label of the population, if it's unknown, an error occurs
  "parameters":
  {
    "atom_oversampling_ratio": 10.0,
    "global_angular_dispersion_in_degrees": 5.0,
    "has_beading": 0,
    "has_tortuosity": 1,
    "has_myelin_sheath": 1,
    "mean_orientation": [ 0.0, 0.0, 1.0 ],  # [x, y, z] coordinates of fiber
                                            # population mean orientation
  "axon":  # Dictionary listing the axon parameters from the fiber population
  {
    "beading":  # Dictionary listing the parameters describing beading
    {
      [ ... ]
    },
    "diameter_distribution": # Dictionary listing the parameters
                             # describing axon diameter distribution
    {
      [ ... ]
    },
    "tortuosity": # Dictionary listing the parameters describing tortuosity
    {
      [ ... ]
    }
  },
  "myelin_sheath":  # Dictionary listing the myelin sheath
                    # parameters from the fiber population
  {
    "g_ratio_distribution":
    {
      [ ... ]
    },
    "has_ranvier_nodes": 1,  # Boolean indicating if Ranvier nodes are integrated
                             # into the fiber geometry
    "internodal_length_to_node_width_distribution":
    {
      [ ... ]
    }
  },
  "volume_fraction": 0.5  # Volume fraction from the fiber population,
                          # the total volume fraction cant exceed 1.0
}
```

The same formalism is applied to astrocytes, oligodendrocytes, dendrites, or soma populations such that the parameters defined are adapted to the geometry of each cell.

4.2.1.2 Structure of the diffusion input dictionary

The structure of the JSON dictionary dedicated to the diffusion process parametrization is presented as follows:

```
Diffusion_Dictionary =
{
  "particle_initialization": "everywhere",  # Choice among intracellular,
                                            # extracellular and everywhere
  "particle_to_membrane_interaction_type" : "equal-step-length-random-leap",
                                            # Elastic bouncing reflection exists as well
  "session_count": 4, # Number of sessions dividing the total
                      # particle computed during the simulation
  "particle_diffusivity_in_um2_per_us" : 3e-3, # For water at 37 degrees Celsius
  "particle_count": 1000,
  "grid_resolution_in_um" : ( 1.0, 1.0, 1.0 ), # Grid resolution from lookup
                                               # table dedicated to Monte-Carlo simulation
  "time_step_in_us": 10.0,
  "total_duration_in_us": 130000.0,  # Diffusion time
  "particle_backup_percentage": 10.0 # Percentage of particle trajectories
                                     # saved for visualization purposes
}
```

4.2.1.3 Structure of the dMRI sequence dictionary

The structure of the JSON dictionary dedicated to dMRI sequences is presented as follows:

```
dMRI_Sequence_Dictionary =
{
  "sequence-pgse-delta12-DELTA43-b0200" :
  {
      "type": "pgse", # PGSE sequence type
        "parameters" :
        {
          "echo_time_in_ms": 75.0,
          "time_offset_to_diffusion_module_in_ms": 9.85,
          "little_delta_in_ms": 12.0,
          "big_delta_in_ms": 43.3,
          "maximum_gradient_slew_rate_in_tesla_per_m_per_s": 200,
          "gradient_time_resolution_in_us": 10.0,
          "gradient_magnitudes" :  # In mT/m
            (
              ( 22.9 )    # All gradients are set with the same magnitude
            ),
          "gradient_orientations" :
            (
              200    # Orientations are randomly selected from a pre-built orientation table
            ),
          "noise_standard_deviation" : 0.0   # Signal attenuation corruption
        }
  },
  "sequence-ogse-f80-b0300" :
  {
    "type": "ogse", # OGSE sequence type
    "parameters" :
    {
      "echo_time_in_ms": 128.0,
      "waveform_type" : "trapezoid-sine",
      "time_offset_to_diffusion_module_in_ms": 10.0,
      "period_in_ms": 12.50,
      "period_count": 4,
      "big_delta_in_ms": 58.0,
      "maximum_gradient_slew_rate_in_tesla_per_m_per_s": 200,
      "gradient_time_resolution_in_us": 10.0,
      "gradient_magnitudes" :
        (
```

```
          ( 59.0 )
        ),
      "gradient_orientations" :
        (
          200
        ),
      "noise_standard_deviation": 0.0
    }
  }
}
```

Only one PGSE and one OGSE sequence were presented in this JSON, nevertheless, a larger number of sequences can be individually defined, such as the sequences specific to the ICORTEX protocol presented in the section 4.3.2.1, with a total of 15 sequences.

4.2.1.4 Distribution of the parameters

MEDUSA via its JSON, offers distinct ways to implement the value of a parameter during the simulation:

- The value is constant.
- The value follows a distribution function with a specific mean and standard deviation. Usually, a Gamma distribution function is applied in MEDUSA.

The following pseudo-code illustrates the implementation of the axon diameter following a Gamma distribution function, and the volume fraction fixed to a constant value:

```
"fiber_population_01":
{
  "type": "fiber-population",
  "parameters":
  {
    "axon":
    {
      "diameter_distribution":
      {
        "type": "gamma-distribution",  # The parameter value follows a Gamma distribution
        "parameters":
        {
          "mean": 0.7,            #  Mean of the Gamma distribution
          "stddev": 0.07,         #  Standard deviation of the Gamma distribution
          "multiplicator": 1e3,   #  Scaling factor
          "upper_boundary": 5.0   #  Upper threshold of the distribution
        }
      }

[ ... ]

    "volume_fraction": 0.5,  # The parameter value is fixed
    }
  }
}
```

The difference between the two approaches lies in the fact that, unlike constant parameters, the parameters defined by a distribution function have values different for each use of the same JSON input, evolving along a band of definition in line with the mean and standard deviation of the distribution function.

4.2.2 Meta simulation dictionary configuration

The JSON formalism presented until now allows the generation of a single complete MEDUSA simulation. Considering the objective of creating a large database of samples, it is not sufficient. Indeed, repeatedly launching a simulation described by the same three JSONs will result in generating look-alike samples, both in terms of geometry and diffusion. Moreover, we want to avoid manually writing thousands of different JSON files. Thus, it's necessary to distribute the parameters within the dictionaries to offer a broad spectrum of representation. To do this, a "Meta-JSON" configuration file was implemented to meet this requirement. This file combines the three original JSONs and is written slightly differently from the classic JSONs. Each parameter can now be characterized by two specific keywords, which are: "Random" or "Linked". The "Random" keyword specifies that the concerned parameter should be randomly distributed according to a uniform distribution. The "Linked" keyword specifies that the parameter depends on another parameter value, which can be modified according to a specific pattern. The Meta JSON configuration file also embeds a "simulation target count" parameter. This parameter indicates the number of samples to generate during the simulation campaign. Unlike the MEDUSA simulator, the Meta JSON script is written in Python and is based mainly on recursivity. The following pseudo-code presents the overall structure of the Meta-JSON dictionary:

```
"one_fiber_campaign_simulation":  # Name of the simulation campaign, helpful for file/folder
                                  # management when simulation campaign results are written
{
  "target_sample_count": 10,  # Total number of samples synthesized by the simulation campaign
  "geometry":  # Sub-dictionary dedicated to the geometry
  {
    [ ... ]
  },
  "diffusion_process":   # Sub-dictionary dedicated to the diffusion process
  {
    [ ... ]
  },
  "dMRI_sequences":   # Sub-dictionary dedicated to the dMRI sequences
  {
    [ ... ]
  }
}
```

The following pseudo-code shows an example of how keywords "Random" and "Linked" can be implemented within a Meta-JSON, taking as an example the axon diameter parameter:

```
"one_fiber_campaign_simulation":
{
  "target_sample_count": 10,
  "geometry":
  {
    [ ... ]
    "populations":
    {
      "fiber_population_01":
      {
        "type": "fiber-population",
        "parameters":
        {
          "atom_oversampling_ratio": 10.0,
          "axon":
          {
            "diameter_distribution":
```

```
            {
              "type": "gamma-distribution",
              "parameters":
              {
                "mean":
                {
                  "random":   #  Key-word specific to Meta-JSON, the distribution
                              #  of the parameter follows a uniform function
                  {
                    "lower_value": 0.5,  #  Upper threshold of the distribution
                    "upper_value": 3.0   #  Lower threshold of the distribution
                  }
                },
               "stddev":
               {
                 "linked":   #  Key-word specific to Meta_JSON, the value of the parameter
                             # follow the value of another parameter from the JSON
                  {
                    "type": "scaling_function",  #  Specify the type of relation
                                                 #  linking the two parameters, here
                                                 #  it's a scaling function
                    "parameters":
                    {
                      "coefficient": 0.1,  # The scaling ratio
                      "linked_paths":   #  Indicate the position of the
                                       #  linked parameter within the Meta-Json
                       [[
                        "one_fiber_campaign_simulation",  #  Successive embedded
                        "geometry",                       #  dictionary key-word
                        "populations",
                        "fiber_population_01",
                        "parameters",
                        "axon",
                        "diameter_distribution",
                        "parameters",
                        "mean"
                       ]]
                    }
                  }
               },
               "multiplicator": 1e3,
               "upper_boundary": 5.0
             }
          },
      [ ... ]
       }
   }
}
```

The benefit of distributing the mean and standard deviation values from a distribution function is to allow the sampling of different variation ranges corresponding to various areas of brain tissues. For example, we do not expect the same variation of axon diameters in the corpus callosum as in the cortico-spinal tracts, which contain larger motor-related fibers.

In conclusion, the use of dictionaries as inputs to launch MEDUSA simulations is simplifying the clarity of parameter management. The MEDUSA input structure is presented in [figure 4.21]. In the future, this input formalism could be the foundation stone of a MEDUSA web service, where the input parameters of these dictionaries will be modified by users via a GUI interface, thus improving

MEDUSA's user-friendliness.

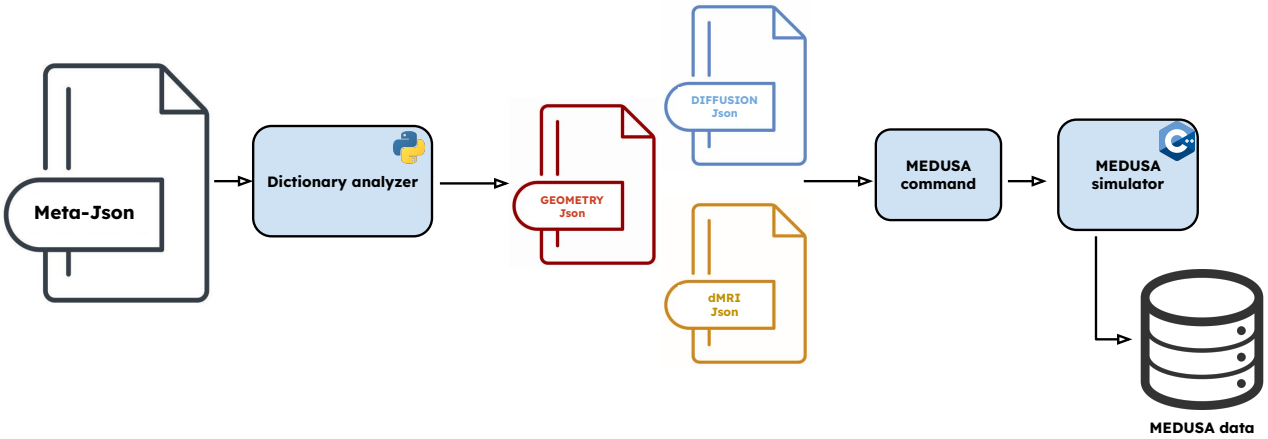


Figure 4.21: *Overview of the MEDUSA input pipeline.*

4.3 Validation

Until now, we have detailed the concepts used to create the MEDUSA framework and described the formalism surrounding the efficient way to generate simulation campaigns with their associated configuration files. But we still have to answer a question: How reliable are MEDUSA simulations regarding ground-truth?

For the simulation of brain samples, the tuning of the generative microstructural parameters of the MEDUSA simulator relies on the in vivo and ex vivo observations made on brain tissue architectures (see chapter 1 1.2.3). A closer investigation of numerous brain matter histology slices would be necessary to draw an accurate distribution table of actual microstructural characteristics of brain matter. Nevertheless, to illustrate the complexity of such a task, we can mention the brain map project from Google, freely available online, including 50,000 cells, all rendered in three dimensions, forming 130 million synaptic connections [Shapson-Coe et al. 2021]. To achieve such a colossal breakthrough, a tiny piece of brain from a 45-year-old woman was cut into slices of 30μm thickness and observed with an electron microscope for an unprecedented data set amounting to 1.4 petabytes. Such work requires colossal human and computational resources. Thus, within the framework of this thesis, we will consider the observations on brain microstructure presented in chapter one to be sufficient. Our confidence in MEDUSA geometry settings is supported by the fact that microstructural parameters are assigned using a Gamma distribution; thus, if the distribution bandwidth is large enough, a large proportion of bio-realistic samples should be synthesized during a large-scale simulation campaign.

As for the diffusion process, it remains fundamental to check that the dMRI signal attenuations resulting from the MEDUSA experiments match the physical ground truth. The simulator attempts to combine realism and computational efficiency using parsimonious cell representation based on atoms, which reduces calculation times, but according to the state of the art, only MEDUSA offers this method. Thus, it is legitimate that the question arises whether the signal resulting from the MEDUSA simulator is coherent and realistic. We also want to quantify the impact of the simulation parameters on the accuracy of the signal in order to identify an acceptable compromise between realism and computational complexity.

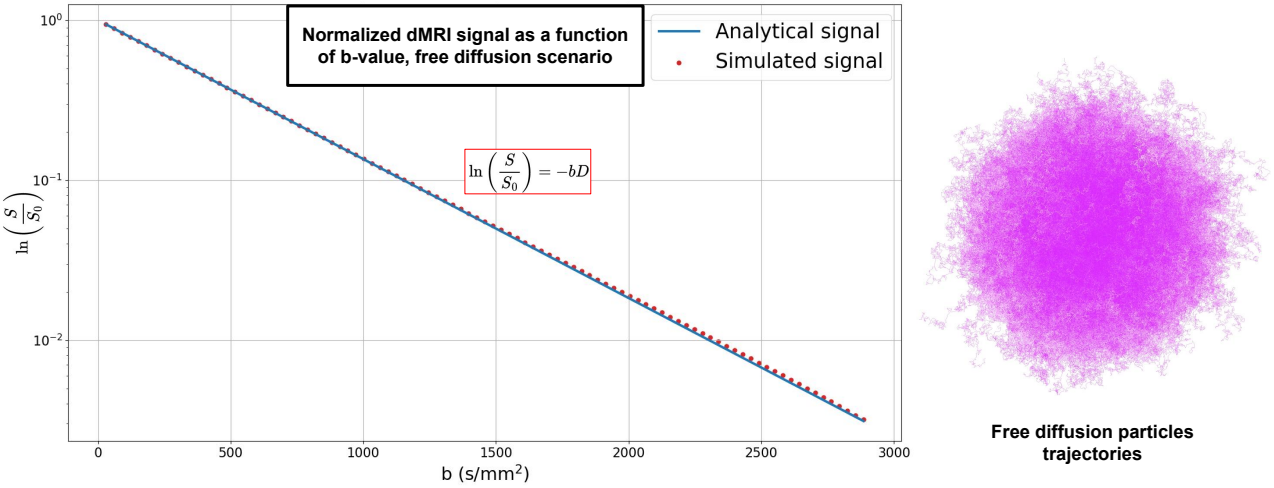


Figure 4.22: *On the left: the plot from the simulated normalized signal concerning b-values matching the analytical computed signal for free diffusion water. On the right: MEDUSA plot of 2% of the* 10^6 *distributed particle trajectories after a diffusion time of 70.2ms and a time step of* $100\mu s$*. Courtesy of Anas Bachiri.*

Consequently, two different validation steps will be presented. The first validation is inspired by the standards set by the previous simulators (see chapter 3 3) and will consist of obtaining the dMRI signal attenuation resulting from a generic spherical geometry for which the analytical diffusion equations have been solved. The second validation is an approach specific to this thesis and relies on diffusion MRI signal attenuation stemming from more complex geometries, thus deprived of any analytical ground-truth.

4.3.1 MEDUSA reliability regarding analytical solutions

Following the recommendations from [Fieremans and Lee 2018], two tests were performed to quantify the reliability of MEDUSA regarding analytical solutions of the diffusion equation: the simulation of the dMRI signal attenuation resulting from the case of a free diffusion of water and, the simulation of dMRI signal attenuations resulting from the case of diffusion in a sphere. These two investigations were performed by my co-worker Anas Bachiri.

In the case of free diffusion, the dMRI signal attenuation is obtained using a set of PGSE sequences whose b-values range from 0 to 3000 $s.mm^{-2}$ and with gradients oriented along a single direction. A Monte-Carlo simulation of the diffusion process was launched for a distribution of 10^6 particles initially located at the same point, for a time step of 100 μs, a diffusion time of 70.2 ms, and a particle bulk diffusivity of $2.10^{-3}\mu m^2.\mu s^{-1}$. The dMRI signal attenuation $\frac{S}{S_0}$ is plotted as a function of the b-values and compared with the analytical signal. As depicted in [figure 4.22], the two plots match exactly, and the straight monotonically decreasing line observed on this log representation assesses the mono-exponential decay of the dMRI signal attenuation.

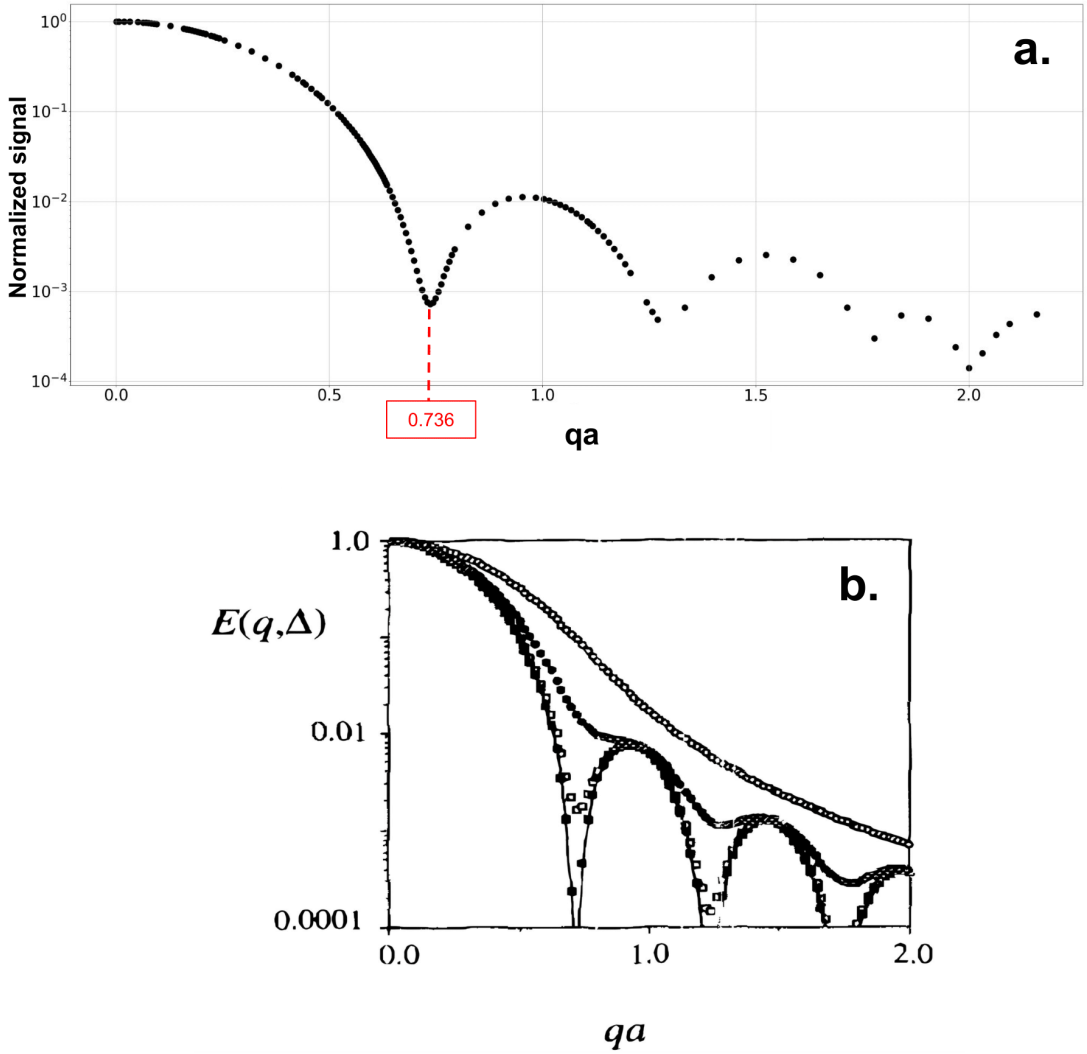


Figure 4.23: *On the top, simulated dMRI normalized signal according to the quantity qa in the case of a sphere, courtesy of Anas Bachiri. On the bottom, the analytical dMRI signal stems from particles trapped in a spherical medium for several values of Δ, adapted from [P. Callaghan 1995].*

For the second experiment, we considered the diffusion within a sphere bounded by an impermeable membrane. Since the analytical form of the dMRI signal attenuation is known from [P. Callaghan 1995], we can compare it with the simulated dMRI normalized signal according to the dimensionless quantity qa, with a being the radius of the sphere and q being the wave vector q norm. In practice, we performed a MEDUSA Monte-Carlo simulation with 10^6 particles within a sphere of radius a = $10\mu m$ during a diffusion time of 500ms and with a time step of $35.5\mu s$. Sequences were all PGSE with gradients oriented toward the same direction, with a Δ value of 498 ms and a δ value of 1.5 ms. The membrane interaction considered is an elastic bouncing reflection (EBR). The quantity qa evolves only through increasing G norm ranging from 0.5 to $3.4e^{-3}T.m^{-1}$. We compared the analytical equation proposed in [P. Callaghan 1995] with the simulated results [figure 4.23] and observed that both curves depict multiple pics, with the first one obtained for qa = 0.736, which corresponds to the expected theoretical value.

4.3.2 MEDUSA validation in the case of complex geometries

Diffusion simulations cannot be validated by comparison with analytic solutions of the diffusion equation in the case of complex geometries such as the one encountered in bio-realistic simulated white

matter with numerous fiber populations crossing each other. We will examine the influence of crucial simulation parameters by benchmarking the signal attenuation stemming from different simulation settings with the signal attenuation obtained from a reference simulation purposely configured with a high level of detail, thus being considered to include a high level of realism regarding the ground truth diffusion phenomenon. Computational times following each simulation setting will also be investigated to reach an optimal tuning of the simulation parameters in order to satisfy both computational requirements and simulation realism.

4.3.2.1 Relative step length and complexity

We first investigate the influence of the time step Δs and step length Δl. They are both related by Einstein's equation:

$$\Delta l = \sqrt{6 \cdot D \cdot \Delta s}, \tag{4.7}$$

where D is the bulk diffusivity, set to $3.10^{-3}\ m^2 s^{-1}$ in the case of water at 36.6°C [Olaf 2002], which represents the average temperature of the human body. The step length value must be adapted to the size of the microstructural elements we want to probe within the simulated geometry, such as small elements require small step lengths [Xing et al. 2013]. From the equation 4.8, we associate Δl with a percentage of the dimension of the smallest simulated element of scene x, and thus define a Relative Step Length ratio (RSL) that can be defined as such:

$$RSL = \frac{x}{\Delta\ l} \tag{4.8}$$

The study of the RSL alone is not sufficient. We can establish a link between the quality of a Monte-Carlo simulation and the capacity of the particles filling the field of view to explore the whole geometrical space during a given diffusion time. Thus, we should consider in addition the number of particles Np as well. The notion of complexity developed in [Matt G. Hall and Daniel C. Alexander 2009] establishes a link between Np and Δs with the the following equation:

$$U = Np.N_i, \tag{4.9}$$

with U the complexity and N_i the number of iterations during the Monte-Carlo process such that:

$$N_i = \frac{TE}{\Delta\ s}, \tag{4.10}$$

with TE, the echo time. Finally we can re-write (4.9) according to (4.10), (4.8) and (4.7) to link the notion of complexity and RSL:

$$U = Np.\frac{6.TE.D}{(RSL.\Delta\ l)^2} \tag{4.11}$$

The previous equation is the union of notions both discussed in [Xing et al. 2013] and [Matt G. Hall and Daniel C. Alexander 2009]. The impact of Np and Δl on a Monte-Carlo simulation is illustrated in [figure 4.24], where we can observe that a good diffusion simulation is achievable only if both parameters are adequately tuned.

The question to answer now is the question of the optimum number of particles and time step value. How can we assess the quality of the results, given that the analytical diffusion equations do not apply directly to our geometries? The higher the number of particles and the smaller the time step,

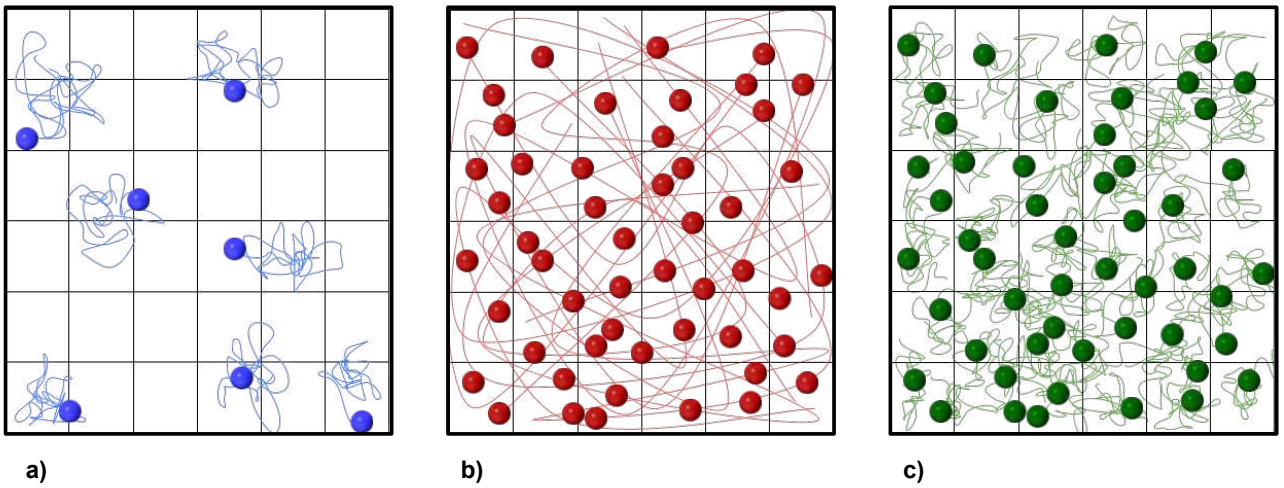


Figure 4.24: *2d illustrations of Monte-Carlo simulations. a) The step length is small enough, but too few particles are distributed, leading to insufficient space coverage. b) The large number of particles allowed to fill the space, but the large step length reduces the chance to probe the finest details of the simulation correctly. c) Both Np and sl are correctly set, leading to an optimal Monte-Carlo simulation.*

the more realistic the simulator is because the simulations mimic the infinitesimal nature of the water particle diffusion phenomenon but at the price of a loss of computational efficiency and a lot of memory consumption. As suggested in the introduction, we need to set up a reference simulation with a high number of particles and a small time step. Following the recommendations from [Xing et al. 2013], we chose an RSL of 3%, a particle count N_p=100000, and an echo time of 130ms. The average size of the elements within simulated geometries has been set around 1μm. Consequently, an RSL of 3% yields a time step of $3.10^{-2}\mu s$ and a number of iterations of $4.3.10^6$. According to [Rafael-Patino et al. 2020], such settings provide an adequate reference simulation, as we can observe on the relative mean absolute error (RMAE) heat map stemming from simulated dMRI signal attenuations compared to analytical solutions in the case of cylindrical geometries [figure 4.25].

The dimension of the smallest element varies according to the simulation as it is randomly distributed. To figure out which population of cells contains the smallest element, the MEDUSA simulator uses the potential offered by Meta-JSON files formalism by linking the value of the step length to the diameter of the smallest distributed sphere atoms. The same approach is used to explore the impact of Np, whose value is linked to RSL, following equation 4.11.

After simulating the reference sample, the focus was put on the simulation of other samples with the time steps tuned via the RSL criterion and the number of particles adjusted via the U complexity parameter. A comparative study was carried out from all the samples and the reference signal was computed for the mean absolute error value from the signals attenuation computed for all the simulated dMRI sequences. The sequences in question are presented in [table 4.1] and correspond to the sequences used for the simulation campaign presented in the next chapter. Following recommendations of [Xing et al. 2013] and [Matt G. Hall and Daniel C. Alexander 2009], we sampled the RSL values at 5%, 8%, 12%, 20%, 35%, 50%, 65%, 90%, and the complexity U to 10^6, 10^7, 10^8 and 10^9 (respectively called complexity 6, complexity 7, complexity 8, complexity 9). To reduce the potential bias of our experiment, we applied the test to 8 different simulated geometries whose characteristics are specified in [table 4.2] and correspond to simulations of three fibers populations.

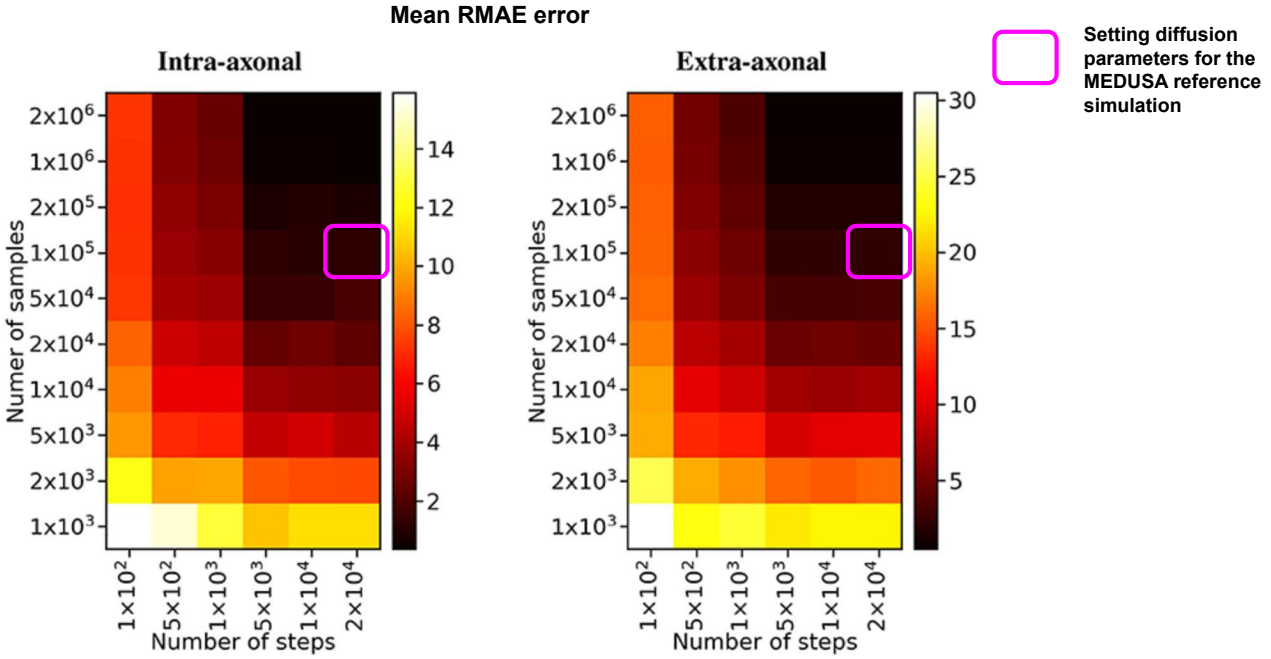


Figure 4.25: *Heat map of the mean RMAE from simulated dMRI signals compared to analytical solutions for all the combinations between the number of steps and the number of samples. The number of samples here means the number of particles. Each cell corresponds to the mean RMAE of 50 repeated simulations. The purple square shows our equivalent configuration for the referential MEDUSA simulation. It should be noted that the actual step iteration count for the referential simulation is 4.3.*10^6*, but the heat map stops at the value 2*10^4*. Adapted from [Rafael-Patino et al. 2020].*

	Big delta (in ms)	Little delta (in ms)	Gradient magnitude	B-value	Gradient orientation count
Sequence 1	31.72	20.24	6.41	30	6
Sequence 2	31.72	20.24	9.07	60	6
Sequence 3	31.72	20.24	11.11	90	6
Sequence 4	31.72	20.24	12.84	120	6
Sequence 5	31.72	20.24	14.36	150	6
Sequence 6	31.72	20.24	15.74	180	6
Sequence 7	31.72	20.24	17.00	210	6
Sequence 8	31.72	20.24	18.18	240	6
Sequence 9	31.72	20.24	19.29	270	6
Sequence 10	31.72	20.24	20.34	300	6
Sequence 11	31.72	20.24	27.58	550	30
Sequence 12	31.72	20.24	31.15	700	30
Sequence 13	31.72	20.24	37.25	1000	30
Sequence 14	31.72	20.24	48.70	1700	60
Sequence 15	31.72	20.24	67.07	3200	60

Table 4.1: DMRI sequences settings for the simulation campaign, adapted from the settings from iCORTEX protocol.

	Field of view (µm³)	Diameter mean* 1 (µm)	Diameter mean* 2 (µm)	Diameter mean* 3 (µm)	Volume fraction 1	Volume fraction 2	Volume fraction 3
Geometry 01	50.6	1.26	2.38	2.53	0.16	0.44	0.25
Geometry 02	34.8	1.74	0.81	1.71	0.22	0.17	0.33
Geometry 03	40.2	2.00	0.90	2.01	0.18	0.17	0.33
Geometry 04	57.6	2.88	1.57	1.90	0.18	0.21	0.39
Geometry 05	48.6	2.29	2.02	2.43	0.17	0.29	0.38
Geometry 06	42.2	2.1	1.92	1.03	0.30	0.25	0.20
Geometry 07	57.8	1.99	1.33	2.89	0.23	0.28	0.30
Geometry 08	47.8	2.39	1.30	1.51	0.18	0.15	0.18

Table 4.2: *Main features of the geometries simulated during the test campaigns. The geometries correspond to the simulation of three fiber populations, accordingly, the diameter and volume fractions 1, 2, and 3 are respectively attributed to fiber populations number 1, 2, and 3.*
** More precisely, it's the mean value from the Gamma distribution function associated with the distribution of axon diameters.*

Two different campaigns were performed, with two membrane interaction settings: first with an equal step length random leap (ERL) and second with an elastic bouncing reflection (EBR). The referential simulation considered in the case of ERL and EBR is different. For ERL, the RSL value is, as mentioned earlier, 3%; for EBR, we take 5%. This was due to RAM limitations encountered during the tests. The results from the campaign are presented in [figure 4.26, 4.27 and 4.29].

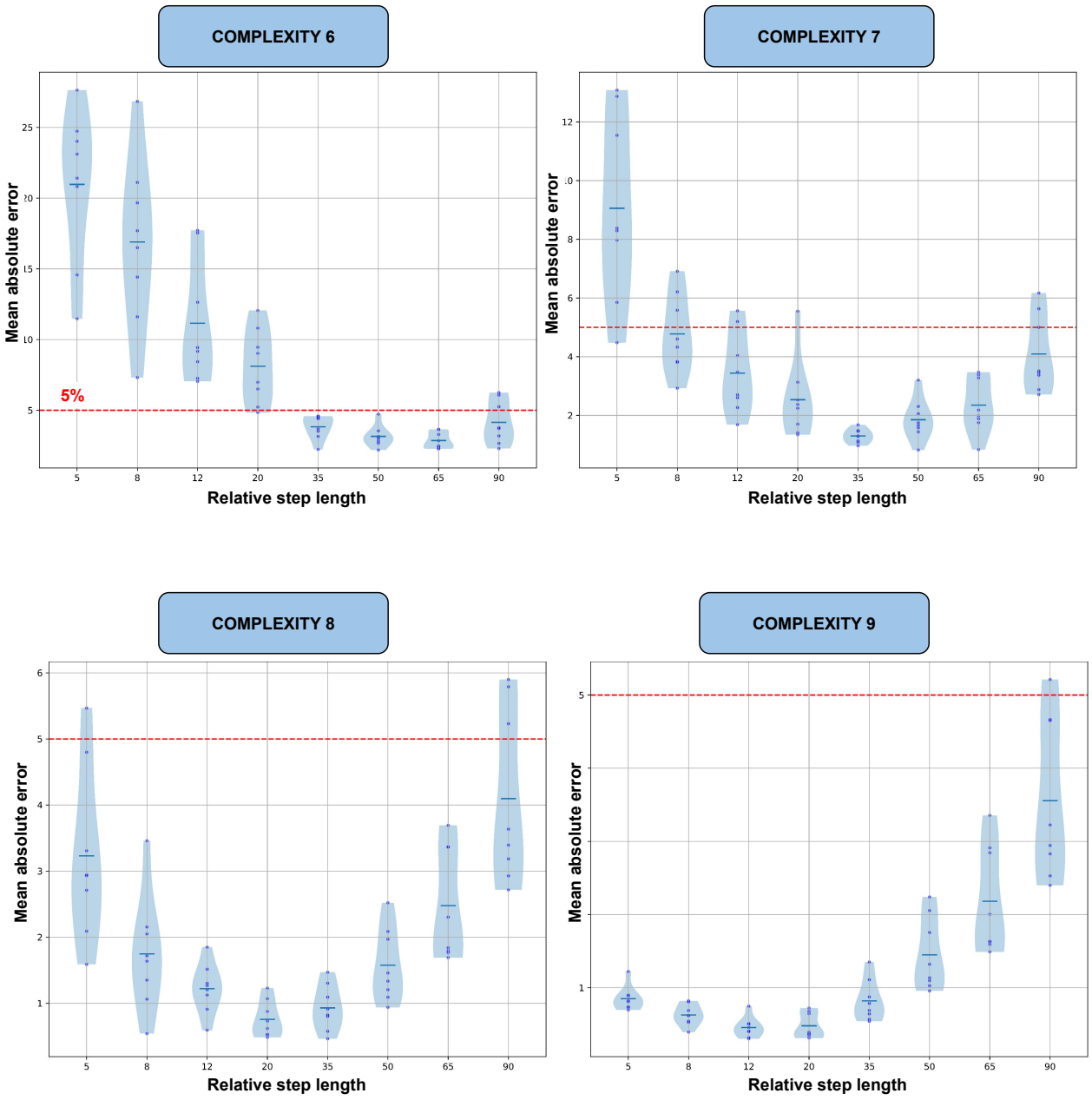


Figure 4.26: *Evolution of the mean absolute error of the diffusion signal from simulations compared with a reference simulation as a function of the relative step length (RSL) for* $U = 10^6$, $U = 10^7$, $U = 10^8$ *and* $U = 10^9$, *case of a random leap membrane interaction.*

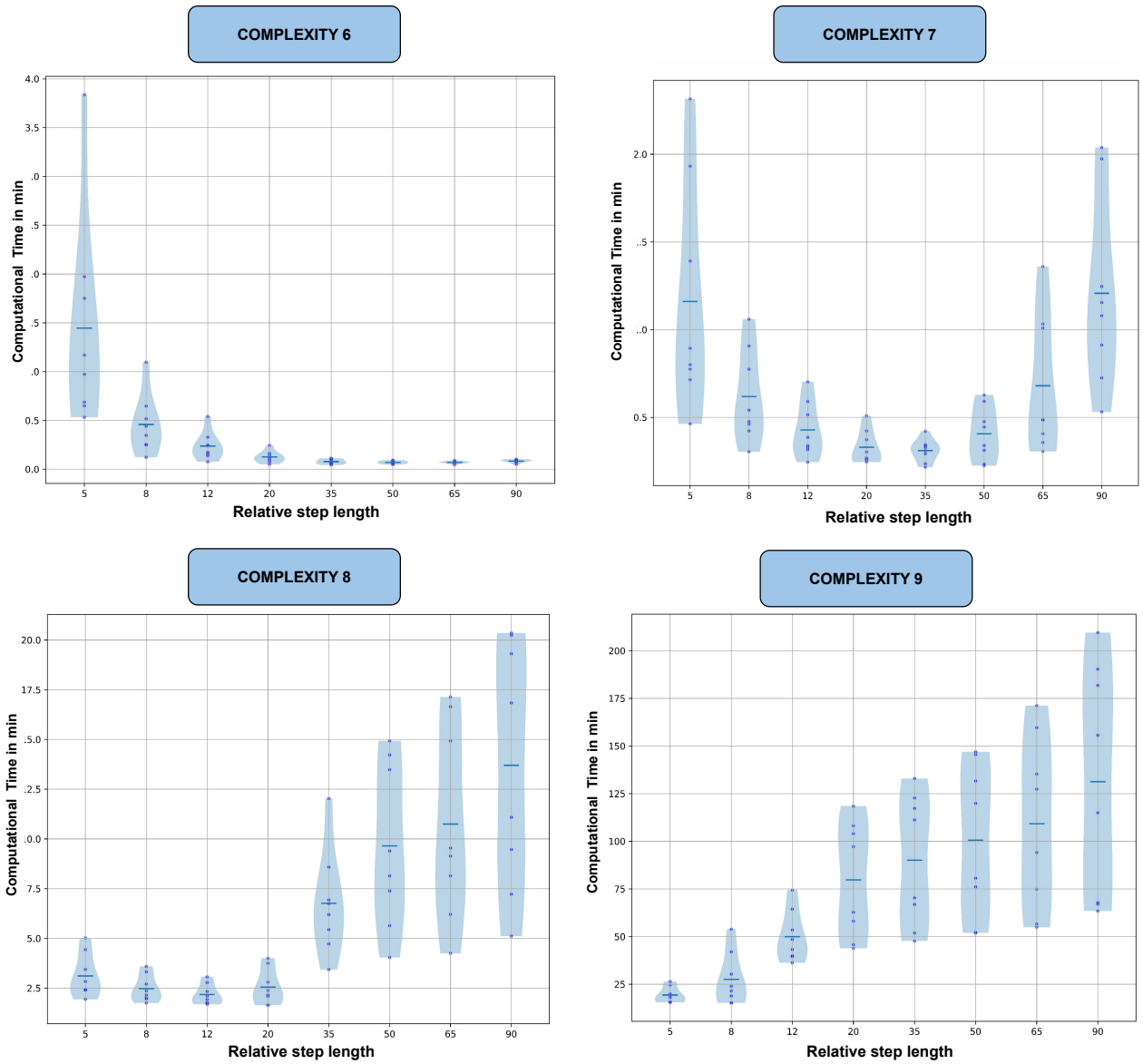


Figure 4.27: *Evolution of the computational elapsed time as a function of the relative step length (RSL) for* $U = 10^6$, $U = 10^7$, $U = 10^8$ *and* $U = 10^9$, *case of random leap membrane interaction.*

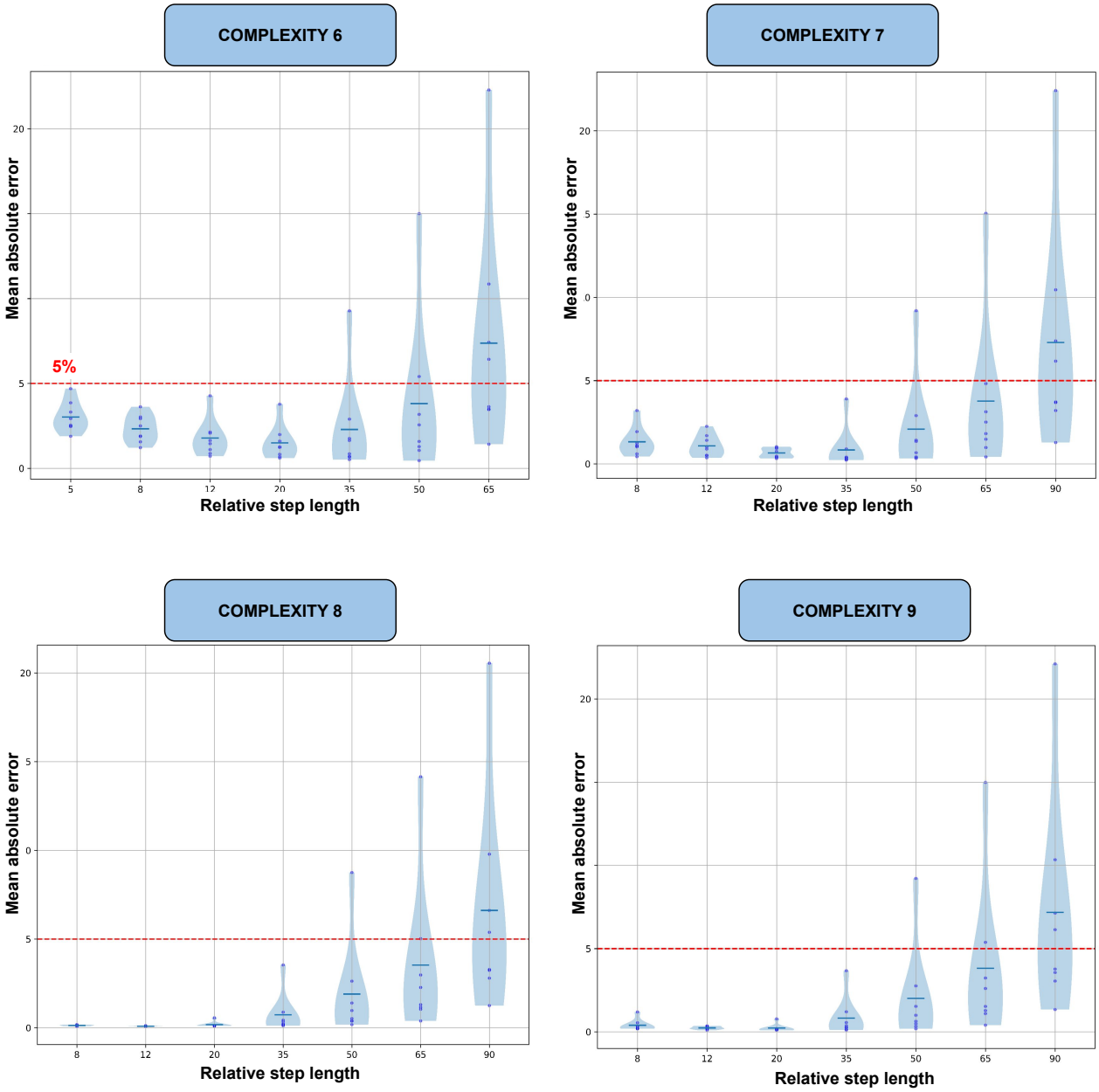


Figure 4.28: *Evolution of the mean absolute error of the diffusion signal from simulations compared with a reference simulation as a function of the relative step length (RSL) for* $U = 10^6$, $U = 10^7$, $U = 10^8$ *and* $U = 10^9$, *case of an elastic bouncing membrane interaction.*

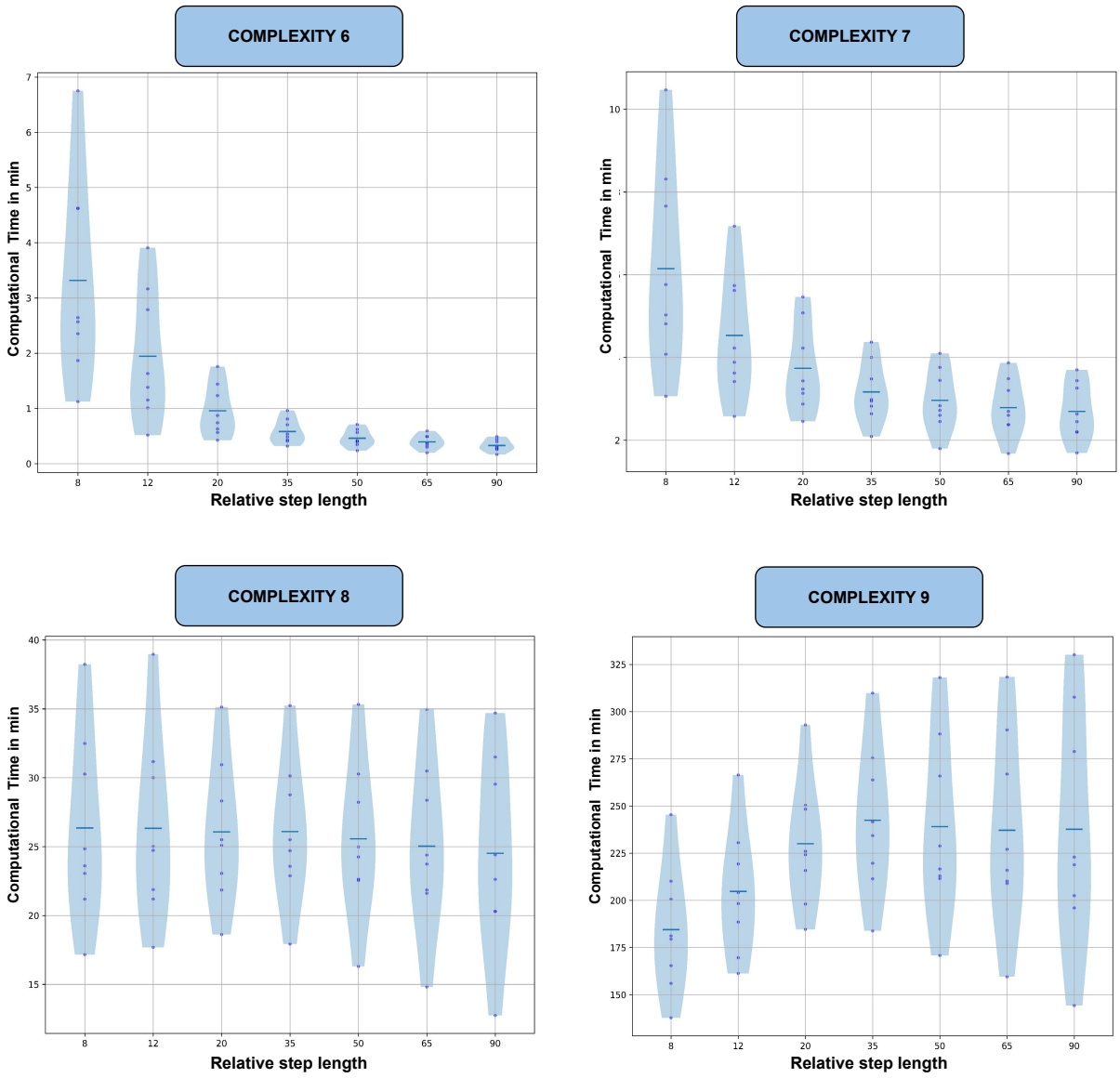


Figure 4.29: *Evolution of the computational elapsed time as a function of the relative step length (RSL) for* $U = 10^6$*,* $U = 10^7$*,* $U = 10^8$ *and* $U = 10^9$*, case of elastic bouncing membrane interaction.*

As expected, higher geometry complexities yield an increase in computational times. Because the complexity is linked with the RSL, the number of particles evolves according to RSL. A smaller value of RSL implies a large number of particles; reciprocally, a larger value of RSL implies a smaller number of particles. The presented results only depict the mean absolute error, but an experiment has also been performed to calculate the mean square error which leads to the same trend.

Concerning the precision requirement, since typical dMRI acquisitions are corrupted by the presence of noise, we set our quality threshold to 5%. In this case, and for the random leap membrane interaction type, the ideal simulation was found for an RSL value of around 15% and a complexity level of 8. The complexity 9 is undesirable as the quality increase is too slim compared to the loss of computational efficacy.

Particular attention must be paid to conclusions concerning campaigns between ERL and EBR particle-to-membrane interaction models because their reference simulations did not follow the same requirements. Thus, the apparent better results from EBR methods must be taken with caution. Nevertheless, globally, we observe that ERL is more sensitive to a lack of particles than EBR, which is coherent with the warnings from the writers of [Xing et al. 2013], mentioning the necessity of a high Np in the case of ERL. EBR seems to be impacted by high RSL values; we can suppose that this is due to the bouncing on membranes that are more frequent in the case of high step lengths with an overall signal impairment.

The number of parameters explored during the simulations implies that the conclusions drawn can have various reasons. For instance, we could have discussed the impact of the field of view on the simulated signal attenuation. Moreover, the number of different geometries proposed here is insufficient to draw accurate general conclusions; thus, a more extensive simulation testing campaign regarding RSL and complexity would be worth being conducted to support the current observations. Nevertheless, the observations relayed here constitute a reasonable basis for an adequate tuning of future simulation campaigns.

4.3.2.2 Impact of the field of view ratio

Both [Xing et al. 2013] and [Matt G. Hall and Daniel C. Alexander 2009] ignored the role of the FOV on signal quality because their tests were performed on simple cylindrical geometries, which do not require large FOV. But in the case of MEDUSA, the question of the FOV is fundamental because the larger it is, the more information is embedded in the diffusion process. However, an infinite-sized FOV isn't necessary to grasp the true nature of the signal, as symmetry patterns are drawn by the cells composing the simulated geometry. Basically, the particles moving towards the geometry are not informing on the precise geometry of each composing element; they instead are averaging the pattern that composes the whole. Knowing this, the FOV from the sample must propose enough repeated patterns to statistically reveal the microstructure characteristics and must be as small as possible to limit computational cost, since a large FOV implies more cells and consequently more particles to explore the space.

The goal here is to find an adequate criterion to tune the FOV satisfying the need for realistic simulation while keeping computational times acceptable. Thus, the same methodology used for the RSL tests was applied here, which means comparing different simulation settings with a reference simulation. Both

ERL and EBR particle-to-membrane interaction models were taken into account. A key difference compared to the previous testing campaigns was that the time step and the number of particles were kept fixed. For ERL, the reference simulation was set with RSL = 20% and U = 10^{10}, whereas testing simulations were set with RSL = 20% and U = 10^9. Since the EBR model reacquires fewer particles [Xing et al. 2013], the reference simulation was set to RSL = 20% and U = 10^9, whereas testing simulations were set to RSL = 20% and U = 10^8.

It was decided to create an intermediate parameter called FOV ratio to tune the FOV value. This parameter links the FOV value to the dimensions of the microstructural elements composing the geometry, such as:

$$FOV = FOV_{ratio} \ * \ \frac{1}{Npop} \left\{ \sum_{Pop_i \in \Omega_{Pop}}^{Npop} D_{SA}(Pop_i) \right\}, \tag{4.12}$$

where Ω_{pop} represent the population ensemble within the simulation, $Npop$ the total number of populations, and $D_{SA}(Pop_i)$ the mean sphere atom diameter distribution for cell population i. The idea was to correlate the FOV with the element composing the geometry, as larger microstructural elements require larger FOV to be correctly observed, whereas small elements require smaller FOVs. The following FOV ratios have been arbitrarily implemented: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27 and 29. The reference FOV ratio was set to 30. The testing simulation campaign was applied to the same eight geometries described in [table 4.2]. The results from this testing simulation campaign are presented in [figure 4.30] based on the study of the mean square error.

If we consider a 5% error acceptable, then a FOV ratio value in the range [10,13] is satisfying for both ERL and EBR. Nevertheless, the number of particles is fixed for every FOV which can be problematic as we can expect that larger samples require a larger amount of particles to grasp the microstructure correctly. Thus, we proposed to launch the same simulation testing campaign but with particle count correlated to the FOV value, such as:

$$Np^{'} = Np * \left\{ \frac{FOV_{ratio}}{10} \right\}^3, \tag{4.13}$$

with $Np^{'}$ is the particle count fitting the dimensions of the FOV, Np is the particle count applied in the previous test and corresponding to an RSL of 20% with a complexity of 9 for ERL, and an RSL of 20% with a complexity of 8 for EBR. Division by a factor of 10 is explained as a FOV ratio of 10 correspond to an average value such that simulations with an FOV ratio above 10 see their particle count increase, and reciprocally, simulations with an FOV ratio under 10 see their particle count decrease.

The grid resolution of the atom LUTs also needs to be optimized. This grid resolution drives the computational time required to identify the neighbors of an atom. For the geometry, a sphere atom is enquiring neighboring sphere atoms, whereas, in the case of diffusion, the particles are enquiring the neighboring sphere atoms. A sphere atom is considered a neighbor if its centroid is located within the same sub-voxel or within one of the 26 adjacent sub-voxels surrounding the voxel of interest within the 3D LUT Cartesian grid. Geometry and diffusion processes have their own LUT and, thus, their own grid resolutions. Knowing this, we can optimize the grid resolution according to the worst-case scenario.

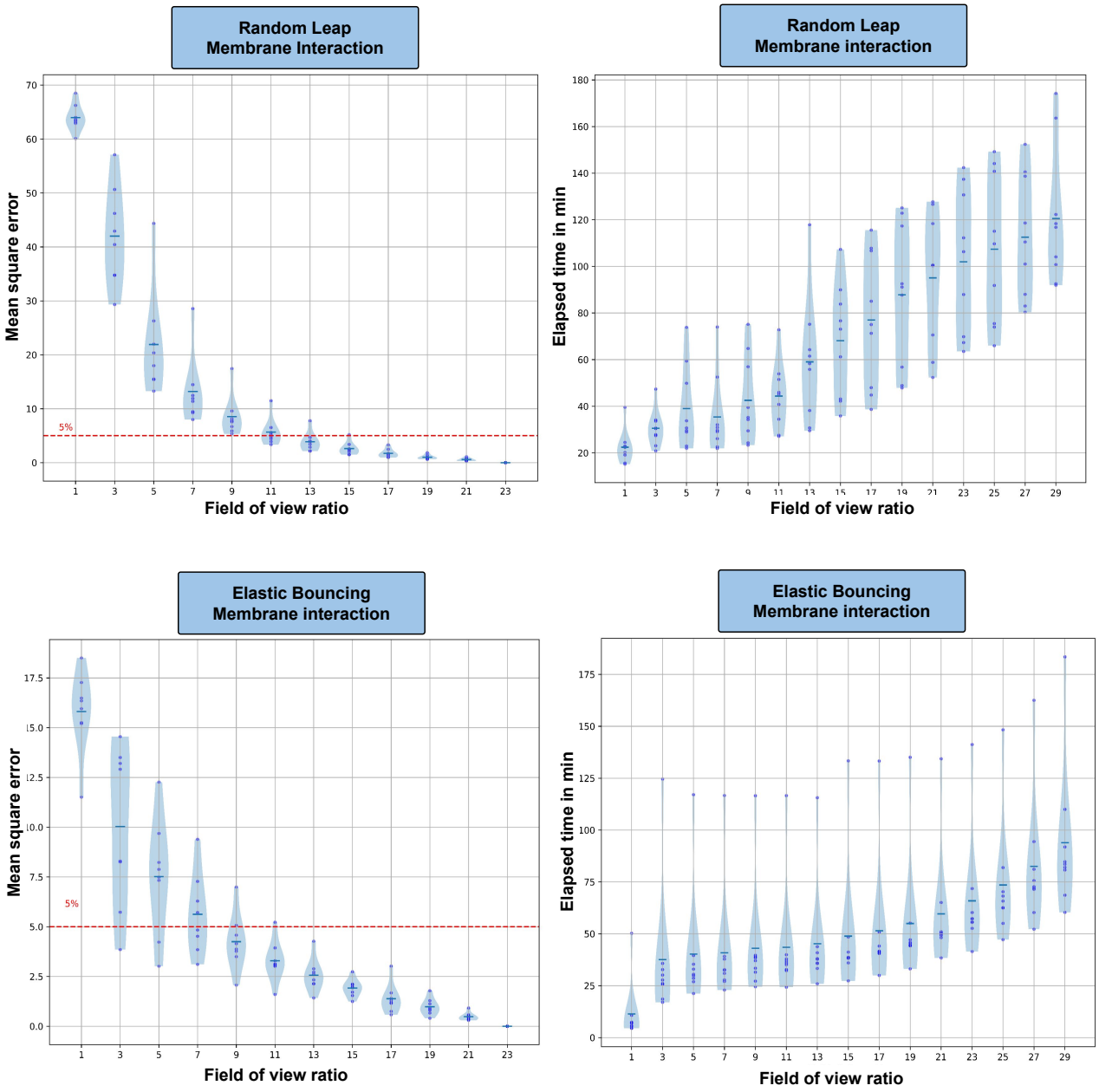


Figure 4.30: *On the left, the evolution of the mean absolute error of the diffusion signal from simulations compared with a reference simulation as a function of the field of view ratio, case of ERL (top) and EBR (bottom). On the right, the evolution of the computational time as a function of the field of view ratio, case of ERL (top) and EBR (bottom).*

For the geometry, this scenario is met with two sphere atoms with the same maximum radius; the first atom belongs to the sub-voxel of interest, and its center is on one edge of the sub-voxel. The second sphere atom is out of the sub-voxel of interest and neighboring sub-voxels. However, because its center is at the edge, its radius can still reach the first sphere atom [figure 4.31]. To avoid this situation, we can conclude that the LUT grid resolution must equal the diameter value of the largest sphere atom of the scene.

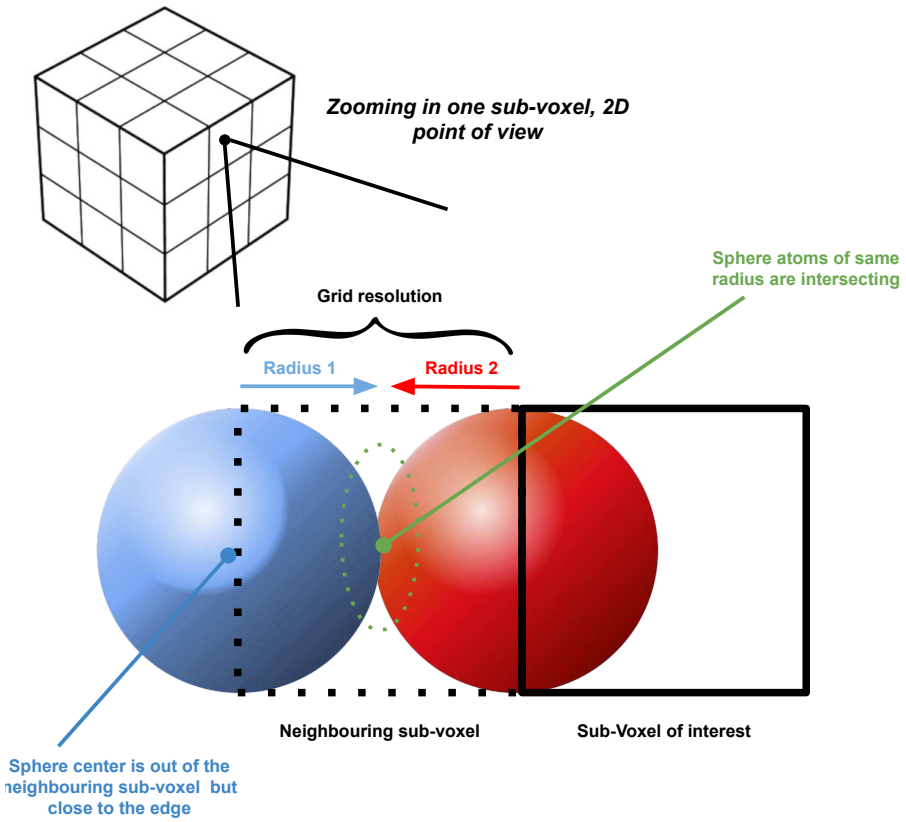


Figure 4.31: *Illustration of the worst-case scenario during a "Remove overlap" procedure. Two sphere atoms see their centers located at the edge of neighboring sub-voxels.*

For diffusion, the worse scenario is met when a particle located at the edge of the sub-voxel of interest is colliding with a sphere atom whose center is on the opposite edge of a neighboring sub-voxel [figure 4.32]. We conclude that optimal LUT grid resolution for diffusion equals the particle step length plus the largest sphere atom radius.

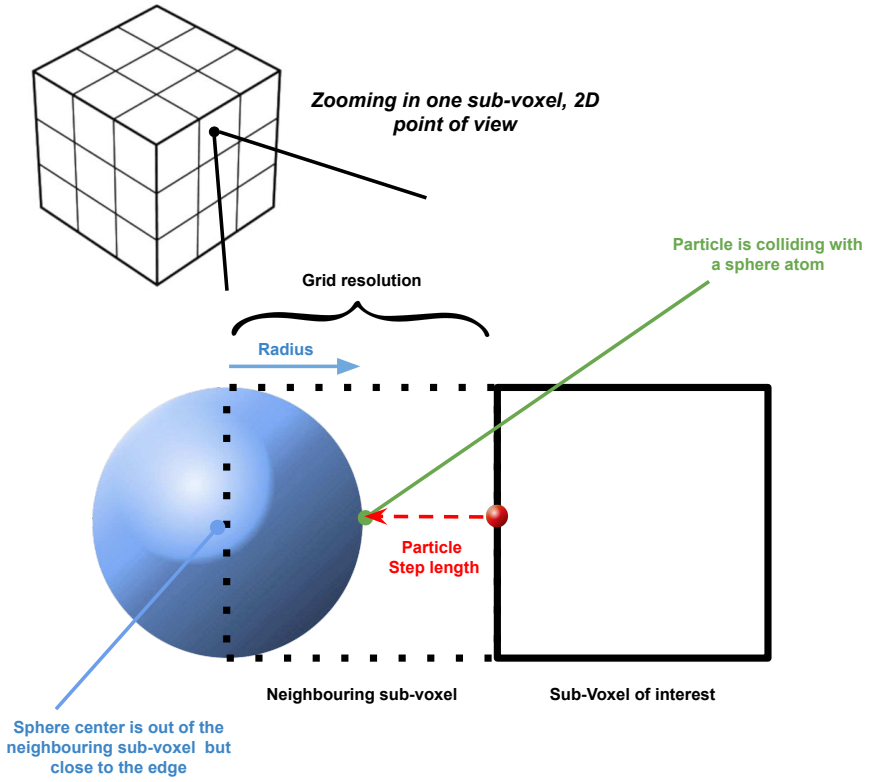


Figure 4.32: *Illustration of the worst case scenario during an iteration stem of the Monte-Carlo procedure. The sphere atom center and the particle are located at two opposite edges of a neighboring sub-voxel.*

After implementing the grid resolution and particle count optimization, we launched again the FOV testing simulation campaign, which results are shown in [figure 4.33].

At first glance, the results look similar to the first FOV testing campaign observed in [figure 4.30]; however, if we look in detail, there is a tiny difference between both results, supporting the idea that the change in the number of particles had no significant effect on the quality of the simulated signals. The computation times are reduced when the FOV ratio is chosen smaller than 10 and increased when the value is chosen larger than 10.

Both FOV testing simulation campaigns support the idea that FOV has an impact on the signal, as we can see that the error decreases when increasing the FOV ratio. The computational time presented corresponds to the simulation's global time, including the geometry and diffusion process. To measure the impact of the FOV ratio on computational times more accurately, we need to observe these two components separately [figure 4.34].

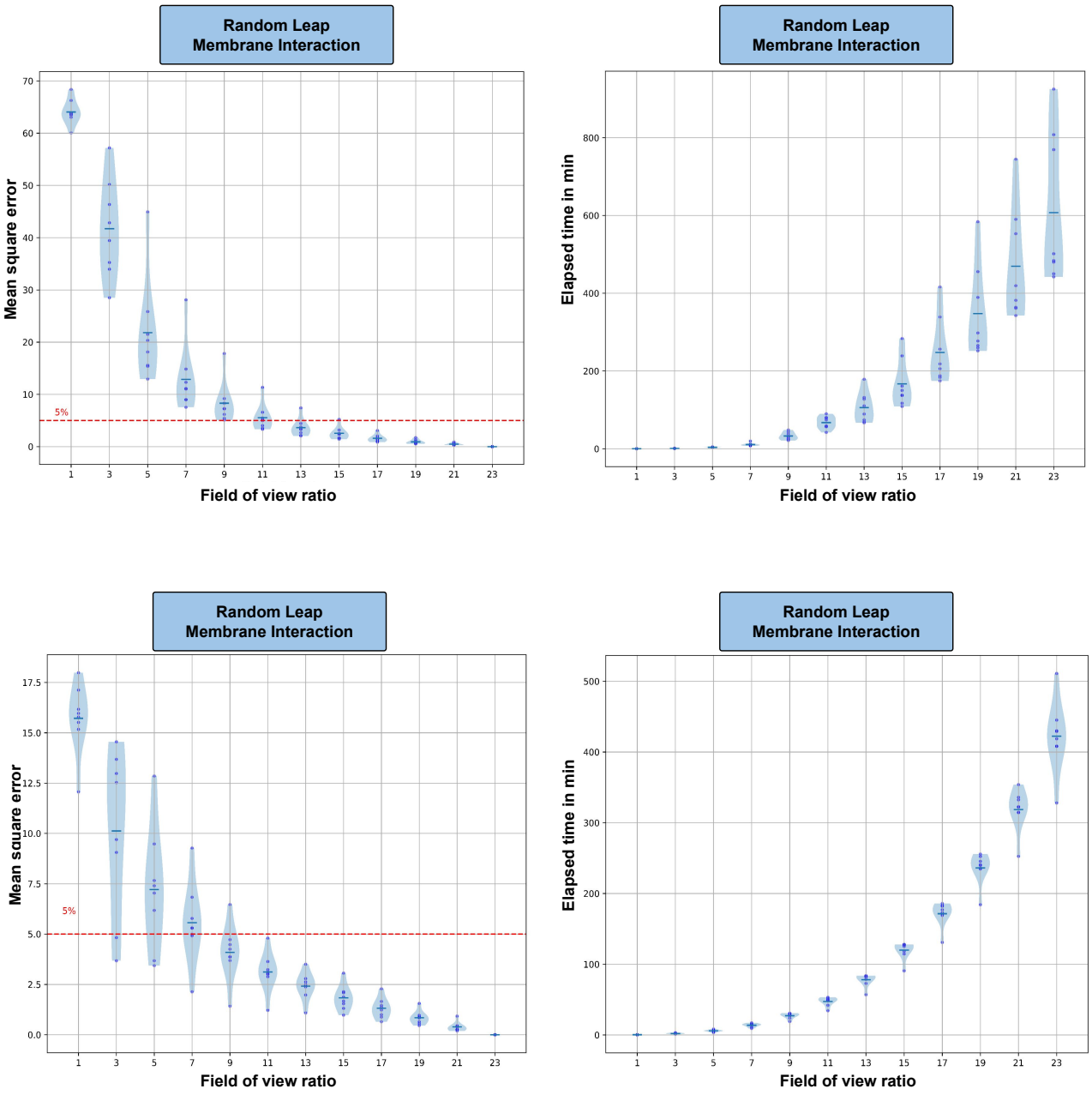


Figure 4.33: *On the left, the Evolution of the mean absolute error of the diffusion signal from simulations compared with a reference simulation as a function of the field of view ratio, case of ERL (top) and EBR (bottom). On the right is the evolution of the computational elapsed time as a function of the field-of-view ratio in the case of EBR and ERL. Implementation of a new way to compute particle count and LUT grid resolution.*

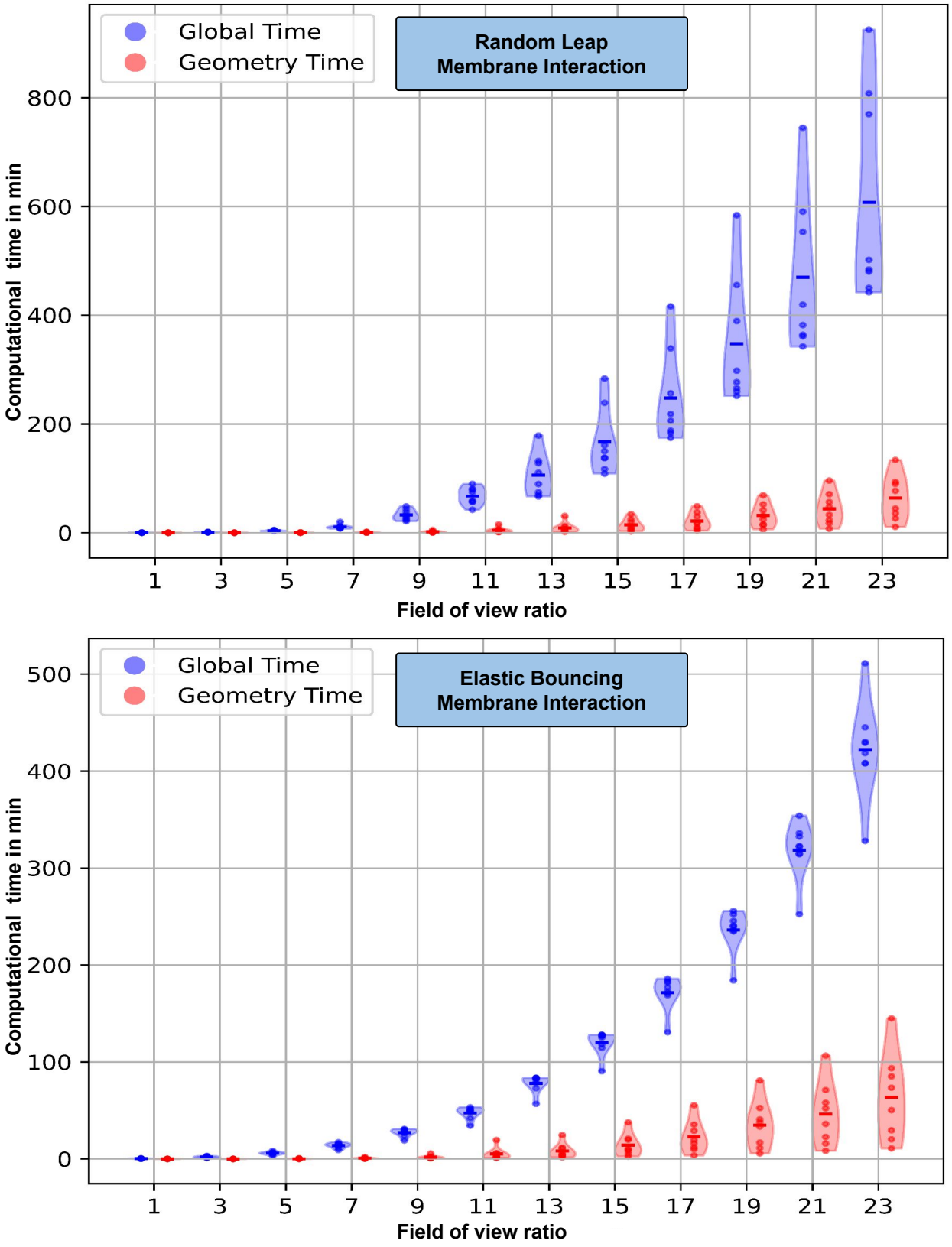


Figure 4.34: *Global computational time compared with geometry computational time, case of ERL and EBR. To obtain the diffusion simulation time, it's necessary to deduct the geometry time from the global time*

We observe that diffusion times are higher than geometry simulation times. As expected, the computational time for geometry and diffusion increases as the FOV increases.

To our knowledge this is the first study of the impact of the FOV on the dMRI signal, as no mention of such tests has been reported before. Nevertheless, The relatively small number of geometries used during the test makes any statistical conclusion tricky. Moreover, it would be worth investigating more in detail the impact of the b-values with respect to the dimensions of the microstructural elements since the sensitivity of dMRI depends on it. The current results are mixing signals stemming from different b-values, which may explain the lack of difference between the first FOV testing simulation campaign and the second improved one.

4.3.2.3 Impact of the overlap removal process

In this last section, we propose to quantify the impact of the remove overlap procedure on the accuracy of the dMRI signal attenuation. We want to quantify its impact on the simulated dMRI signal. To do so, we reproduced a methodology similar to that of previous sections. We consider the signal attenuation stemming from a reference sample of one fiber population obtained after 500 remove overlap iterations, and we compare it with samples generated after 0, 20, 50, 100, 200, 300, and 400 iterations. To take into account the role of the b-value, we considered three simulation scenarios: first, a unique dMRI sequence with a b-value of 200 s/mm^2, then a unique sequence with a b-value of 3200 s/mm^2 and finally, a simulation gathering 15 different sequences with b-values ranging from 30 to 3200 s/mm^2, thus following the imaging protocol established for our future large simulation campaign (see next chapter). The resulting plots are reported in [figure 4.35]:

As expected, the larger the number of iterations for the remove overlap procedure, the lower the dMRI signal attenuation MAE. The geometry arising from a lack of iterations within the remove overlap process remains unplausible, with overlapped cells erasing the microstructural details. Nevertheless, the information of more "gross" microstructural characteristics, such as mean orientation, is preserved. The difference results from the tests with b-value equal to 200 s/mm^2 and 3200 s/mm^2 support this statement: the error from b = 3200 is greater than the one equal to 200 because high b-values are more sensitive to details. The resulting error is an average of both sensitivities for the simulation, gathering simultaneously high and small b-values. An unexpected result can be observed, before decreasing, the signal error is slightly increasing before decreasing. We can explain this by the relatively organized structure of the initial fully overlapped structure, thus providing a coherent signal decay. The first step of the remove overlap procedure is increasing the disorder of the geometry before reaching an equilibrium.

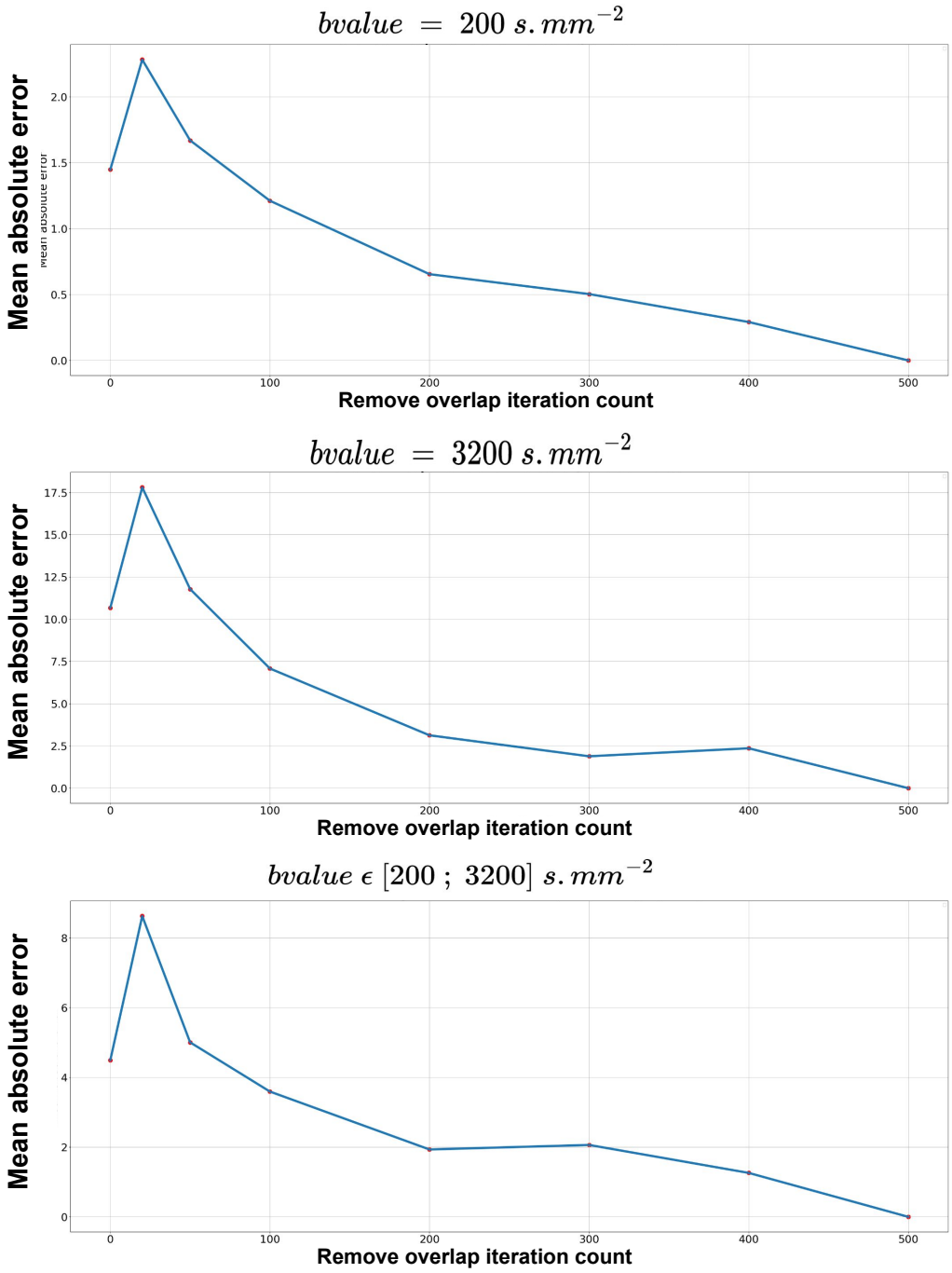


Figure 4.35: *Evolution of the mean absolute error of simulated dMRI signals compared to a reference simulation as a function of the remove overlap iteration count, case of an ERL, b-value = 200s/mm^2, b-value = 200s/mm^2 and b-value in [30, 3200]s/mm^2.*

To conclude this final testing campaign, we can mention that the amount of different samples generated is less numerous than in the previous testing campaigns. It was performed on a local computer, whereas the previous campaign needed the use of an HPC facility. Nevertheless, the observations confirm the sensitivity of the b-value to geometry details and help us quantify the impact of the remove overlap procedure on the signal.

4.3.3 Conclusion

We provided two arguments to support the use of MEDUSA as a reliable simulator. First, the analytical results for the diffusion in a sphere geometry confirm that MEDUSA correctly simulates the diffusion phenomenon in simple geometries. Second, a more empirical approach enabled us to frame the ranges of variation of the various parameters of the geometry and diffusion according to computational times and signal quality. Further investigations are worth conducting in the future, such as questioning the role of geometry details like Ranvier nodes on the dMRI signal or the influence of the voxel boundary limits on the signal. The final aim of this thesis is to propose a decoding model of the white matter brain tissue based on the use of a deep learning algorithm trained with simulated data. Consequently, the ultimate test to validate MEDUSA will also be brought by the validation of the model.

Now that MEDUSA has been presented and validated, we can confidently set the settings of a realistic large simulation campaign. The next chapter will describe this simulation campaign in detail.

Chapter 5

Driving a large MEDUSA simulation campaign

Figure 5.1: *Joliot-Curie supercomputer. Installed at the end of 2017 at the CEA's "Très grand centre de calcul" (TGCC) in Bruyères-le-Châtel (Essonne), with 22 petaflops/s of power, Joliot-Curie serves French and European researchers through the GENCI national and PRACE European research infrastructures. Picture adapted from 2022 Genci's activity report [GENCI 2024].*

Chapter outline

5.1 Computational architecture 137

5.1.1 Kokkos framework 137

5.1.2 TGCC facility 138

5.1.3 Docker virtual containers 140

5.2 Monitoring large scale simulation campaigns 140

5.2.1 Tuning the geometry sampling 140

5.2.2 Tuning the diffusion process simulation 143

5.2.3 Tuning of the dMRI sequences 143

5.3 Simulation campaign results and post-processing 145

5.3.1 Compressed representation of the dMRI signal attenuation 149

5.3.2 From geometric and diffusion data to CSV representations 151

5.3.3 Data augmentation 151

5.4 Conclusion 153

This thesis aims to propose a computational model to decode the white matter microstructure. This model is based on a machine learning algorithm trained on simulated samples mimicking the membrane geometry of WM tissue microstructure as well as their respective dMRI signature. To properly feed the machine learning algorithm, a significant number of samples must be delivered; this objective is facilitated by the computational efficiency of the MEDUSA simulator, allowing realistic sample generation within low computation times. From the beginning, the choice was made to run the simulation campaign on a high-performance computing (HPC) facility as approximately tens of thousands of samples were targeted.

In the following chapter, we will present the various concerns that governed the simulation campaign, starting with the choice of the computational architecture and the description of the "Très Grand Centre de Calcul" (TGCC), the HPC facility on which our simulations have been performed. We will then explain in detail the chosen strategy to sample the microstructural parameter space and launch large simulation campaigns. Finally, the campaign results and the data post-processing pipeline will be presented at the end of this chapter.

5.1 Computational architecture

The computational efficiency of our simulation campaign is crucial to synthesize numerous samples and thus provide enough material to train the decoding model. Consequently, the following sections are dedicated to the different questions guiding the choice of an optimized computational framework for the simulation campaign.

5.1.1 Kokkos framework

As shown previously, a MEDUSA simulation involves different steps, two of them being real computational bottlenecks: the cell overlap removal and the Monte Carlo method to simulate the diffusion process of water. The efficiency of these two steps is directly linked to their optimization. Fortunately, these algorithms can be parallelized and can, therefore, benefit from the use of multicore GPU or CPU architecture. The parallelization programming language depends on the hardware architecture on which the code is launched. For instance, an NVIDIA® GPU (Graphics Processing Unit) architecture involves a CUDA (Compute Unified Device Architecture) coding language [CUDA 2024]. In contrast, a full multicore CPU (central processing unit) architecture can be parallelized using the OpenMP [OpenMP 2024] library. The first release of MEDUSA [K. Ginsburger 2019] was designed for NVIDIA® GPU parallelization and was consequently written in CUDA. The future of HPC seems to be turning toward all GPU, as demonstrated by the opening in 2019 of the "Jean Zay" supercomputer hosted in the IDRISS [IDRISS 2022] HPC facility located in Orsay Essone, whose architecture aims to capitalize on the growing importance of artificial intelligence, which is heavily dependent on GPU architecture. However, many HPC facilities still operate on a CPU basis, and their replacement is not going to happen any time soon, given the prohibitive price of GPU installations. In addition, even if we assume that, in the future, all supercomputers will be equipped with GPU architectures, there is no guarantee that they will come from the same manufacturer since Intel® or AMD® also propose their own GPU. As proof of this concern, the future "Jules Verne' exaflopic supercomputer to be installed at the TGCC in 2025 could be based on AMD GPUs. Consequently, our code must be adaptive if portability to any architecture is desired.

MEDUSA framework should work on any computational architecture for two reasons: first, not to rely on the HPC facility architecture that will operate our code, and second, to be able to run benchmarks to identify the best architectures. However, creating parallelization code specific to each type of architecture is tedious work. To remedy this problem, the MEDUSA framework was highly parallelized using the Kokkos toolkit [Trott et al. 2022], a programming model for writing parallelized applications, portable in terms of performance, open to all HPC platforms. Kokko's use in MEDUSA required a redesign of the parallelization procedures, which resulted in a novel MEDUSA release being compatible with all types of computational architectures.

In terms of performance, Kokkos and CUDA programming models were used to implement Monte Carlo particle transport in an application that models simple 1D photon attenuation on NVIDIA® GPUs [Bossler and Valdez 2018]. The benchmark was performed on three NVIDIA® GPU architectures: Tesla K40, Tesla K80, and Tesla P100; on average, Kokkos was up to 11% slower than the CUDA version for the K40 and K80 but only 5% slower for the P100, which reinforced our choice to integrate Kokkos into MEDUSA as this loss in efficiency is deemed satisfactory given the flexibility offered by Kokkos.

PARTITION	STATUS	TOTAL	DOWN	USED	FREE	TOTAL	DOWN	USED	FREE	MpC	CpN	SpN	CpS	TpC	GpN	GPU Type
rome	up	292224	256	248837	43131	2283	2	1958	323	1781	128	8	16	2	0	
skylake	up	79392	0	51111	28281	1654	0	1079	575	3687	48	2	24	1	0	
a64fx	up	3744	0	0	3744	78	0	0	78	666	48	1	48	1	0	
v100	up	1200	0	1132	68	30	0	30	0	4375	40	4	10	2	4	nvidia
v100l	up	1080	0	1080	0	30	0	30	0	9861	36	2	18	2	1	nvidia
v100l-os	up	1080	0	1080	0	30	0	30	0	9861	36	2	18	2	1	nvidia
hybrid	up	960	0	96	864	20	0	2	18	3750	48	2	24	1	1	nvidia
xlarge	up	560	0	352	208	5	0	5	0	26803	112	4	28	1	1	nvidia
v100xl	up	144	0	144	0	2	0	2	0	40277	72	4	18	2	1	nvidia

Figure 5.2: *TGCC's partitions overall characterization, with MpC the amount of memory per core, CpN the number of cores per node, SpN the number of sockets per node, CpS the number of cores per socket, TpC the number of threads per CPU core and GpN the number of threads per GPU.*

5.1.2 TGCC facility

HPC relies on very high-powered supercomputers for its development and requires exceptional infrastructures to host and operate them. The "Très Grand Centre de Calcul" (TGCC, DAM Ile de France, CEA), located at Bruyères-le-Châtel and conceived by the CEA in 2006, is part of a national vision to host and operate world-class power and visibility supercomputers in France. Currently, the TGCC welcomes the "Joliot Curie" supercomputer funded by "GENCI" ("Grand Equipement National de Calcul Intensif") [GENCI 2024], which is France's contribution to the European PRACE infrastructure (Partnership for Advanced Computing in Europe) [PRACE 2024]. The TGCC is also intended to be a forum for scientific exchange and the promotion of intensive computing.

We regularly obtained resources from the GENCI through the DARI project call, the last one operating on the Joliot-Curie TGCC supercomputer being accepted for the 2023/2024 period, with 5 million core hours allocated to the Ginkgo team. TGCC resources are separated into different partitions, each with its specific computational architecture, as shown in [figure 5.2]. As previous campaigns were performed on GPU [K. Ginsburger 2019], we decided to ask for CPU allocation this time to prove MEDUSA's portability in both architectures. Accordingly, the resources from the "ROME" partition were allocated to the project, with 2283 nodes equipped with 2 AMD CPUs (AVX2, 64 cores per CPU) and 256GB RAM/node, yielding a total of 292224 cores.

To run a job on the partition of a supercomputer, a request is submitted using a configuration file including all the information regarding the computational resources to allocate and the code to launch. Notably, the total computation time, the number of nodes, and the number of cores per node can be tuned. The MEDUSA simulation strategy relies on a two-level fold parallelization approach: N_{node} parallel nodes iteratively execute N_{sample} MEDUSA simulations where the code is parallelized within each node over the N_{core} allocated cores. To reach a total simulated sample count $N_{TotalSampleCount}$, the user must adequately tune the number of nodes N_{node} and the number of samples per node $N_{SamplesPerNode}$ such as $N_{TotalSampleCount} = N_{node}.N_{SamplesPerNode}$. The simulation campaign overall scheme is presented in [figure 5.3].

Scheduler - To launch a job on the TGCC, the demand is examined by an internal scheduler that checks the requested resources and compares them to the requests on hold from other users and the available resources on the current partition. Depending on the resources requested, the task is positioned in a waiting queue, ranging from instant execution to several days of waiting. Other criteria help the

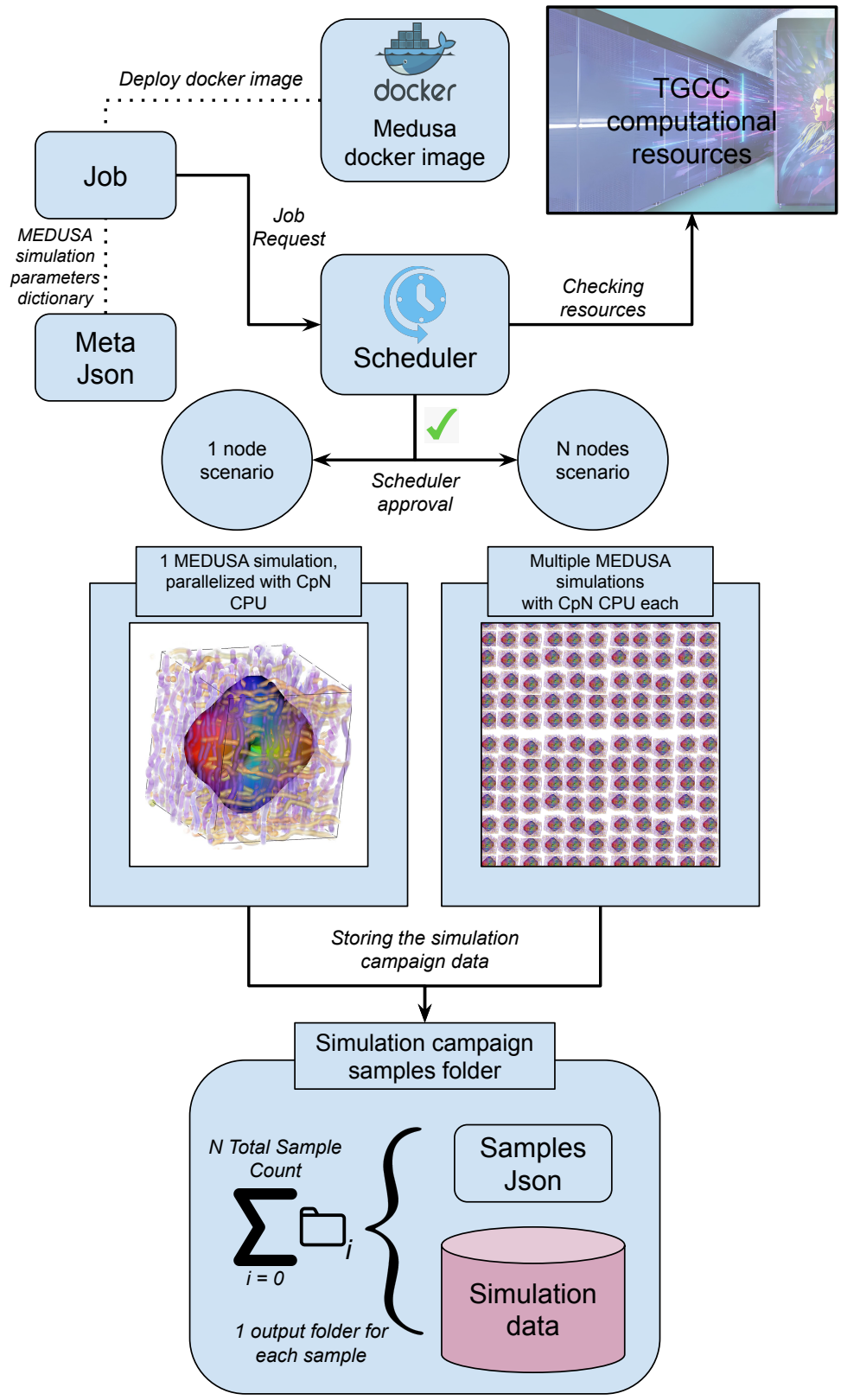


Figure 5.3: *Overall simulation campaign diagram: the job request instigates the deployment of the MEDUSA image from the docker container via the TGCC Peacock interface. Details of the simulation campaign are provided via a Meta-JSON. Depending on the resources available, the request is validated by the scheduler. The scenario of one and N MEDUSA sample simulations are presented. The simulation sample data are regrouped in individual folders and gather signal decay files for each gradient sequence and the three JSON files (geometry, diffusion, dMRI sequence) allowing the sample synthesis.*

scheduler in its choice, such as the number of hours consumed by the project or the usage frequency, promoting a homogenous use of the computational hours, thus facilitating the flow of requests on the partition.

Empirically, we can observe that one of the main criteria that limits the validation of a job by the scheduler is the total computational requested time, as a request exceeding 24 hours has little chance of being accepted. In the case of an application designed as a single sequential block computing for more than 24 hours, this implies the need for intermediate backups to split the procedure into several jobs. Yet, in the case of MEDUSA, each node is simulating $N_{SamplesPerNode}$ samples independently from other nodes. Thus, no backup is required between job requests. However, in the case of an unfinished procedure within a node, the time limit can result in the cancelation of the undergoing simulation.

5.1.3 Docker virtual containers

Although it is possible to locally compile a framework at the TGCC, the use of containers is highly recommended to avoid compatibility and dependency issues. DOCKER containers [DOCKER 2024] behave as a virtual box, including every code, library, and dependency required for MEDUSA. For security reasons, as the original Docker container tool needs access to root access, the containers are deployed within the TGGC through an intermediate containerization tool called "Peacock" [PEACOCK 2024].

5.2 Monitoring large scale simulation campaigns

The following section will present the approach developed in the frame of the thesis to monitor large-scale simulation campaigns, including the tuning geometry, diffusion process, and dMRI signal attenuations.

5.2.1 Tuning the geometry sampling

In the frame of our study, the simulated cells aim to depict the microstructure of WM only. WM microstructure is mostly composed of axonal bundles representing approximatively 60% of the WM volume fraction [Mottershead et al. 2003; Perge et al. 2009; Stikov et al. 2015]. This thesis proposes a proof of concept for implementing a computational model of WM. Consequently, the contributions of glial cells and micro-capillaries on the dMRI signal were neglected at first glance, and the geometries simulated during the campaign only depict axonal fibers.

Simulating a large number of white matter samples involves a relevant sampling of the microstructural parameter space. Currently, the MEDUSA framework relies on the definition of a JSON geometrical configuration file to generate a single sample. The structure, of this configuration file, has been presented in the previous chapter and relies on the tuning of microstructural parameters either with a constant value or with a specific distribution from which the corresponding parameter can be randomly drawn

When moving from the simulation of a single WM sample to a large number of WM samples, one has to create as many configuration files as the number of samples, each time with different microstructural

parameters. To achieve this more automatically, a second level of sampling has to be considered which role is, at a higher level, to sample the constant parameters and the distribution parameters to generate various WM samples with respect to this second-level sampling strategy. In addition, the tuning of certain parameters can depend on the tuning of other parameters. For instance, the global field of view is typically set up depending on the distribution of axon diameters to generate samples including enough axons to statistically provide robust dMRI simulations of the diffusion process. This second-level sampling strategy is implemented with a Meta-JSON configuration file.

The geometries simulated were separated into three categories, respectively representing one, two, and three fiber populations, which is sufficient to cover most voxel configurations at millimeter resolution, each characterized by a specific Meta-JSON dictionary. The parameters describing the geometry will be part of the data set feeding the decoding model of the WM microstructure. As such, the distribution range of the geometry parameters must be wide enough to ensure a good sampling of simulated geometries. However, the more distributed the parameters and the wider the distribution range, the more simulated samples are required to fill the parameter vectorial space accurately; thus, as a first attempt and to limit the number of simulations needed during this campaign, only six generative parameters were distributed as follows:

- The mean axonal diameter of the Gamma distribution (AD),
- The mean global angular dispersion of the Gamma distribution (GAD),
- The volume fraction (VF),
- The three coordinates from the vector of the mean orientation of the fiber population.

As a result, the size of the vector combining the geometry settings changes according to the number of fiber populations within the sample as every population within a sample sees their AD, GAD, VF, and mean orientation distributed. Accordingly, samples with one, two, or three fiber populations have respective microstructural parameter vectors of sizes 6, 12, and 18. An additional parameter corresponding to the sample's population size is added to the vector. [Table 5.1] summarizing the tuning of geometrical parameters for the target simulation campaign.

The volume fraction of each fiber population is uniformly distributed in accordance with the volume fractions of the other populations so as not to exceed a total volume fraction of 1.0. In practice, this involves subtracting the volume fraction previously given for a population from the desired maximum volume fraction. We decided to threshold the minimum volume fraction of each population at 0.15 so that their dMRI footprint would not be confused with the one from other populations.
The mean orientation implementation differs according to the number of fiber populations simulated within the sample:

- If only one fiber population is simulated, the orientation vector is arbitrarily fixed to (0,0,1). This choice comes from the fact that the acquired signal resulting from the dMRI sequence can be rotated to generate samples for all the other mean fiber orientations.
- If two fiber populations are simulated, one population sees its mean orientation fixed to (0,0,1), whereas the other population's mean orientation results from a rotation in a 2D plan following an angle θ randomly distributed according to a continuous and uniform distribution function. All rotated variants of each simulated WM sample can also be easily obtained by applying rotation to the generated dMRI signal.

Microstructural feature		Sampling strategy		Range
Field of view		Field of view ratio equal 10		[5 ; 30] in µm³
Oversampling ratio		Fixed		20
Axon diameter		Gamma distribution	Uniform distribution of the mean	[0.5 ; 3.0] in µm
			Uniform distribution of the standard deviation	[0.05 ; 0.3] in µm
Global Angular Distribution		Uniform distribution		[5° ; 15°]
Mean orientation	One fiber population	Fixed		Population 1 ->(x1, y1, z1) with x1 = 0, y1 = 0, z1 = 1
	Two fiber population	First population : fixed Second population : rotation along one axis		Population 1 ->(x1, y1, z1) with x1 = 0, y1 = 0, z1 = 1 Population 2 ->(x2, y2, z2) with x2 = 0, y2 = -sin(theta), z2 = cos(theta) with theta uniformly distributed in [10°; 80°] range
	Three fiber population	First population : fixed Second population : rotation along one axis Third population : rotation along two axis		Population 1 ->(x1, y1, z1) with x1 = 0, y1 = 0, z1 = 1 Population 2 ->(x2, y2, z2) with x2 = 0, y2 = -sin(theta), z2 = cos(theta) with theta uniformly distributed in [10°; 80°] range Population 3 ->(x3, y3, z3) with x3 = cos(theta)*sin(phi), y3 = -sin(theta), z3 = cos(theta)*cos(phi) with theta and phi uniformly distributed in [10°; 80°] range
Tortuosity	Has tortuosity	Fixed		on
	Magnitude	Fixed		0.02 in µm
	Angular dispersion	Fixed		2.5°
	Wave length	Fixed		25 (spheres atoms concerned)
Myelin sheath	Has myelin sheath	Fixed		on
	G-ratio	Gamma distribution	Mean	0.7 (dimensionless)
			Standard deviation	0.07 (dimensionless)
Ranvier nodes	Has Ranvier Nodes	Fixed		on
	Internodal length	Gamma distribution	Mean	2.5 in µm
			Standard deviation	0.5 in µm
Volume fraction	One fiber population	Uniform distribution		Population1 : V1 = [0.5 ; 0.9]
	Two fiber population	Uniform distribution		Population1 : V1 = [0.15 ; 0.45] Population2 : V2 = [0.15 ; x] such as x = 0.9 - V1
	Three fiber population	Uniform distribution		Population1 : V1 = [0.15 ; 0.45] Population2 : V2 = [0.15 ; x] such as x = 0.6 - V1 Population3 : V3 = [0.15 ; y] such as y = 0.9 - V1 - V2
Beading	Has Beading	Fixed		off

Table 5.1: Tuning of the microstructural parameters as provided by a specific Meta-JSON configuration file used to perform large-scale simulation aiming at the simulation of a dictionary of plausible WM samples.

- If three fiber populations are simulated, the first two follow the same pattern as presented above while the third fiber population sees its mean orientation resulting from the random rotation in the 3D sphere following an angle θ and ϕ randomly distributed according to a continuous and uniform distribution function.

[Figure 5.4] depict examples of WM samples containing 1, 2, and 3 simulated fiber populations with different mean orientations.

5.2.2 Tuning the diffusion process simulation

The settings regarding the simulation of the diffusion process were made according to the results stemming from validation tests (see chapter 4 4.3.2.1). Consequently, the complexity of the Monte Carlo simulation U = $N_{iteration} * N_{particle}$ was chosen to be equal to 10^9 with $N_{particle}$ = 300.000 and RSL = 10%. The particle-to-membrane kept for the campaign was random leap type (ERL) with a bulk diffusivity of 3.10^{-3} corresponding to a body at 37 degrees. According to the validation study from the previous chapter, those settings assure a realistic diffusion simulation for an expected average simulation time of 30 minutes per sample.

5.2.3 Tuning of the dMRI sequences

The choice of dMRI sequences during the campaign brings the question of the characteristics of the dMRI data feeding the machine learning algorithm as different choices of b-values imply different sequence ranges on which the algorithm is trained to decode the microstructure. Several dMRI sequence configurations might result in good algorithm training; nevertheless, after training with synthetic data, the model must be validated according to dMRI data stemming from hardware MRI and acquisition protocols. Thus, we chose to simulate dMRI sequences mimicking the settings of an acquisition protocol designed by the Ginkgo team and currently running in the lab called "iCORTEX".

iCORTEX imaging protocol - Within the framework of this project entitled "In vivo mapping of the CORTex cytoarchitecture, myeloarchitecture of brain functional networks using EXtreme scale analytics and EXtreme fields"(iCORTEX), the ambition is to characterize the cyto- and myelo-architecture of brain functional networks to understand the organization of the cortical tissue at a scale approaching the mesoscopic scale with high and ultra-high magnetic fields using different MRI modalities such as functional, diffusion and quantitative MRI. The iCORTEX project relies on the massive acquisition of gradient echo, T1, T2, and T2* as well as dMRI images with a dense q-space sampling on a cohort of about a hundred healthy subjects aged 20 to 80 years old who will all undergo about 11 imaging sessions including 6 sessions on a 3T Prisma MRI scanner and 5 sessions on a 7T MRI scanner. Two 3T MRI sessions are dedicated to diffusion-weighted imaging relying on PGSE and trapezoidal OGSE dMRI sequences. The first set of imaging data on the 20 young subjects aged 20 to 35 is expected by the end of 2024. Bosco Taddei operates the iCORTEX project under the supervision of Cyril Poupon (Ginkgo/Gaia team) and Christophe Pallier (Unicog team).

The iCORTEX dMRI acquisitions, therefore, constitute a valuable framework to evaluate our computational model. Therefore, the dMRI sequences simulated during the campaign mimic the sequences composing the dMRI iCORTEX imaging protocol. [Table 5.2] below describes the PGSE sequence schemes part of the iCROTEX protocol and simulated during the campaign.

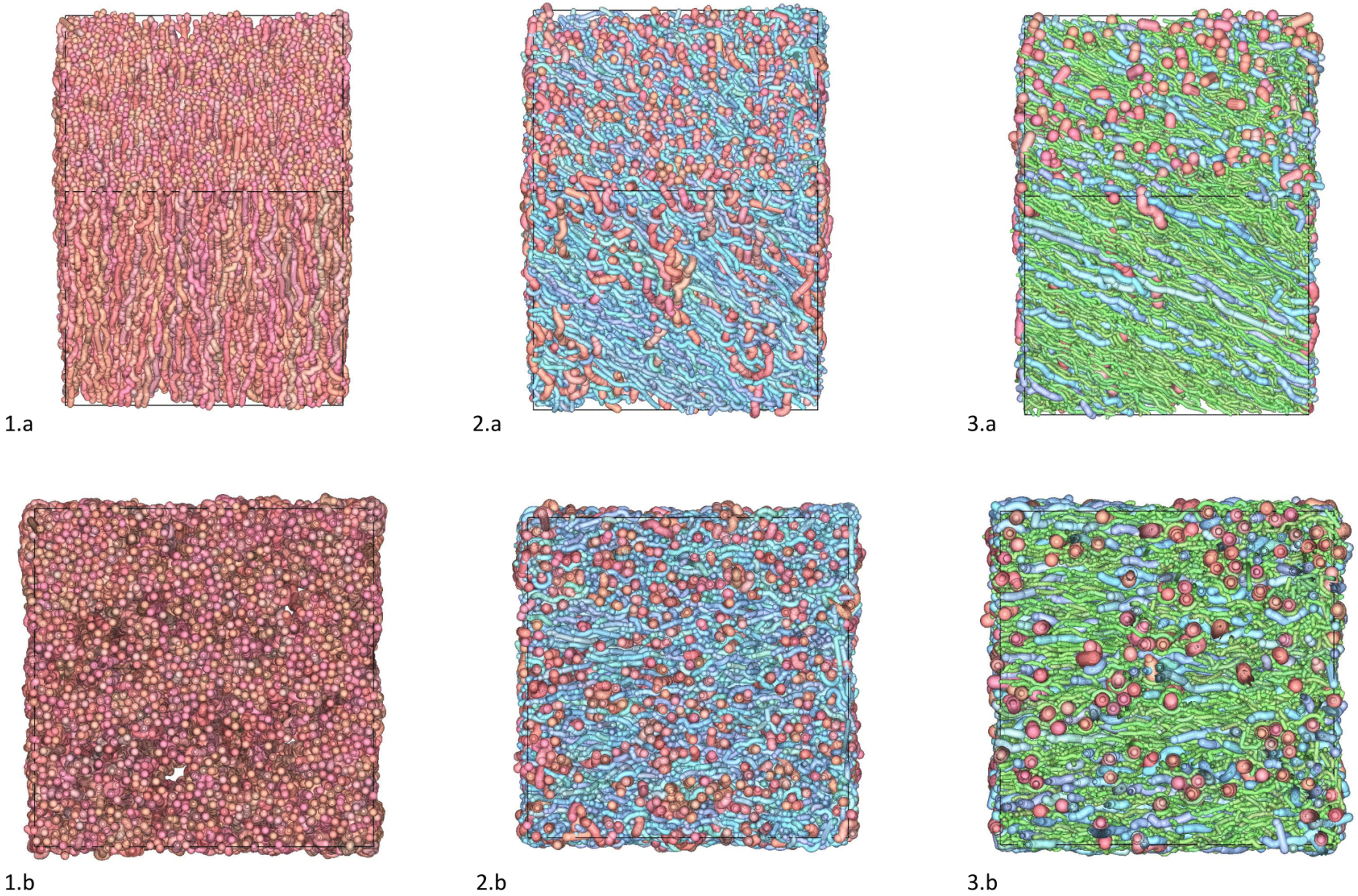


Figure 5.4: *3D rendering of white matter samples mimicking one single fiber population (1) 2 crossing fiber populations (2) 3 crossing fiber populations (3) with various microstructural characteristics. Front top view (a) and top view (b) are shown. Each color represents a single fiber population.*

	Δ (ms)	δ (ms)	Gradient magnitude ($mT.m^{-1}$)	B-value ($s.mm^{-2}$)	Gradient orientation count
Sequence 1	31.72	20.24	6.41	30	6
Sequence 2	31.72	20.24	9.07	60	6
Sequence 3	31.72	20.24	11.11	90	6
Sequence 4	31.72	20.24	12.84	120	6
Sequence 5	31.72	20.24	14.36	150	6
Sequence 6	31.72	20.24	15.74	180	6
Sequence 7	31.72	20.24	17.00	210	6
Sequence 8	31.72	20.24	18.18	240	6
Sequence 9	31.72	20.24	19.29	270	6
Sequence 10	31.72	20.24	20.34	300	6
Sequence 11	31.72	20.24	27.58	550	30
Sequence 12	31.72	20.24	31.15	700	30
Sequence 13	31.72	20.24	37.25	1000	30
Sequence 14	31.72	20.24	48.70	1700	60
Sequence 15	31.72	20.24	67.07	3200	60

Table 5.2: dMRI sequences settings for the simulation campaign, adapted from the settings from iCORTEX protocol.

5.3 Simulation campaign results and post-processing

The simulation campaign led to the creation of 10287, 15767, and 16619 samples, including respectively one, two, and three fiber populations, for a total of 42673 simulated samples. The overall time of the simulation campaign took approximately two months; still, it should be noted that the total simulation time was mainly driven by the TGCC Joliot-Curie supercomputer's scheduler since a job request can take several days to be validated. Independently from the scheduler, we have observed in the case of three fiber populations simulations, with 100 nodes containing 32 CPUs each, the synthesis of 5328 samples in 24 hours and so approximately 20 samples per hour and per node. This result is in accordance with the expected times drawn after the validation campaign (see chapter 4 4.3.2.1).

It is difficult to present a non-exhaustive representation of all the simulation campaign samples. However, the key geometric characteristics of some samples are shown in [figures **??**, **??**, **??**]. Precisely, [figure **??**] presents different samples according to their axon diameter and volume fraction, showing the wide variety of geometries synthesized during the campaign. [Figure **??**] presents two samples with different GAD, demonstrating the deterioration of geometry regularity for large GAD. Finally, [figure **??**] presents two samples of three fiber populations with a different distribution of their respective volume fractions, their volume fraction results from both the number of fibers in the samples and their diameter.

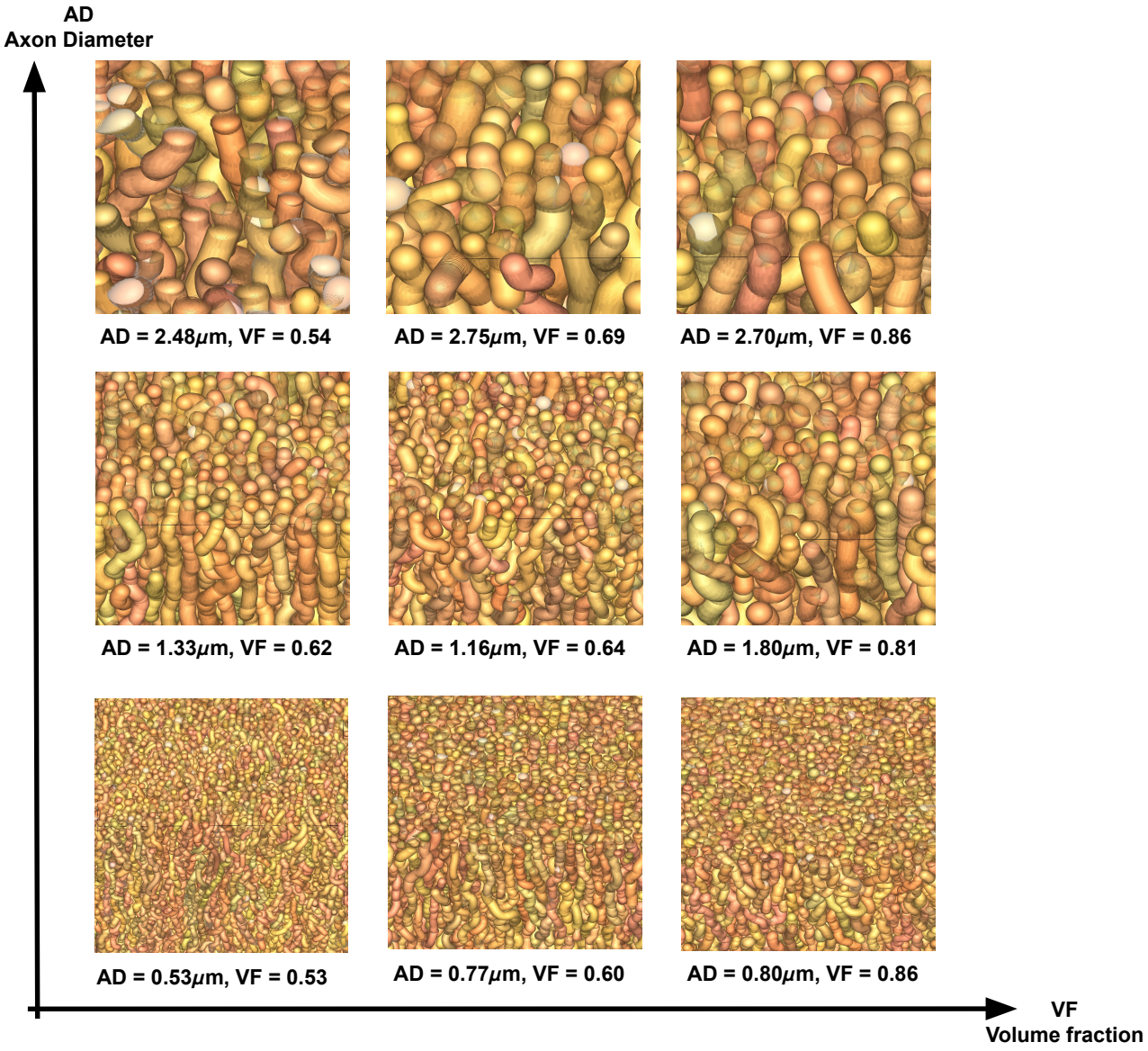


Figure 5.5: *Simulation campaign results: volume fraction (X-axis) and axon diameter(Y-axis) of different samples stemming from the simulation campaign data-set. The field of view is the same for each snapshot in order to catch the differences between the sample's axon diameters.*

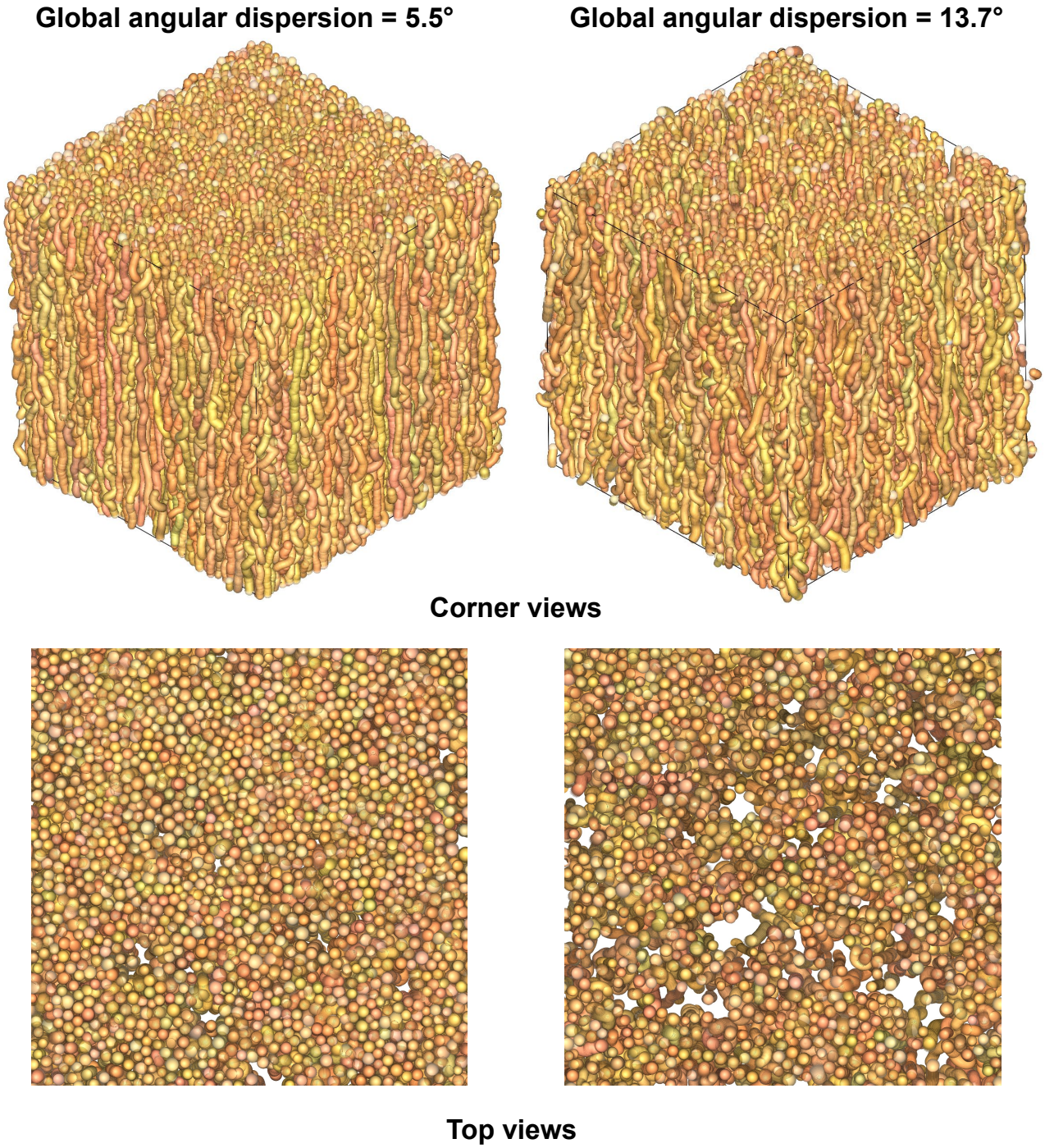


Figure 5.6: *Simulation campaign results: global angular dispersion (GAD) of two samples of one fiber population stemming from the simulation campaign is presented with two different point views (corner and top). The two samples reflect the minimal and maximal threshold from GAD distribution ([5°, 15°]). Visually, a significant difference is spottable between the two samples as the sample with more GAD shows more tortuosity resulting from the fibers crossing each other whereas the samples with low GAD show a certain regularity particularly spotted in the top point of view.*

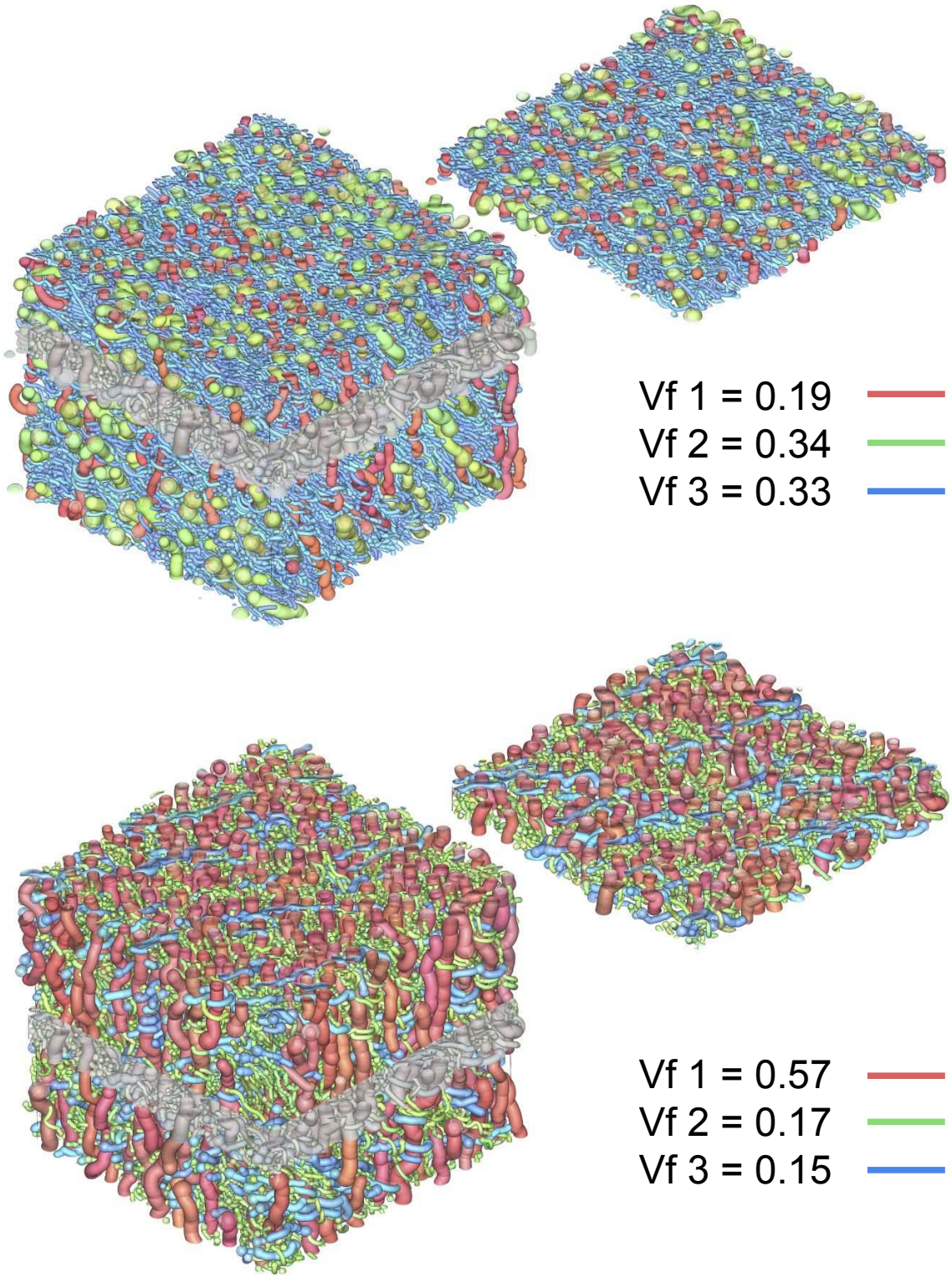


Figure 5.7: *Two samples from the simulation campaign are presented. They both include 3 fiber populations and a slice cut stemming from the center of the sample voxel is also delivered on the side. The populations are differentiated with color RGB, and their different volume fraction values Vf1, Vf2, and Vf3 are presented relatively to a specific population color.*

The results for each simulated sample are organized within a dedicated directory on disk containing the geometry and diffusion footprint information. The geometry description is stored in a JSON dictionary, whereas the simulated dMRI signal decay is stored in a sub-directory where each iCORTEX sequence has a dedicated file storing the dMRI attenuations relative to the number of gradient orientations from the sequence.

It is challenging to quantify the quality of the simulated dMRI signals directly from the simulation output; consequently, we designed a post-processing pipeline, which had a threefold purpose:

- Enabling us to study the dMRI signal decay in more detail through the use of dMRI reconstruction techniques of the ODF and associated metrics.
- Grouping each sample's geometry and diffusion footprint in a synthetic table.
- Increasing the database by further generating for each native sample a series of inherited samples with random rotation and noise.

The post-processing pipeline scheme is described in [figure 5.8].

5.3.1 Compressed representation of the dMRI signal attenuation

The samples synthesized after the simulation campaign were exported from the TGCC storage system and stored on the Neurospin storage system. At this point, the dMRI data correspond to the full set of dMRI signal attenuation acquired for the various iCORTEX dMRI sequences along the different gradient orientation sets; yielding 270 dMRI values for each sample.

To further take into account the parsimony of the diffusion propagator, each of the different shells composing the 270 values are further summarized by:

- collecting the rotationally invariant DTI metrics including the apparent diffusion coefficient (ADC), the radial diffusivity ($D_\perp$), the longitudinal diffusivity ($D_\parallel$), and the fractional anisotropy (FA) (see chapter 2 2.3.3.1),
- decomposing the signal attenuation on the current shell onto a spherical harmonics basis as proposed by [Descoteaux et al. 2007], resulting in a reduced set of spherical harmonics coefficient N_{SH} such as:

$$N_{SH} = \frac{(SH_{Order} + 1).(SH_{Order} + 2)}{2}\ , \tag{5.1}$$

  where SH_{Order} is the spherical harmonic count. In our case, $SH_{Order} = 4$ thus $SH_{Count} = 15$.

The original set of 270 diffusion MRI signal attenuations could be reduced to:

$$N_P = (N_{DTI} + N_{SH})\ .\ N_{Shell}, \tag{5.2}$$

with N_P the total number of diffusion parameters for one sample, N_{DTI} and N_{SH} the number of parameters resulting respectively from diffusion tensor imaging and spherical harmonics techniques and N_{Shell} the total number of sequences. Since we mimic the iCORTEX protocol, $N_{Shell} = 15$, thus $N_P = 285$.

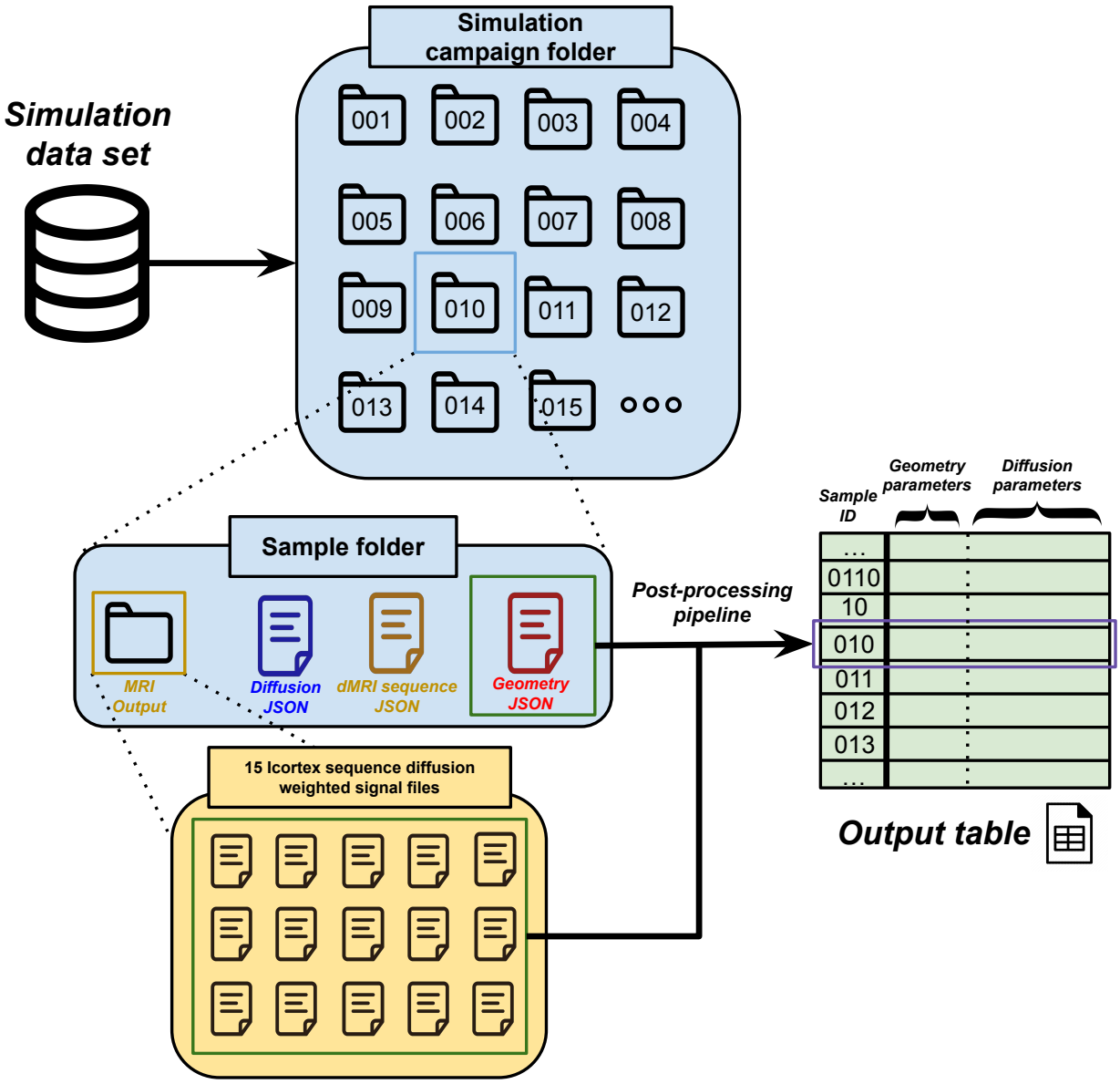


Figure 5.8: *Overall diagram of the post-processing pipeline: each simulated sample holds a folder in which their description input JSONs and dMRI signal decay output data are stored. A post-processing pipeline gathers the data from the simulation within a CSV table, with each line corresponding to a sampling and each row corresponding to a parameter.*

We arbitrarily decided to compress the original signal with these two representations, but alternative approaches could have been considered, such as polar harmonics decomposition as proposed by Ozarslan in his SHORE model [Özarslan, Koay, and Basser 2013] or Taylor series decomposition integrating orders higher than just the second-order tensor, as proposed by Fieremans, Novikov, and Kiselev [Novikov, Fieremans, et al. 2019].

5.3.2 From geometric and diffusion data to CSV representations

After completing the simulation campaign, the data is stored in different folders, each containing the geometric and diffusion information for each sample. It's tedious to read the data correctly in this state as approximately 8.5 million files are stored, for a total amount of 34.6 GBytes.

To collect geometric and diffusion information from the simulation campaign, a table representation in which rows represent the samples and columns represent their geometric and diffusion descriptions has been chosen. The first column gathers the geometry characteristics, and the following 285 columns gather the computed diffusion representation. As the number of geometrical parameters varies depending on whether the simulation is composed of one, two, or three populations, a specific table is drawn for each scenario with 7, 13, and 19 columns dedicated to geometry, respectively. [Figure 5.8] illustrates the structure of the output table created by the post-processing pipeline.

5.3.3 Data augmentation

The native simulated database does not take into account 2 further effects:

- the presence of noise corrupting the real dMRI signal attenuation; despite MEDUSA can add noise on the fly during the simulation of the signal attenuation, it was deliberately decided to keep the signal attenuation free of noise to be able to add it a posterior since it strongly depends on the hardware and imaging protocol;
- the rotation of the simulated geometry to arbitrarily represent all the inherited dMRI signal representations simply resulting from the rotation of the geometry and of its associate dMRI signature.

To take into account these two points, we artificially corrupted the dMRI signal attenuations using a noise standard deviation similar to that of the real dataset acquired in the frame of the iCORTEX project. In addition, we applied rotations to the virtual WM samples generated using the MEDUSA framework [figure 5.9].
The desired number of rotations sets the size of the final database. We choose to apply 100 rotations to each simulated sample, creating a dataset containing 4.267.300 samples. Nevertheless, this enlarged dataset does not come without a downside, as most parameters are invariant to the rotation. Indeed, only the mean orientation of fibers and spherical harmonics of order higher than one sees their values change; the first spherical harmonic corresponds to a sphere and thus is invariant to the rotation.

The histograms [figures 5.10, 5.11, 5.12,5.13,5.14,5.15] present the distributions of the geometry and diffusion parameters from the data set after rotation and noise algorithm application. The histograms are separated according to the total number of fiber populations within the samples and provide the distribution of the parameters according to all populations within the sample. The distribution of the AD, GAD, and VF presented in the histograms match the desired distribution bandwidth from

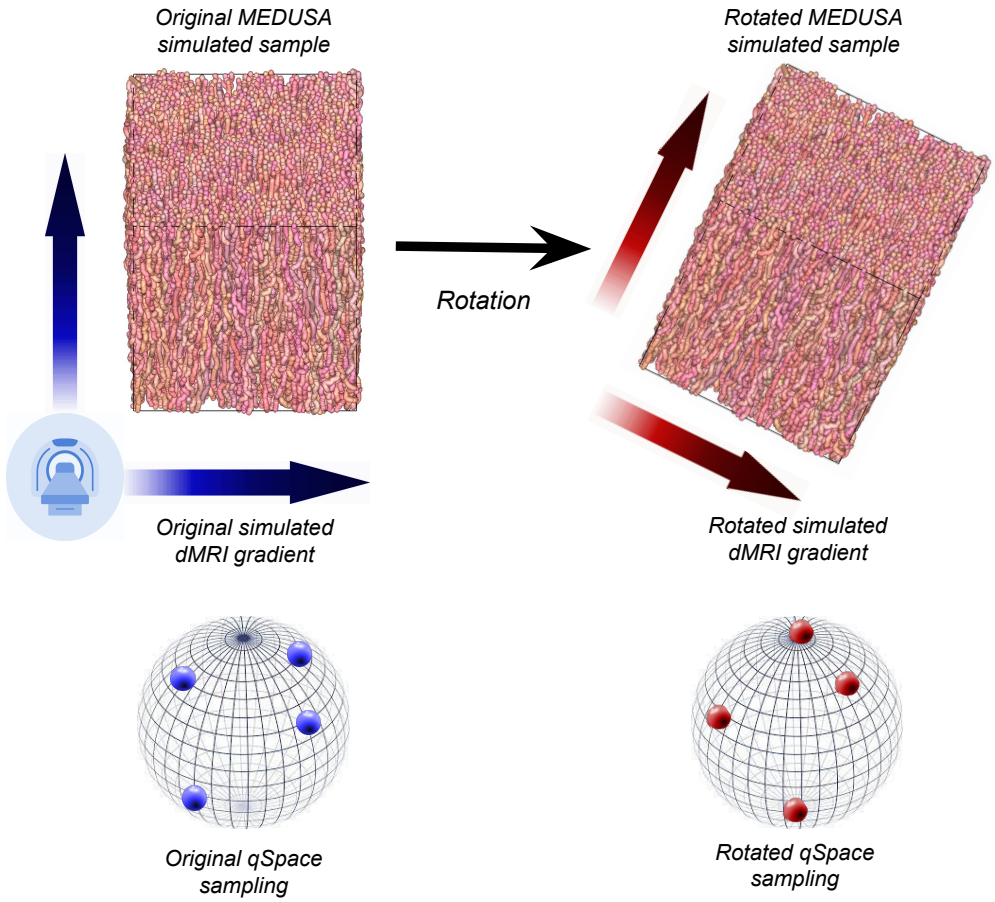


Figure 5.9: *Illustration of the rotation algorithm: one fiber population simulated samples and its dMRI signature is rotated, resulting in a resampling of the q-space.*

[table 5.1]. More precisely, we observe an exponential decrease of the VF histogram in the case of three fiber populations, which is explained by the relation we set between the VF of the different fiber populations, as the total should not go beyond 1.0. The distribution of the fiber population mean orientation along the X and Y axes from the cartesian basis depicts maxima for -1 and 1 values while mean orientation along the Z axis depicts maximum value for 0 value. That is explained by the fact that we arbitrarily decide to orient the first fiber population along (0, 0, 1), and the rotation applied to the mean orientation is done through the use of a rotation matrix tuning pitch, yaw, and roll rotations according to angles distributed uniformly between 0 to 2π resulting in an oversampling of mean orientations along XY plan.

Concerning the tensor diffusion parameters, ADC, $D_{\perp}$ and $D_{\parallel}$ values less than 1.0 to 1.1 $10-9m2/s$ are generally acknowledged for white matter [Radiopaedia 2005-2024] which is in line with the histograms. FA are unitless, varying between 0 (maximal isotropic diffusion) and 1 (maximal anisotropic diffusion) as expected. Only spherical harmonics of order 0, 3, 9, and 12 are represented. The missing spherical harmonics histograms show Gaussian distributions similar to the spherical harmonics of order 6, 9, and 12. Precisely, spherical harmonics range from -10^{-1} to 10^{-1} for orders 1, 2, 3, 4, 5 and 6; -10^{-2} to 10^{-2} for order 7, 8, 9, 10, 11 and 12; and finally -10^{-3} to 10^{-3} for order 13 and 14. We are observing non-Gaussian distribution of spherical harmonics of order 0 and 3. The consistency of diffusion parameters is difficult to establish using histograms alone. ODFs were built from randomly picked simulations to check the coherence of the diffusion signature with the mean orientation of the fiber population. This check was done only for a few samples and, therefore, does not ensure the accuracy of the entire data set.

The post-processing pipeline benefits as well from the parallelization of its code with a total rotation count of 100. The rotation and noising of 38600 original samples generated 3.860.000 samples and took 1 hour and 48 minutes to achieve on Intel Xeon 2.60GHz desktop station with 16 CPUs.

5.4 Conclusion

The simulation campaign and its results presented in this chapter are the culmination of several previous simulation campaigns performed during the time of this thesis. Indeed, the large number of parameters and the complexity of the simulations, combined with the campaign's length and weight, made several attempts necessary to reach the outputs presented. In the future, we can expect new simulation campaigns with more samples synthesized and different parameters distributed. The next chapter presents the deep learning model developed to decode the microstructure of cerebral WM using the data set synthesized during the simulation campaign to train its neural network.

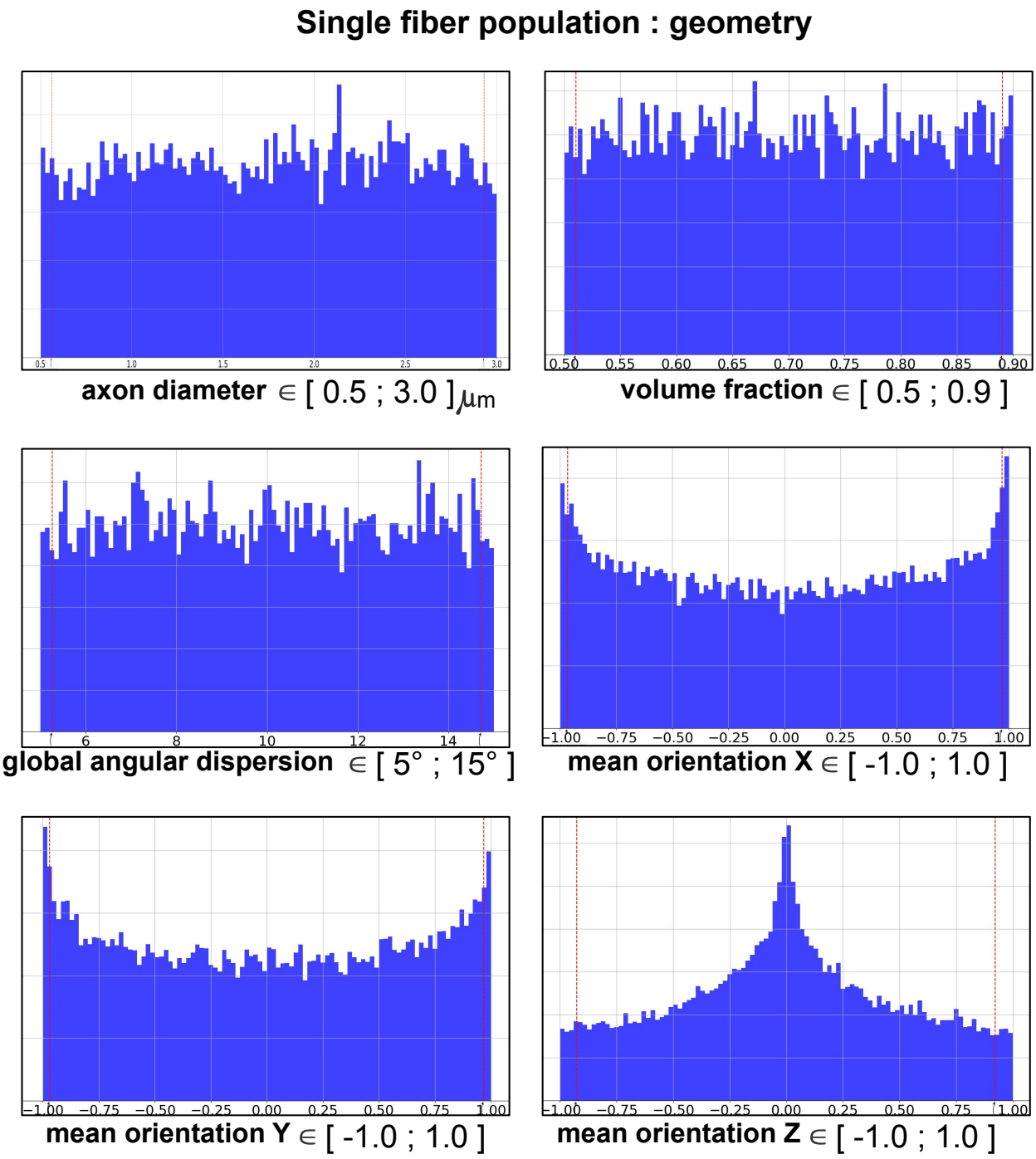


Figure 5.10: *Histograms of geometry parameters from simulations gathering one simulated fiber population. The histograms gather parameters stemming from 926.000 different samples.*

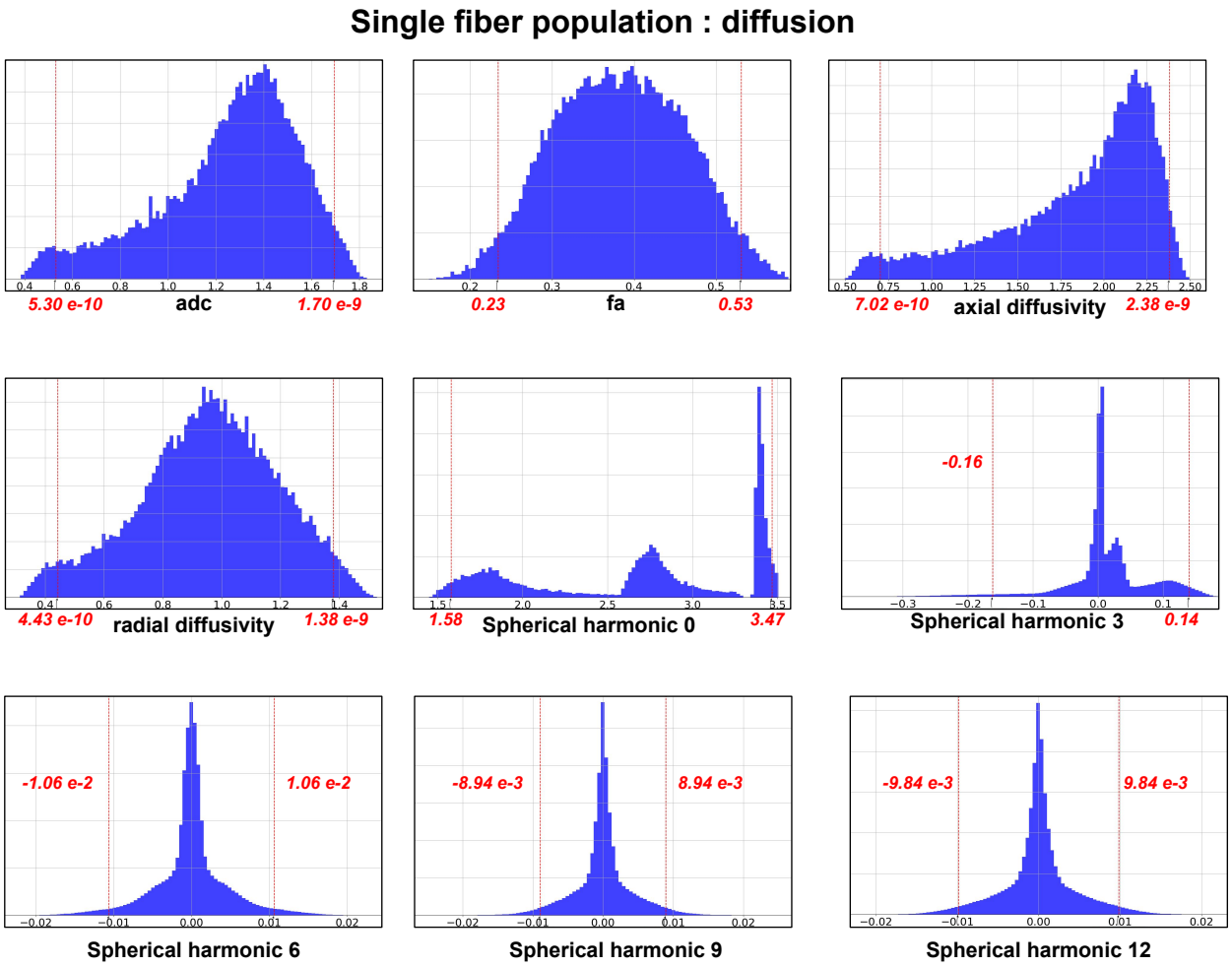


Figure 5.11: *Histograms of diffusion parameters from simulations gathering one simulated fiber population. In red and framing each histogram, each parameter's first and last quartile value. The histograms gather parameters stemming from 926.000 different samples.*

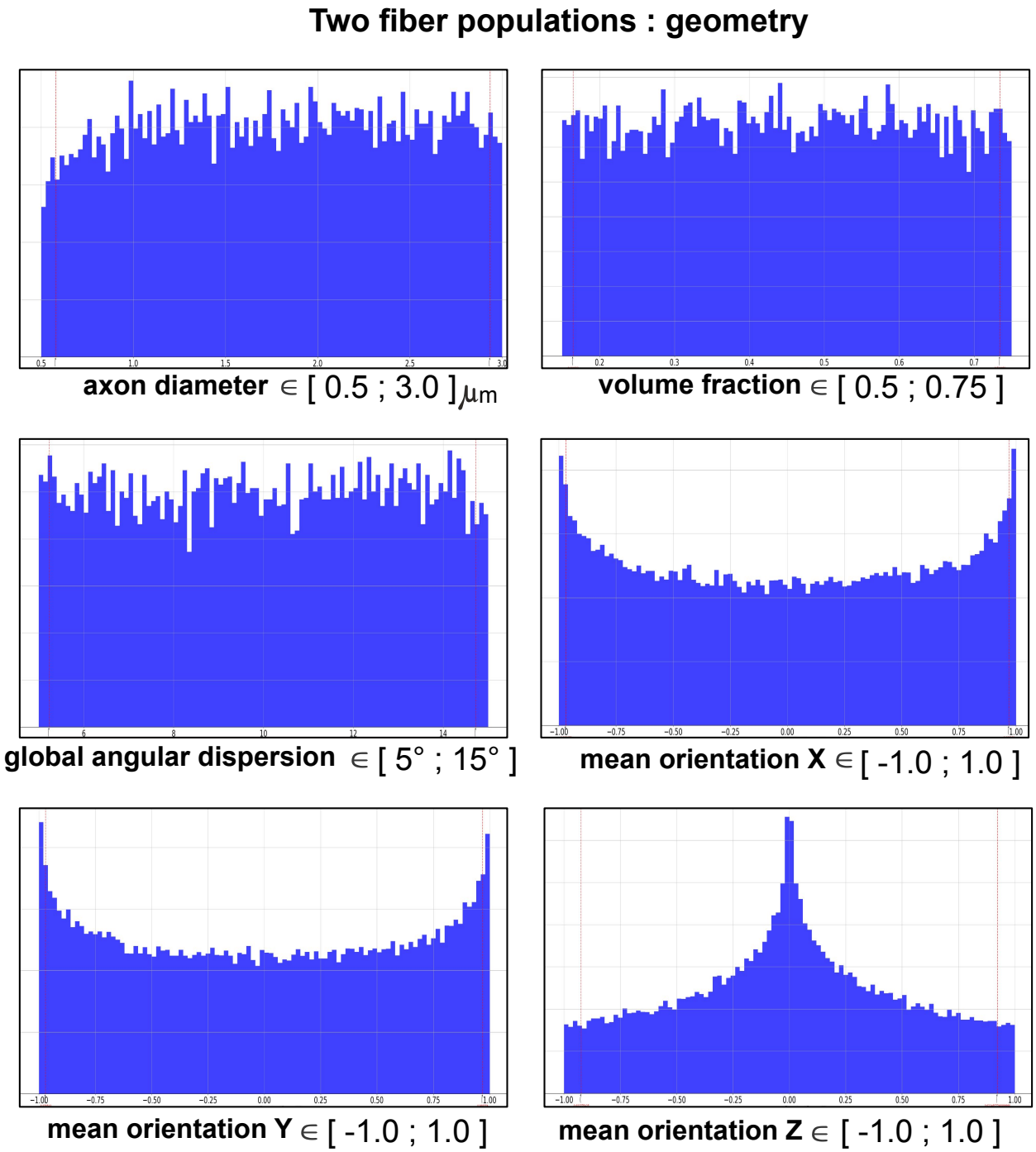


Figure 5.12: *Histograms of geometry parameters from simulations gathering two simulated fiber populations. The histograms gather parameters stemming from 1.496.700 different samples.*

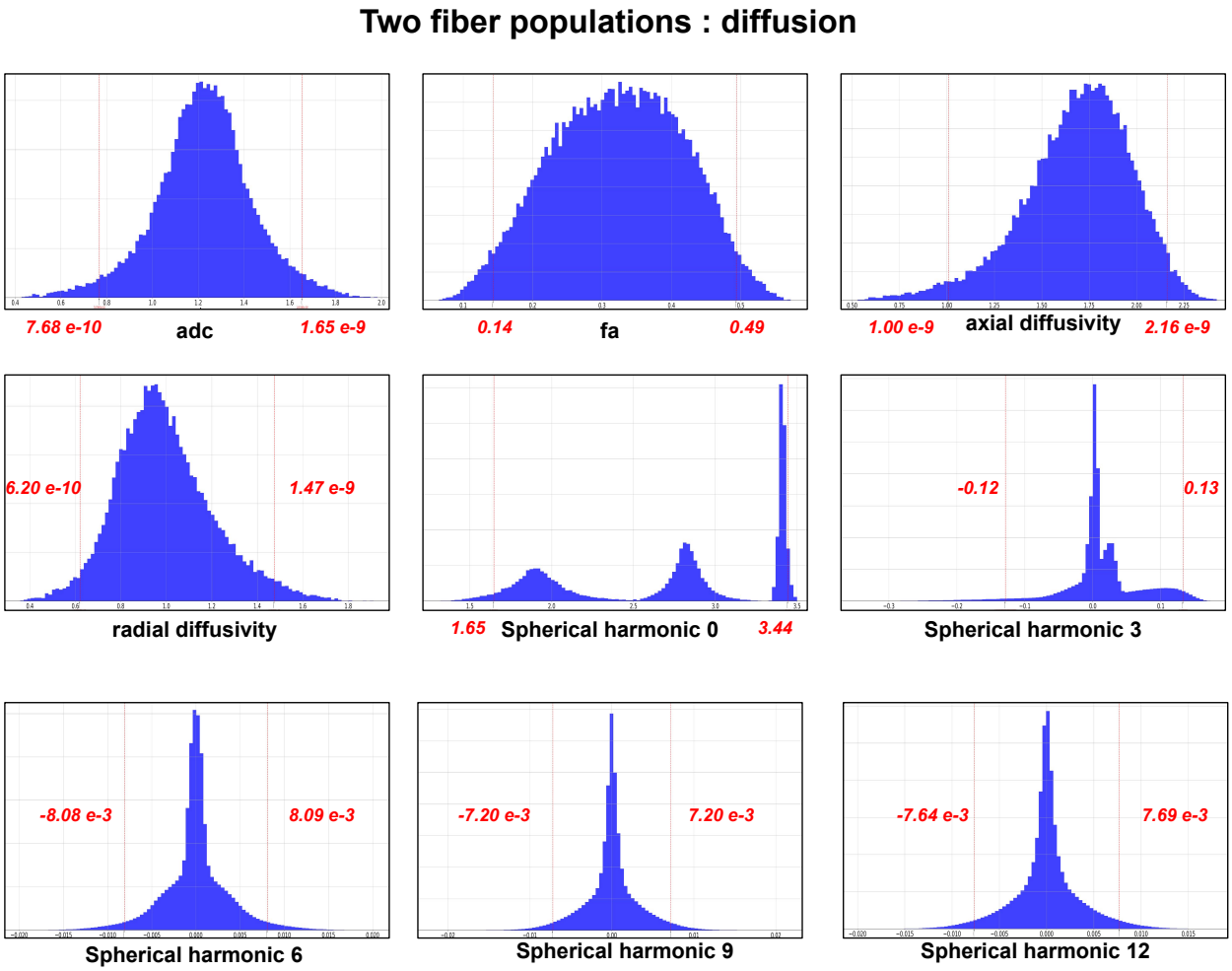


Figure 5.13: *Histograms of diffusion parameters from simulations gathering two simulated fiber populations. In red and framing each histogram, each parameter's first and last quartile value. The histograms gather parameters stemming from 1.496.700 different samples.*

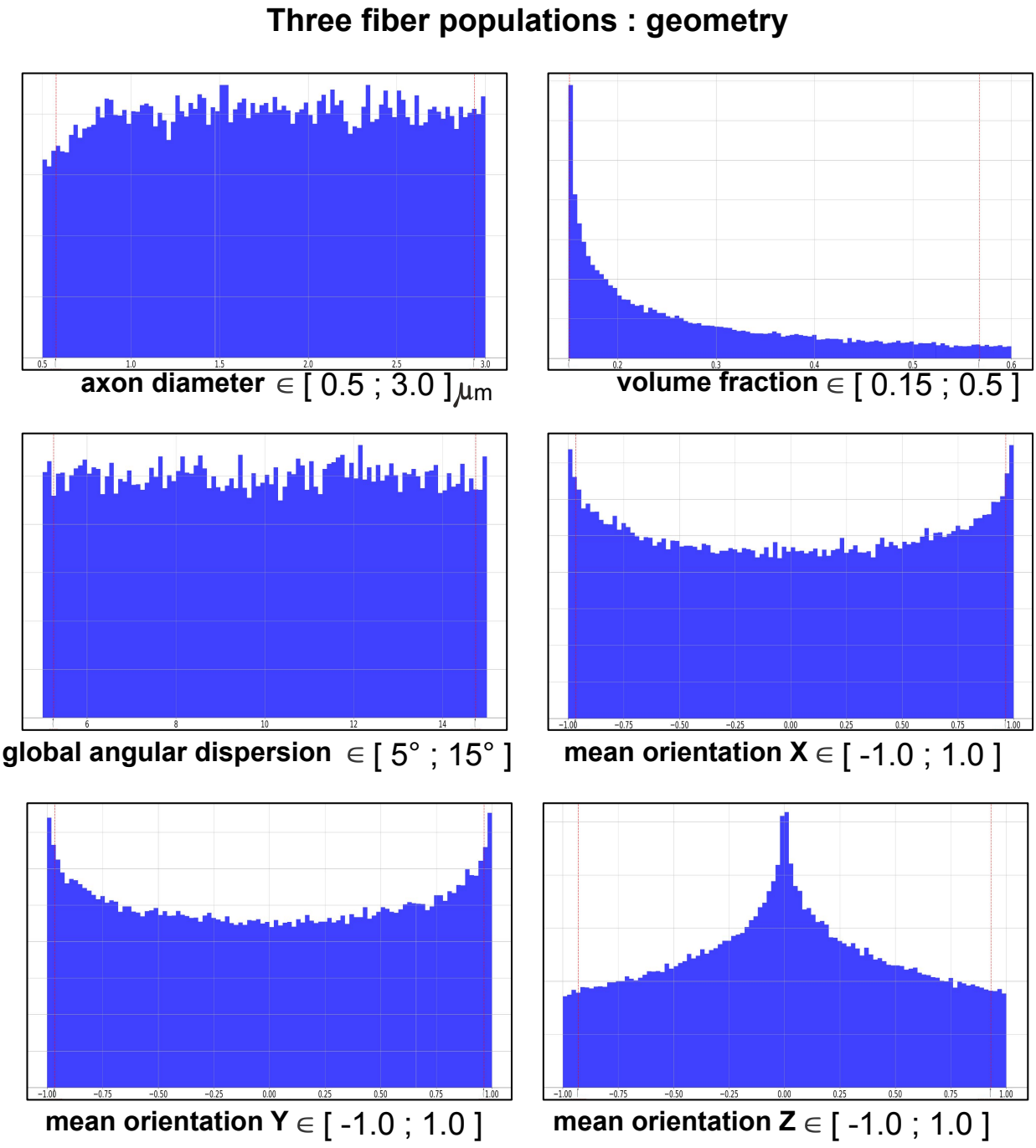


Figure 5.14: *Histograms of geometry parameters from simulations gathering three simulated fiber populations. The histograms gather parameters stemming from 1.437.300 different samples.*

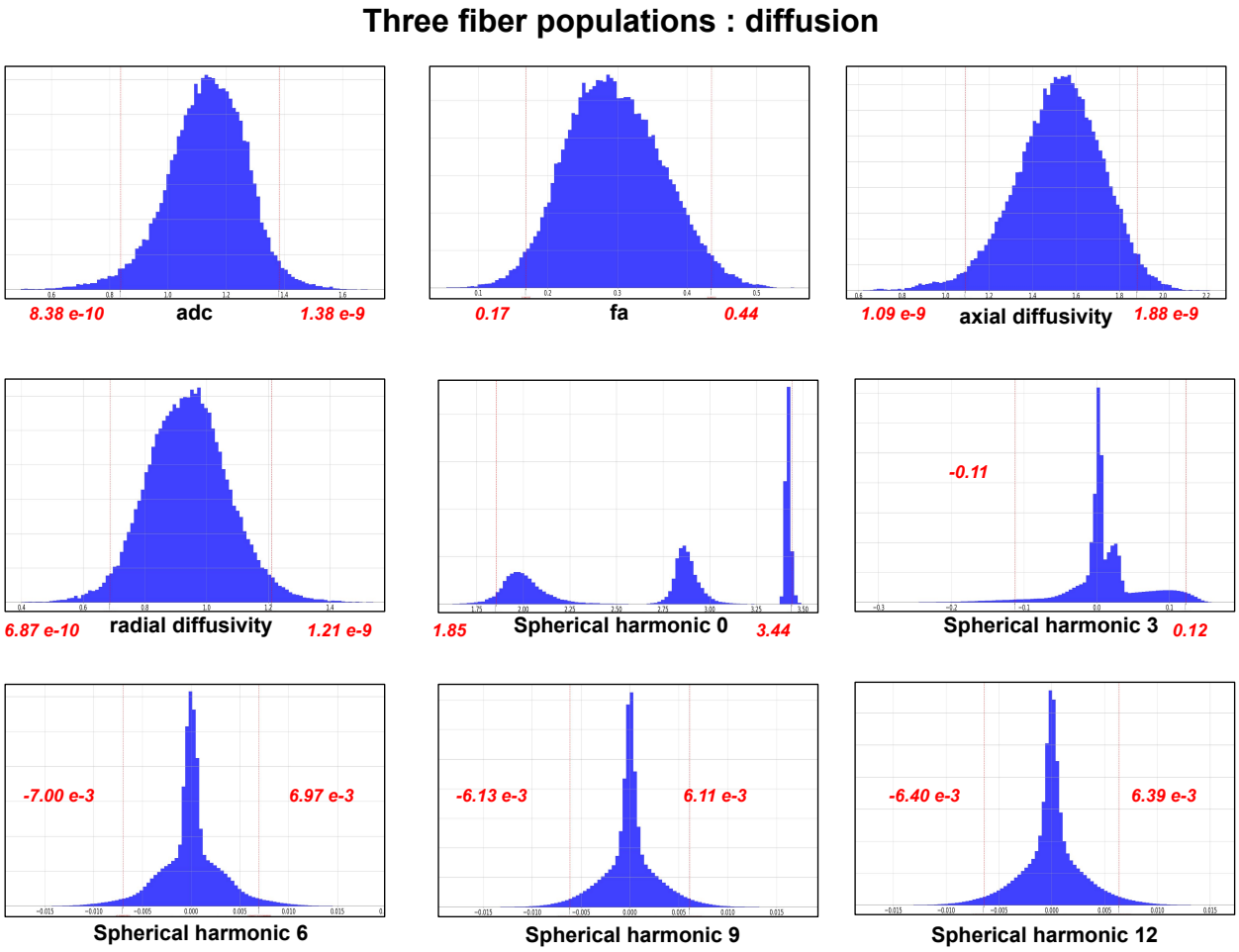


Figure 5.15: *Histograms of diffusion parameters from simulations gathering three simulated fiber populations. In red and framing each histogram, each parameter's first and last quartile value. The histograms gather parameters stemming from 1.437.300 different samples.*

Chapter 6

Implementing a neural network decoding the white matter microstructure

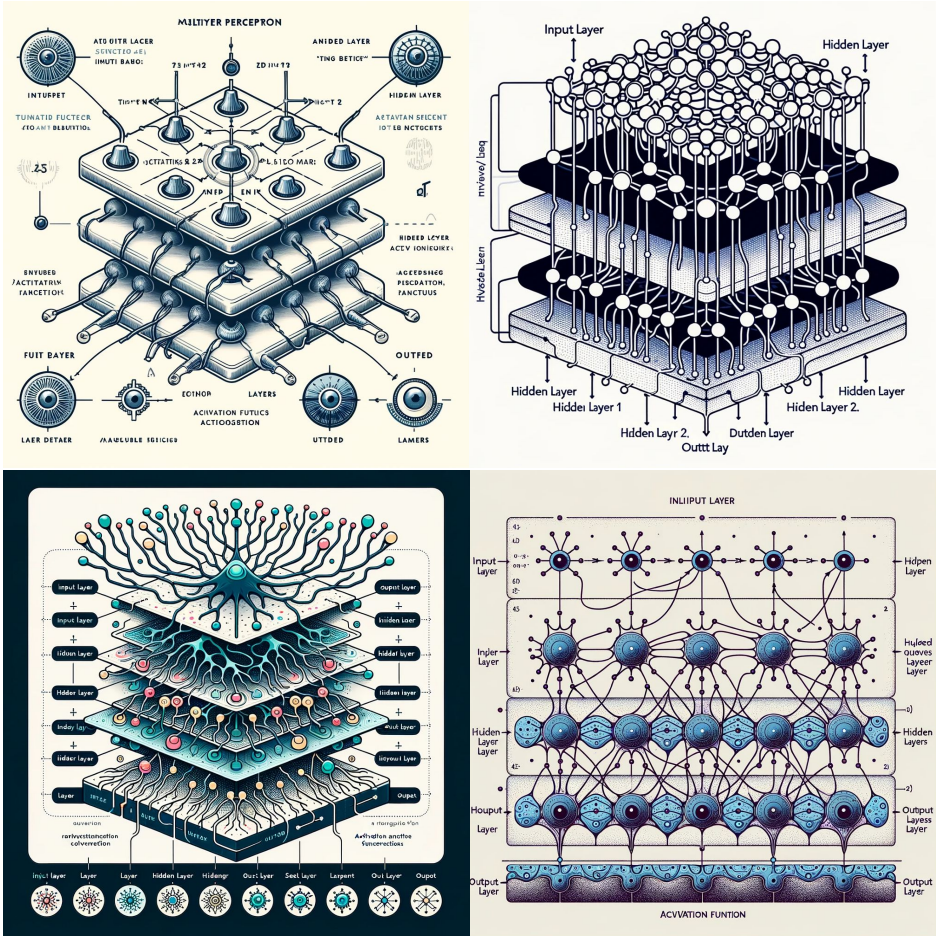


Figure 6.1: *Artificially generated pictures depicting a multilayer perceptron neural network. These images have been generated with the online AI tool "dall-e 3" [openai 2024].*

Chapter outline

6.1 Machine learning algorithms 163
6.1.1 Linear regression 163
6.1.2 K-nearest neighbour 163
6.1.3 Support-vector machine 164
6.1.4 Decision tree 164
6.1.5 Random Forest 164
6.1.6 Extra trees 164
6.2 Neural network: an overview 165
6.2.1 Historical background 165
6.2.2 Multi-layer perceptron architecture 167
6.3 MEDUSA neural network implementation 174
6.3.1 General design of the decoding pipeline 174
6.3.2 Tuning of the NEDUSA white matter neural network 174
6.3.3 Training and testing the networks 180
6.3.4 Discussion 194
6.3.5 Conclusion 196

Machine learning (ML) is a subtopic of artificial intelligence (AI) involving algorithms trained to identify recurring patterns in data sets. Unlike analytical models, ML's purpose is not to determine the exact function or relationship characterizing a problem but to propose a model whose predictions are consistent with empirical observations of the phenomenon. The benefit of such an approach is that it is only limited by the size and quality of the training dataset stemming from the phenomenon to study, without regard to any model hypothesis inherent to a mathematical expression. ML functioning leads to a limit, or rather an intellectual frustration, because the result expressed by the ML algorithm can be translated into a logic escaping from human sense, evolving with a black box logic. As an illustration to the previous statement, Ali Rahimi, a researcher in artificial intelligence, assessed that ML algorithms have become a form of "alchemy" [Hutson 2018], thus depicting the logic inherent in the optimization of ML algorithms, beyond absolute mathematical rigor.

Nevertheless, ML revealed itself to be a powerful engineering tool for research, as illustrated by the great number of scientific papers published on this topic over the years, with a daily publication of around 100 papers related to ML reported in 2019 [data-mining 2019]. When it comes to ML, several steps of reflection must be taken into account to ensure a good implementation of the algorithm, as presented hereafter:

- One should determine and organize a training data-set feeding the ML model. In our case, the data-set results from the simulation campaign presented in the previous chapter and organized as a table of parameters gathering millions of MEDUSA simulated sample parameters describing both WM microarchitectures and their corresponding dMRI signatures. Data-set management

is crucial because an under- or over-sampling of the learning vectorial space can lead the ML algorithm to make inaccurate predictions, drifting away from the empirical observations.

- The type of ML algorithm should be chosen according to the task it is intended to perform, the dataset's size, and the available computing resources. Once chosen, the ML algorithm is trained using the training data-set.

- Finally, the ML algorithm is exposed to data stemming from real situations where parameters are potentially distributed differently from the training data-set. This step reveals the ML algorithm's robustness, facing all the specificities from real case scenarios.

This chapter presents the state of the art concerning the different existing ML algorithms with an emphasis on the neural network approach as it is the one chosen for our model. The design of our microstructure decoding pipeline will be presented as well as the performance of these models in terms of learning capability and precision. Ultimately, this chapter will discuss the approach, and propose some work in progress and improvements to be achieved in the future concerning our model.

6.1 Machine learning algorithms

The first use of the term machine learning dates back to 1959, following the creation of the first program playing draughts that learns and improves while playing [Samuel 1959]. The advent of ML is therefore largely linked to the second half of the 20th century, correlated to advances made in computer science. The following section proposes to list and describe some of the most popular ML models relevant in the field of supervised regression problems, which corresponds to our situation aiming at proposing a decoding model for WM microstructure.

6.1.1 Linear regression

Linear regression [Gauss 1809; Legendre 1805] is a straightforward approach for tackling regression problems with a few but strong assumptions. First, it is assumed that the relationship between features x and target y is approximately linear, such as the conditional mean $E[\mathrm{Y}|\mathrm{X} = \mathrm{x}]$ can be expressed as a weighted sum of the features x. This setup allows the target value to deviate from its expected value with respect to an observation noise which is assumed to follow a Gaussian distribution [d2l 2024]. Linear models are by construction restrictive, which can weaken their predictive performance due to an oversimplification of the problem.

6.1.2 K-nearest neighbour

The K-Nearest Neighbors (KNN) algorithm tackles classification and regression problems. It is a non-parametric approach that does not make any underlying assumption about the distribution of data. The KNN algorithm finds the nearest neighbors to a given data point based on a distance metric, such as the Euclidean distance. The class or value of the data point is then decided by the preponderance vote or average of the K neighbors. This approach allows the algorithm to adapt to different patterns and make predictions based on the local structure of the data [Gfg 2024].

6.1.3 Support-vector machine

Developed in the 1990s [Vapnik, Golowich, and Smola 1996], the support-vector machine (SVM) model distinguishes different classes within the data by finding the optimal hyperplane that maximizes the margin between the closest data points of opposite classes, which can be used for classification and regression problems. The number of features in the input data determines the dimension of the hyperplane space. The lines that are adjacent to the optimal hyperplane are known as support vectors as these vectors run through the data points that determine the maximal margin. These algorithms are widely used in machine learning as they can handle both linear and nonlinear classification and regression tasks. However, when the data is not linearly separable, kernel functions are used to transform the data into higher-dimensional space to enable linear separation [IBM 2023].

6.1.4 Decision tree

a decision tree is a supervised machine-learning algorithm, which is used for both classification and regression problems and built as a flowchart-like tree structure where each internal node denotes a test performed on an attribute. Each branch represents an outcome, and each leaf node holds a class label. A decision tree is constructed by recursively dividing the training data-set into subsets established on the values of the attributes until a stopping criterion is satisfied, such as the maximum depth of the tree or the minimum number of samples required to split a node [Gfg 2023a]. However, trees do not handle linear relationships as any linear relationship has to be approximated by splits, creating an inefficient step function. Moreover, slight changes in the input feature can have a big impact on the predicted outcome. Trees are also quite unstable as a few changes in the training dataset may lead to a completely different tree.

6.1.5 Random Forest

A Random Forest (RF) algorithm constructs multiple decision trees, thus forming a forest. The results from several combined trees are likely to be better than those from a single tree. Given a dataset with several features, an RF algorithm will sample subsets of observations with different features from the dataset during a process known as ”bootstrapping”. When assembling the decision tree, RF chooses the most optimal split at each node, this process is repeated on a different subset of the data with varying features until the specified number of trees has been assembled. After obtaining results from all the trees, the final prediction will be obtained through majority voting for classification or averaging for regression.

6.1.6 Extra trees

The Extremely Randomized Trees algorithm, also known as Extra Trees (ET), constructs multiple trees similarly to RF algorithms. Different subsets of the data may introduce different biases in the results obtained, hence ET prevents this by sampling the entire dataset. Extra Trees are also reducing variance thanks to the randomized splitting of nodes within the decision trees.

Several computational models have shown potential in decoding the microstructure of WM, based on Random forest [Nedjati-Gilani 2017] or Extra trees [K. Ginsburger 2019]. [Figure 6.2] illustrates the different machine learning presented. Nevertheless, with a database containing several million samples

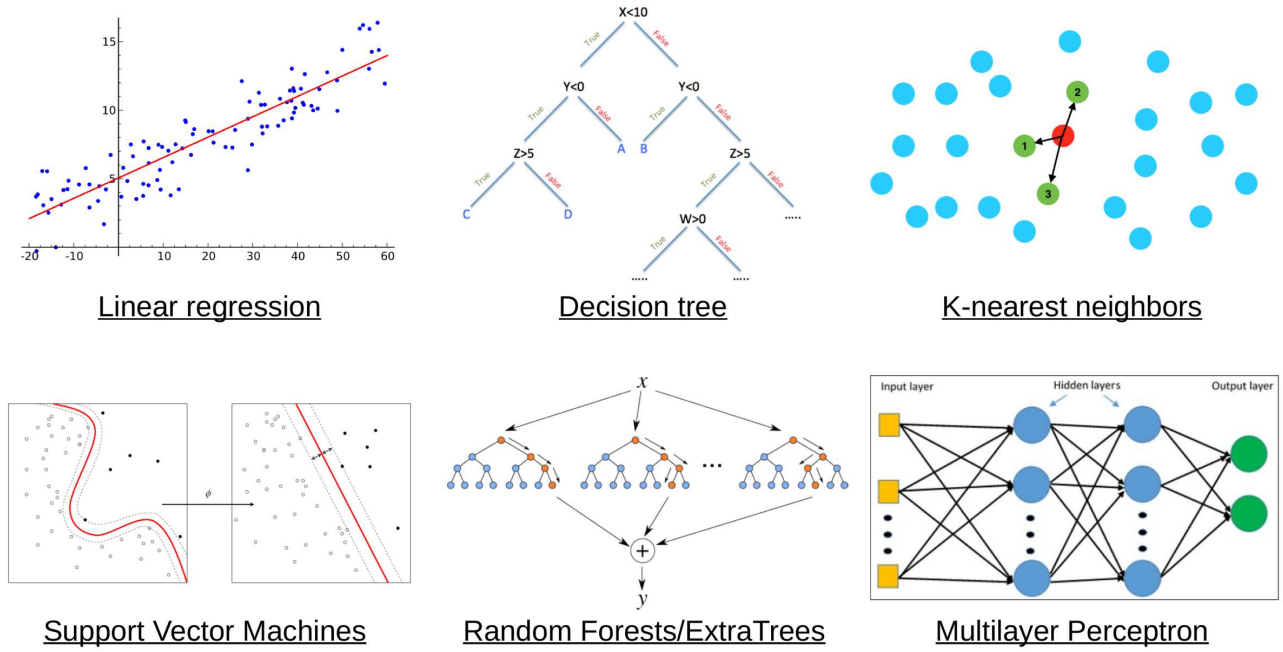


Figure 6.2: *Illustration of the principle of important machine learning algorithms for supervised learning of tabular data. Adapted from [K. Ginsburger 2019]*

and given the popularity of deep learning in the world of ML, we decided to explore the potential of a neural network approach trained to decode WM microstructure from its dMRI signature.

6.2 Neural network: an overview

Before introducing the neural network architecture used to establish a MEDUSA computational model of MW microstructure, a brief overview of the history of neural networks is first presented to introduce the concepts.

6.2.1 Historical background

In 1943, McCulloch and Pitts [McCulloch and Pitts 1943] modeled an artificial neuron as a regulator that receives input from other neurons and, depending on the total weighted input, is either activated or remains inactive. These weights can be positive to be excitatory or negative to be inhibitory. In 1958, Rosenblatt [Rosenblatt 1958] demonstrated that simple networks of such models of artificial neurons called "perceptrons" could learn from examples. It was shown that networks of such models of artificial neurons have properties that can be compared to the cognitive capacities of the human brain as they can perform complex pattern recognition, and the weights, by which the input from other artificial neurons is multiplied, can be assimilated to synaptic potentials. Moreover, like the human brain, neural artificial networks proved to operate even if some of the neurons are destroyed. Nevertheless, Minsky and Papert [Minsky and Papert 1969] demonstrated that perceptrons alone could only solve the very limited class of linearly separable problems.

In the '70s, the first multilayered neural network was designed [Fukushima 1975], followed in the 80s, by the research on "error back-propagation method" [Rumelhart, Hinton, and Williams 1986; LeCun et al. 1988], which made complex networks of neurons learn from examples, succeeding in solving problems that were not linearly separable. Nevertheless, neural network methods, also known as deep

learning methods, had to wait for two more technological revolutions before becoming the essential method of machine learning it is today:

- GPU - The first revolution came from the advent in the late 90's of GPU architectures, allowing faster processing of parallel tasks, particularly suited to deep learning problems. Thanks to GPUs, deep learning algorithms began to compete with other machine learning models such as support vector machines, offering better results using the same training data-set [dataversity 2024].
- Big Data - The second revolution started in the 2000s with the increase in the range of sources and types of data available for neural network training, linked with the growth of the internet. Accordingly, in 2009, Fei-Fei Li [Fei-Fei, Deng, and K. Li 2009] assembled a free database of more than 14 million labeled images called ImageNet, announcing the era of Big Data.

Since the 2010s the complexities of neural networks and their accuracy have increased inexorably in parallel with the improvement of GPUs and the wide spreading of HPC facilities. A non-exhaustive list of the remarkable breakthroughs made in neural network research is listed below:

- Google®'s "Cat Experiment" creating a neural network learning to recognize cats by watching unlabeled images taken from frames of YouTube videos [Le 2013].
- Facebook®'s DeepFace system using neural networks identifying faces with 97.35% accuracy [Taigman et al. 2014].
- Advent of the transformer architecture [Vaswani et al. 2017] allowing faster parallel training of neural networks on sequential textual data, originating the current enthusiasm for AI virtual assistants.

The popularity and rise of neural network models undeniably participated in ensuring our decision to use deep learning for our model. However, it is necessary to remember that a large number of the networks that have recently made deep learning so popular, like vision detection or chatbots, respond to problems far from the one here discussed. Indeed, our problem here is to train a neural network to predict the microstructure of WM according to its dMRI signature based on a labeled table dataset, which makes it a supervised regression problem. The main components of a neural network solving a supervised regression problem are detailed in the following section.

6.2.2 Multi-layer perceptron architecture

A neural network can be assigned different tasks such as classification, computer vision, or noise filtering, each requiring specific neural network and optimization methods. The Multi-Layer Perceptron (MLP) architecture was shown to be suited for supervised regression in numerous papers [Murtagh 1991; Agirre-Basurko, Ibarra-Berastegi, and Madariaga 2006; Nassif, Ho, and Capretz 2013]. MLPs are part of the feed-forward networks family characterized by the direction of the flow of information between their neuron layers. MLP has the particularity to fully connect the neurons from each layer to the neurons of the following layer as presented in [figure 6.3a].

The input layer constitutes the first layer of an MLP whereas the output layer is the last layer of an MLP. Input and output layers are respectively composed of as many neurons corresponding to the different parameters defining the input and the output of the data stemming from the problem we wish to tackle. The information, or impulse, from the input, is carried from the input layer to the output layer, passing by the intermediary hidden layers composing the MLP. The impulse, from a neuron, is weighted by a parameter judiciously called "weight". These weights determine the importance of a given impulse, with the largest weights contributing more significantly to the output.

Let us describe the functioning of an MLP with N hidden layers: the impulse x_j^l produced by a neuron j from a layer $l \in$N is defined by the following equation:

$$x_j^l = \sum_{k=1}^{n} h_k^{l-1} w_{jk}^l + b_k^l, \tag{6.1}$$

with w_{jk}^l the weight associated with the connection between the neuron k in layer $l-1$ and the neuron j in layer l, b_k^l the bias associated with neuron k in layer l and h_k^{l-1} being the output impulse from

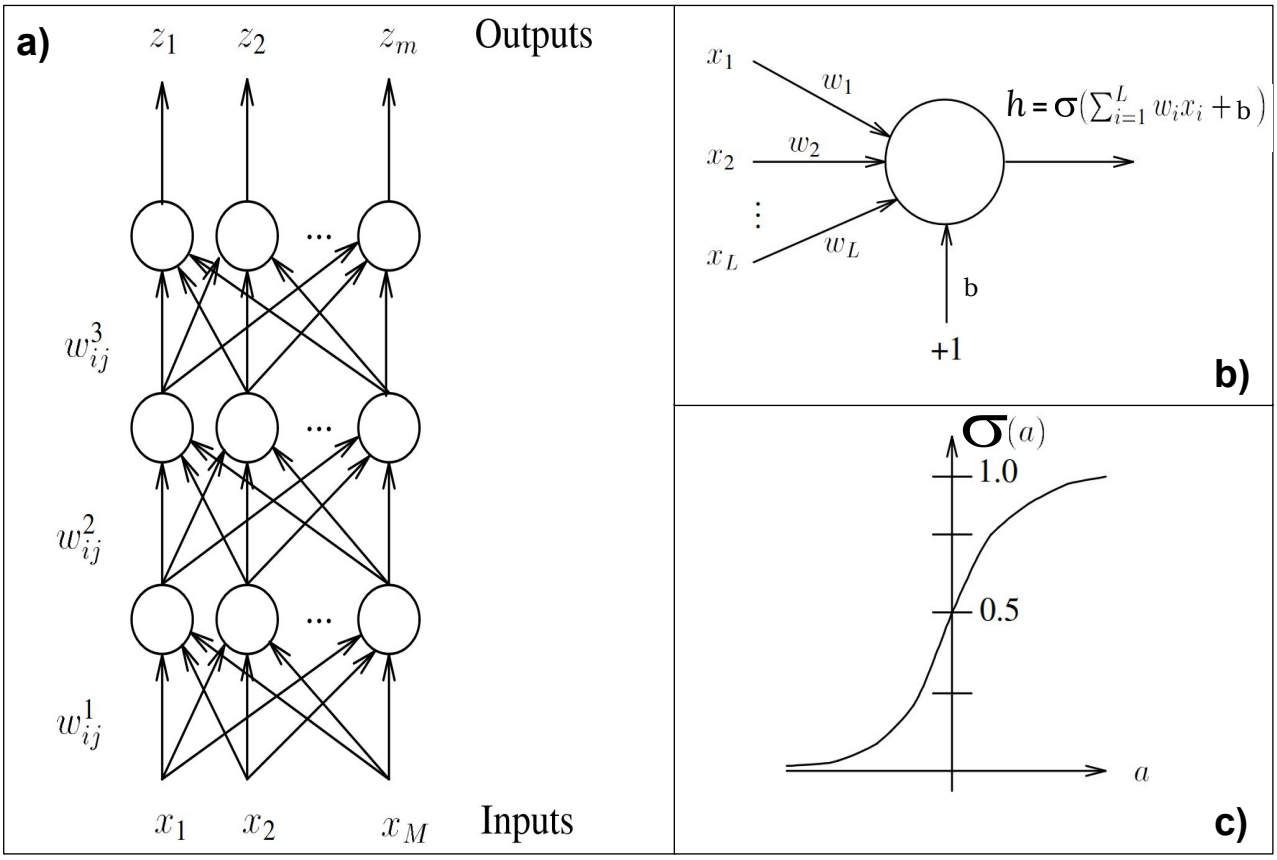


Figure 6.3: *a) Overall structure of a multilayer perception with x_{ji} the output impulse of node i to a node j, and w_{kij} the weight connecting node i in layer k-1 to node j in layer k. Layer 1 is the first hidden layer, and the inputs can be considered Layer 0. Predicted parameters are presented as the output Z. b) Zoom on a neuron j from layer k: the input impulses x_{ji} from neurons belonging to the previous layer k-1 are weighted by w_{kij} and summed. A bias b is added to the summed impulse and the total is entered in an activation function σ informing the excitatory or inhibitory state of the neuron. c) Example of a sigmoïd activation function. Adapted from [Ruck, Rogers, and Kabrisky 1990]*

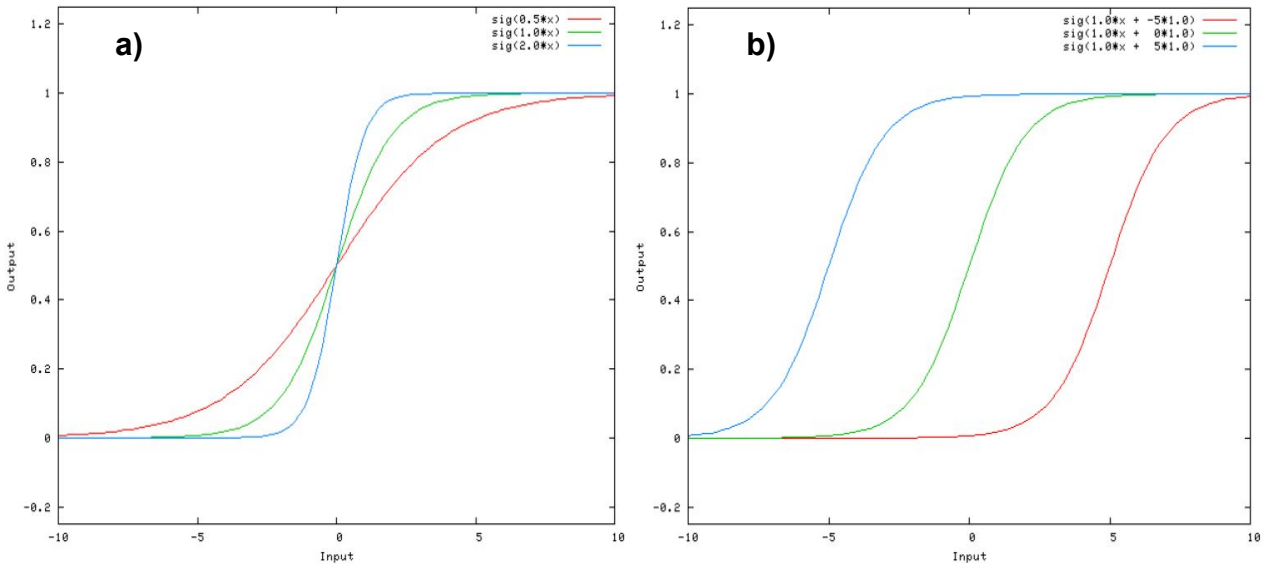


Figure 6.4: *Effect of bias and weight on activation function: a) Plot of sigmoids without bias, passing by the origin and weighted by w value equal to 0.5, 1.0, and 2.0. b) Plot of sigmoids with weight w=1.0 and bias equal to -5, 1, and 5. Adapted from [A. W. He 2017]*

neuron k in layer $l-1$ such as:

$$h_k^{l-1} = \sigma(w_{jk}^{l-1} x_k^{l-1} + b_k^{l-1}), \tag{6.2}$$

with σ the activation function, w_{jk}^{l-1} the weight associated with the connection between the neuron k in layer $l-2$ and the neuron j in layer $l-1$, x_k^{l-1} the impulse produced by a neuron k from a layer $l-1$ and b_k^{l-1} the bias associated with neuron k in layer $l-2$.

The impulse from layer l is then itself transmitted to layer $l+1$ and formalized by a h_k^l function. This iterative process of transmitting the impulse from one layer to the next defines this neural network as a forward propagation network, as illustrated by [figure 6.3a/b]. The activation function adds non-linearity to the neural network model and different functions can be used including the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, the hyperbolic tangent function $tanh(x) = \frac{\sinh(x)}{\cosh(x)}$, the exponential function e^x, and the standard rectified linear unit (ReLU) function $\text{ReLU}(x) = \max(0, x)$ [Pytorch 2024]. The bias acts as a translation of the activation function. Without bias, the activation function would be restrained from passing via the origin [figure 6.4], which would limit the flexibility of the model. By introducing biases, the model can better capture the complex patterns and non-linear relationships in the data, which generally leads to better prediction performance [Turing 2024].

The training of the neural network consists of tuning the weights and biases from every neuron to build a network correctly identifying the targetted problem. This training is achieved by an iterative submission of data to the network to improve its parametrization. The weights and biases associating a layer $l-1$ to a layer l can be formalized using a matrix representation:

$$W_{l-1,l} = \begin{bmatrix} w_{11} & \dots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{m1} & \dots & w_{mn} \end{bmatrix}, \qquad B_{l-1,l} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}, \tag{6.3}$$

with m being the number of neurons in the layer $l-1$ and n the number of neurons in layer l. B is a column matrix as only one bias is associated with each neuron. The matrix W_N and B_N regroup the

whole set of weights and biases associating the N layers of the neural network. The size of the matrix W_N and B_N characterizes the complexity of the MLP. It is reported that modern neural network models can contain up to a trillion parameters [Thirunavukarasu et al. 2023].

Various algorithms can implement the training of the neural network [Robotics 2020], but one of the most prominent ones is the back-propagation algorithm [LeCun et al. 1988; Rumelhart, Hinton, and Williams 1986]. To introduce back-propagation, let us first define the target function F_D that the neural network wants to approximate, from a training data-set D. To find F_D, it is necessary to determine the optimum W_N and B_N matrices by solving the following equation:

$$w_{jk}^{l}(t+1) = w_{jk}(t) - \alpha \frac{\partial F_D}{\partial w_{jk}^{l}(t)} , \tag{6.4}$$

$$b_k l(t+1) = b_k(t) - \alpha \frac{\partial F_D}{\partial b_k^{l}(t)} , \tag{6.5}$$

with $w_{jk}^{l}(t)$ and $b_j^{l}(t)$ respectively the weight and bias connecting node k in layer $l-1$ to node j in layer l during the training step t and α the learning rate. The partial derivative $\frac{\partial F_D}{\partial w_{jk}^{l}(t)}$ and $\frac{\partial F_D}{\partial b_k^{l}(t)}$ are symptomatic of the gradient descent technique used in backpropagation error method. The hyperparameter α affects the speed of gradient descent and $\alpha > 0$ ensures the decrease of the partial derivative.

For clarity, the following demonstration is supposed to be achieved for a training step t. To find the optimal weight and bias, $\frac{\partial F_D}{\partial w_{jk}}$ and $\frac{\partial F_D}{\partial b_k}$ is decomposed as the multiplication of three partial derivatives such as:

$$\frac{\partial F_D}{\partial w_{jk}^{l}} = \frac{\partial h_k^{l-1}}{\partial w_{jk}^{l}} \frac{\partial x_k^{l}}{\partial h_k^{l-1}} \frac{\partial F_D}{\partial x_k^{l}} \quad , \tag{6.6}$$

$$\frac{\partial F_D}{\partial b_k^{l}} = \frac{\partial h_k^{l-1}}{\partial b_k^{l}} \frac{\partial x_k^{l}}{\partial h_k^{l-1}} \frac{\partial F_D}{\partial x_k^{l}} \ . \tag{6.7}$$

According to equations 6.1 and 6.2 we can deduce the following equations:

$$\frac{\partial x_k^{l}}{\partial h_k^{l-1}} = \sum_{k=1}^{n} w_{jk}^{l}, \tag{6.8}$$

$$\frac{\partial h_k^{l-1}}{\partial w_{jk}^{l}} = \frac{\partial}{\partial w_{jk}^{l}} \left(\sigma(w_{jk}^{l-1} x_k^{l-1} + b_k^{l-1}) \right), \tag{6.9}$$

$$\frac{\partial h_k^{l-1}}{\partial b_k^{l}} = \frac{\partial}{\partial b_k^{l}} \left(\sigma(w_{jk}^{l-1} x_k^{l-1} + b_k^{l-1}) \right) \ . \tag{6.10}$$

Equations (6.9) and (6.10) depend on the activation function chosen during the training. As an example $\frac{\partial h_k^{l-1}}{\partial w_{jk}} = \sigma(x_j^{l})\sigma(1 - x_j^{l})$ for a sigmoid activation function σ or $\frac{\partial h_k^{l-1}}{\partial w_{jk}} = (1 - tan(x_j^{l})^2)$ for an hyperbolic tangent activation function tan.

The partial derivative $\frac{\partial F_D}{\partial x_k^{l}}$ is the error function between the computed impulse x_k^{l} for a layer l and the correct impulse y_k^{l}. We can infer this error using a Mean Square Error (MSE) function or a Mean Absolute Error (MAE) formulated as follows:

$$\text{MSE} \quad : \quad \frac{\partial F_D}{\partial x_k^{l}} = \frac{1}{2}(x_k^{l} - y_k^{l})^2, \tag{6.11}$$

$$\text{MAE} \quad : \quad \frac{\partial F_X}{\partial x_k^l} = \frac{1}{2}|x_k^l - y_k^l|. \tag{6.12}$$

The error function has an impact on the training of the neural network and is therefore one of the parameters to be identified for the training. In particular, MSE is more sensitive than MAE to large errors.

For a specific activation function and error function, and thanks to the combinations of the previous equations, a new estimation of the weight and the bias can be obtained, associated with the neurons from layers l and $l-1$. Nevertheless, the backpropagation training seeks to estimate new weights and biases for every layer of the MLP. Accordingly, the neural network is supplied with a ground truth impulse at its output allowing the computation of new weights and biases at layer N only. These new weights and biases supply the network with the correct impulse for the neurons of the layer $N-1$. By iteration, all the weights and biases from every layer are then modified, starting with the output layer and finishing with the input layer. This backward iteration process, associated with the gradient descent algorithm gives this method the name of backpropagation.

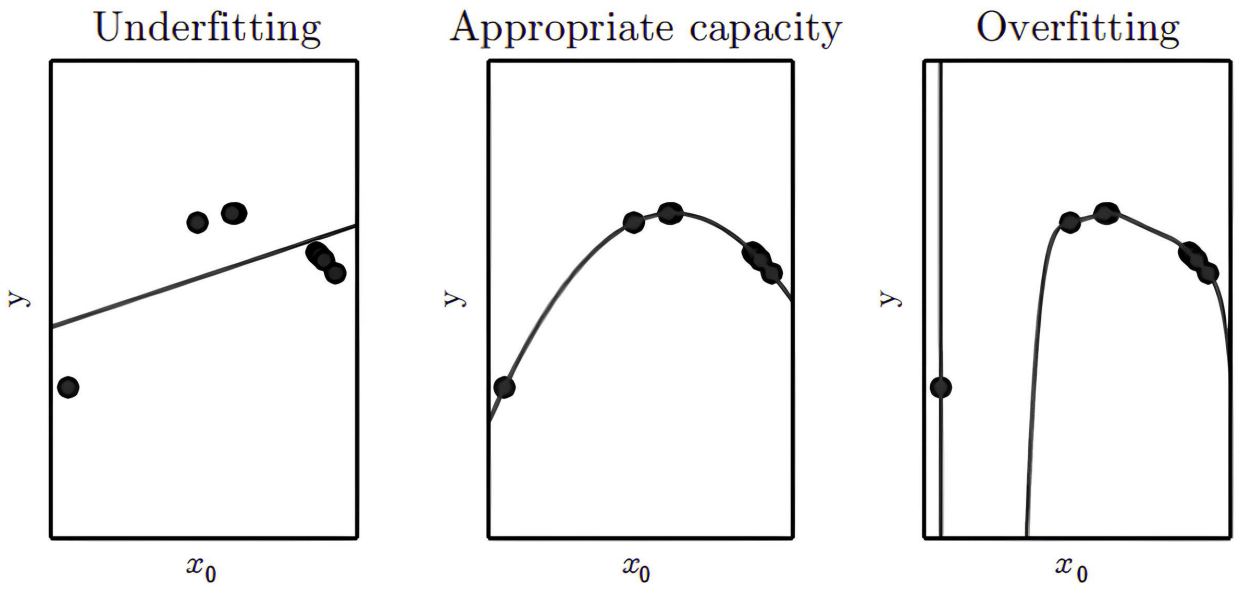


Figure 6.5: *Illustration of overfitting. The left illustration depicts an underfitting, referring to the situation in which a model does not perform well on the data it was trained on and is also unable to generalize its results well. The right illustration depicts an overfitting as the curve matches the data but is unable to generalize its results well. The middle illustration depicts the optimal curve, striking a compromise between underfitting and overfitting, maximizing both generalization and prediction performance. Adapted from [Aliyu, Mokhtar, and Hussin 2022]*

The different steps of MLPs can be summarised as follows:

- Step 0 - After setting the number of layers and neurons per layer of the MLP, the weights and biases are initialized.

- Step 1 - The network is fed with data stemming from the data-set associated with the target problem. Iteratively, samples of the dataset are inputted to the network creating impulse crossing every layer of the network until reaching the output layer and providing an answer in the form of a vector storing parameters estimations, this is the prediction. This step characterizes the feed-forward aspect of MLP networks.

- Step 2 - The prediction is compared to the correct answer, supplied by the training data set and allows the backpropagation method to tune the weights and biases of the network to match the correct answer.

- Step 3 - Steps 1 and 2 are repeated with different samples from the data set until reaching a total training step count. A training step is called an ”epoch”.

- Step 4 - After reaching the desired epoch count, the training is stopped. To estimate the generalization performance of the neural network, one needs to test it on independent data, which have not been used to train the network. This independent data-set is called the testing data-set and aims to prevent the neural network from overfitting. Overfitting occurs when the network tries to predict closely or exactly the training data, thus failing to fit additional data or predict future observations reliably [figure 6.5]. The error stemming from the prediction of the testing data is called accuracy and informs on the capacity of the model to correctly and robustly fit the target function.

Concerning step 0, different initialization strategies exist and are crucial to improve the algorithm's learning capabilities as it can determine whether the algorithm converges at all, with some initial points being so unstable that the algorithm encounters numerical difficulties and fails altogether [Goodfellow, Bengio, and Courville 2016a].

MLP design comes with the tuning of numerous parameters such as the number of layers, number of neurons, initialization of weight and biases, learning rate, epoch count, activation and error function. Tuning of the parameters is not straightforward and often comes through iterations over empirical experiments to capture their impact on the convergence of the neural network. The following section will describe in detail the choices made to tune our MLP network described to the decoding of WM microstructure.

6.3 MEDUSA neural network implementation

The MEDUSA simulation campaign performed on the TGCC HPC facility resulted in the synthesis of 42 673 samples augmented by the noise and rotation algorithm for a total of 4 267 300 samples stored in labeled CSV tables. Each sample is characterized by 285 diffusion parameters corresponding to 4 DTI parameters (ADC, FA, $\lambda_{\parallel}$, $\lambda_{\perp}$) and 15 coefficients of the spherical harmonics decomposition of the signal acquired on each shell of the 15 iCortex dMRI sequences. For each sample, a set of 7, 13, or 19 generative microstructural parameters is recorded according to the number of fiber populations created in the sample and corresponding respectively to 1, 2, or 3 populations. A neural network was designed considering the set of dMRI (DTI and SH) parameters as the inputs of the neural network and the generative microstructural parameters as the output parameters. The following sections present the different concerns surrounding this design.

6.3.1 General design of the decoding pipeline

The neural network seeks to decode the WM microstructure of samples according to their dMRI signature. The first approach developed was to store all the simulated samples together, without distinguishing the number of populations within the sample. It resulted in the creation of a unique CSV table with columns describing generative parameters some of them being unfilled since one fiber population sample is described by only 7 geometric parameters whereas two or three fiber populations have respectively 13 and 19 generative parameters. These unfilled parameters were associated with a ”null” value and led to poor training of the neural network as it was unable to interpret the ”null” value correctly.

The sampling strategy was then improved to propose 4 different neural networks rather than one to overcome the issue of the unfilled column. The first neural network was fed with all the simulated samples and only aimed to infer the number of fiber populations within the samples. The task of this first network resembles a classification problem but it remains a regression as the answer might drift from whole numbers like 1,2 or 3 to decimal numbers, illustrating the uncertainty of the prediction. The following three networks are respectively dedicated to predicting the microstructural parameters in the case of one, two, and three fiber populations, avoiding thus the problem of unfilled parameter values. The arrangement of the data-set parameters according to the neural network to be trained is presented [figure 6.6].

The network dedicated to the prediction of the number of fiber populations is placed upstream of the decoding pipeline in a way to guide the prediction according to the number of populations encountered within the sample, as illustrated in [figure 6.7]. The decoding pipeline and its neural networks have been coded using the PyTorch library adapted for C++ API [Pytorch 2024].

6.3.2 Tuning of the NEDUSA white matter neural network

Different strategies exist to find the optimal configuration of an MLP neural network [Bronlee 2019], which are listed below:

- Experimentation - Experimentation consists of trying several settings until reaching a satisfying convergence of the training; it is a common strategy since the tackled problem is likely to be unique as well as the stemming data-set.

Data-set for population count decoding network						
Samples ID	**Generative parameters** (1 in total)	**Diffusion parameters** (285 in total)				
	Population count	Sequence 1 -> Sequence 15				
		Diffusion tensor imaging parameters				Spherical harmonics coefficient
		Apparent diffusion coefficient	Fractional anisotropy	Radial diffusivity	Longitudinal diffusivity	Coefficient 0 -> coefficient 14

Data-set for 1 2 or 3 total fiber population generative parameters decoding network										
Samples ID	**Generative parameters** (6, 12 or 18 in total)						**Diffusion parameters** (285 in total)			
	1 total fiber population number 1 -> 3 total fiber population number 1						Sequence 1 -> Sequence 15			
							Diffusion tensor imaging parameters			Spherical harmonics coefficient
	Axon diameter	Global angular dispersion	Volume fraction	Mean orientation X	Mean orientation Y	Mean orientation Z	Apparent diffusion coefficient	Fractional anisotropy	Radial diffusivity	Longitudinal diffusivity	Coefficient 0 -> coefficient 14

Figure 6.6: *Layout of parameters for different neural networks. The top table presents the parameters required for training the neural network dedicated to population count decoding whereas the bottom table presents the parameters required for training the neural network dedicated to decoding the microstructure of samples filled with respectively 1, 2, or 3 fiber populations.*

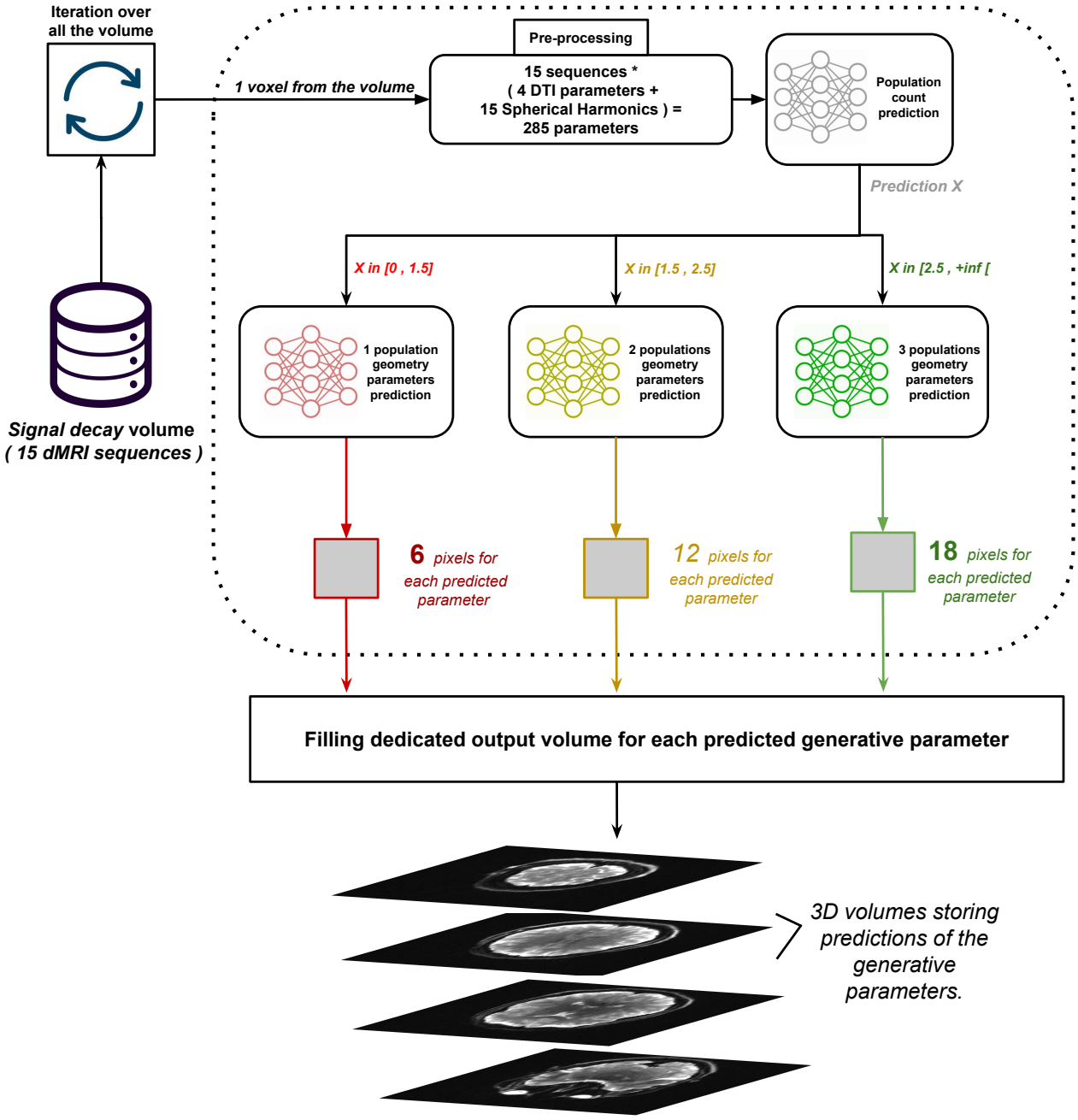


Figure 6.7: *Illustration of the overall decoding pipeline: the 285 diffusions parameters from one voxel are computed by the first neural network aiming to predict the number of populations within the sample. According to the result, the data are then transferred to the network dedicated to this population count. A set of geometry parameters is then predicted to fill an output prediction volume. The process is iteratively achieved until decoding the full volume. A total of 36 maps are predicted by the decoding pipeline, as a sum of the number of parameters predicted by the 3 networks dedicated to decoding 1, 2, and 3 populations samples geometrical parameter.*

- Intuition - Despite being based on a ”black box” logic, deep learning seems to have its own logic, of which we can have an intuition after several experiments. Intuition can sometimes be useful for the experienced neural network designer to improve learning convergence.
- Inspiration - Although each neural network is unique, it is still possible to classify them into certain categories making the numerous neural networks developed over the years potential inspirations to the neural network being designed.

The network designed here is the result of a combination of these three strategies and resulted in the setting of a neural network whose various parameters are described hereafter:

6.3.2.1 Number of layers

Increasing the number of hidden layers might improve the accuracy depending on the complexity of the problem [Goodfellow, Bengio, and Courville 2016b]. Nevertheless, it may also cause the accuracy estimated from any test data-set to decrease whereas the network overfits the training data-set as illustrated in [figure 6.5]. In our model, a balance was found resulting in the design of an MLP network including 11 layers.

6.3.2.2 Number of neurons per layer

Similar to the strategy used to tune the number of layers, it may be useful to increase the number of neurons per hidden layer to make the MLP more suitable for complex problems. Input and output layers have a fixed number of neurons equal to the input and output size of the data-set considered for the training. In hidden layers, the number of neurons per layer decreases from the first post-input layer to the last pre-output layer starting from 4096 neurons to 16 neurons following decreasing powers of 2 for computational optimization purposes. This decrease in the number of neurons was motivated by the will to reduce the dimensionality of the problem by projecting the data onto a lower dimensional subspace that captures its essence.

Increasing the number of layers and the number of neurons per layer implies some increased computational resources required to train the network and more memory storage to structure the MLP weights and biases.

6.3.2.3 Activation function

The choice of activation functions is crucial in the design of the MLP, as they act as buttons adjusting the network parameters. A different activation function can be set up for each layer of the network, but we have chosen to keep a ReLU activation function for all layers. ReLU activation function standing for "rectified linear unit" was computed for our model, and is defined as the positive part of its argument such as:

$$\sigma(x) = \max(0, x), \tag{6.13}$$

with σ the ReLU activation function. Several activation functions were tested, such as the Gelu (Gaussian Error Linear unit) function used in [Fang et al. 2023], but ReLU has shown satisfactory results and has the advantage of not requiring any heavy processing. [Figure 6.8] presents a plot of the ReLU function.

6.3.2.4 Optimizer

The Optimizer is the method in charge of updating the weights, biases, and learning rate during the learning stage. Various optimizers exist and the ADAM (adaptive moment estimation) optimizer was selected for our model because it is computationally efficient, has little memory requirement, and is well suited for problems that are large in terms of data/parameters [Kingma and Ba 2014]. The ADAM optimizer comes with the tuning of the learning rate, set in our model to 10^{-3}, and beta

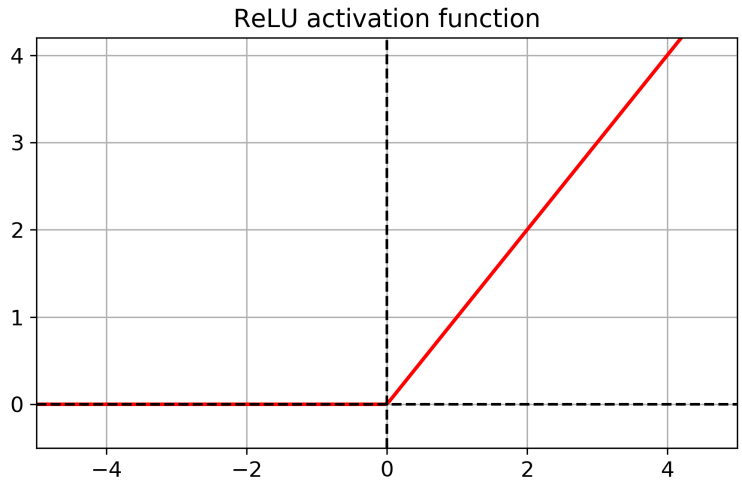


Figure 6.8: *Rectified linear unit function plot.*

parameters β_1 and β_2 respectively fixed to 0.9 and 0.999 according to [Keras 2024]. Betas control the moving average of the gradient at stake in the application of ADAM's stochastic gradient descent [Alabdullatef 2020]. One of the advantages of the ADAM optimizer lies in its ability to adapt the learning rate during the learning process, thus accelerating the convergence of the learning.

6.3.2.5 Batch size

The batch size is a further hyperparameter that defines the number of samples the network works through before updating the parameters using backpropagation. Within a batch, samples are drawn randomly from the database which allows smoothing of the loss function curve providing an adequate representation of the model learning. Indeed, batches make the model adapt to a larger number of samples, thus eliminating the potential specificity of each sample during training. The batch size, therefore, acts as an average of the samples, and for our model, we set it to 20 000 samples, which represent approximately 2% of the total samples composing the training data-sets.

6.3.2.6 Dropout

Dropout refers to removing forward and backward random connections of neurons within the network resulting in the reduction of potential overfitting [towardsdatascience 2024]. The neurons are dropped out by a probability set in our model to 10%, following the recommendations of the state of the art and after experimenting with different values of dropout probability. Indeed, empirically we observed that a dropout probability set too high can lead the model to be unable to learn.

6.3.2.7 Initialization

During backpropagation, the gradients propagate back through the network layers and decrease significantly. As a result, the weights associated with the first layers are poorly updated at each iteration of the optimization process [Gfg 2023c]. This phenomenon is called the vanishing gradient problem and can be tackled by the Kaiming initialization [K. He et al. 2015] technique implemented in our model that facilitates efficient training of the parameters from the first layers [Gfg 2023b].

6.3.2.8 Data normalization

Successful models are built on the quality of the data they are trained on. To enhance this quality, some data preprocessing is required such as normalization which is necessary because features from the data don't contribute equally to the final prediction as they differ in range and unit. Normalization ensures uniformity of the numerical magnitudes of features. Within the pipeline dedicated to neural networks, we have implemented a data normalization pre-processing procedure before training the neural networks. This command allows each parameter in the database to be normalized according to its distribution, without the need to save the normalized version locally. Various normalization methods are available, such as rescaling, logarithm or z-score [Patel 2022]. Each of these normalization methods requires information about the mean and standard deviation, which can be estimated using distribution histograms of the parameters (see **??**). Our model benefited from a rescaling normalization of the generative parameters, ranging them in [0,1] while the population count and the diffusion parameters were not normalized.

6.3.2.9 Batch normalization

As input data, the impulses stemming from the network's hidden layers can be normalized. This technique is called "batch normalization" as it's computed with the mean and standard deviation of the parameter distribution within the batch under consideration. According to [Ioffe and Szegedy 2015], batch normalization is good for preventing the "internal covariate shift" phenomenon, defined as a change in the distribution of network activation functions due to the change in network parameters during the training stage. These small changes in input distribution add up fast and amplify greatly deeper into the network. Ultimately, the input distribution received by the deepest neurons changes greatly between every epoch. As a result, these neurons need to continuously adapt to the changing input distribution, meaning that their learning capabilities are severely bottlenecked. Furthermore, batch normalization is said to be good against the vanishing gradient and overfitting problems [Vinod 2020; Ioffe and Szegedy 2015], which may add up with dropout and initialization techniques. The effects of batch normalization on neural network learning are still debated, as shown by [Santurkar et al. 2018] stating that batch normalization improvement in learning is more due to the predictive and stable behavior it brings to gradients rather than reducing the internal covariate shift.

The previous settings were set to the 4 different networks of our pipeline, indiscriminately to the complexity of the neural network task correlated to the number of parameters to be decoded. In the future, an adaptation of the different network settings is desirable to adapt the design of neural networks for each task.

6.3.3 Training and testing the networks

Each of the 4 networks constituting the decoding model beneficiated from its own training with its dedicated data-set. Networks learning was divided into two steps: the training and the testing:

- The different neural networks dedicated to the prediction of the number of fiber populations and to the prediction of the generative parameter for the 3 cases of 1, 2, 3 populations were trained using 772 000, 962 000, 1 496 700 and 1 437 300 samples respectively.
- The different neural networks dedicated to the prediction of the number of fiber populations and to the prediction of the generative parameter for the 3 cases of 1, 2, 3 populations were

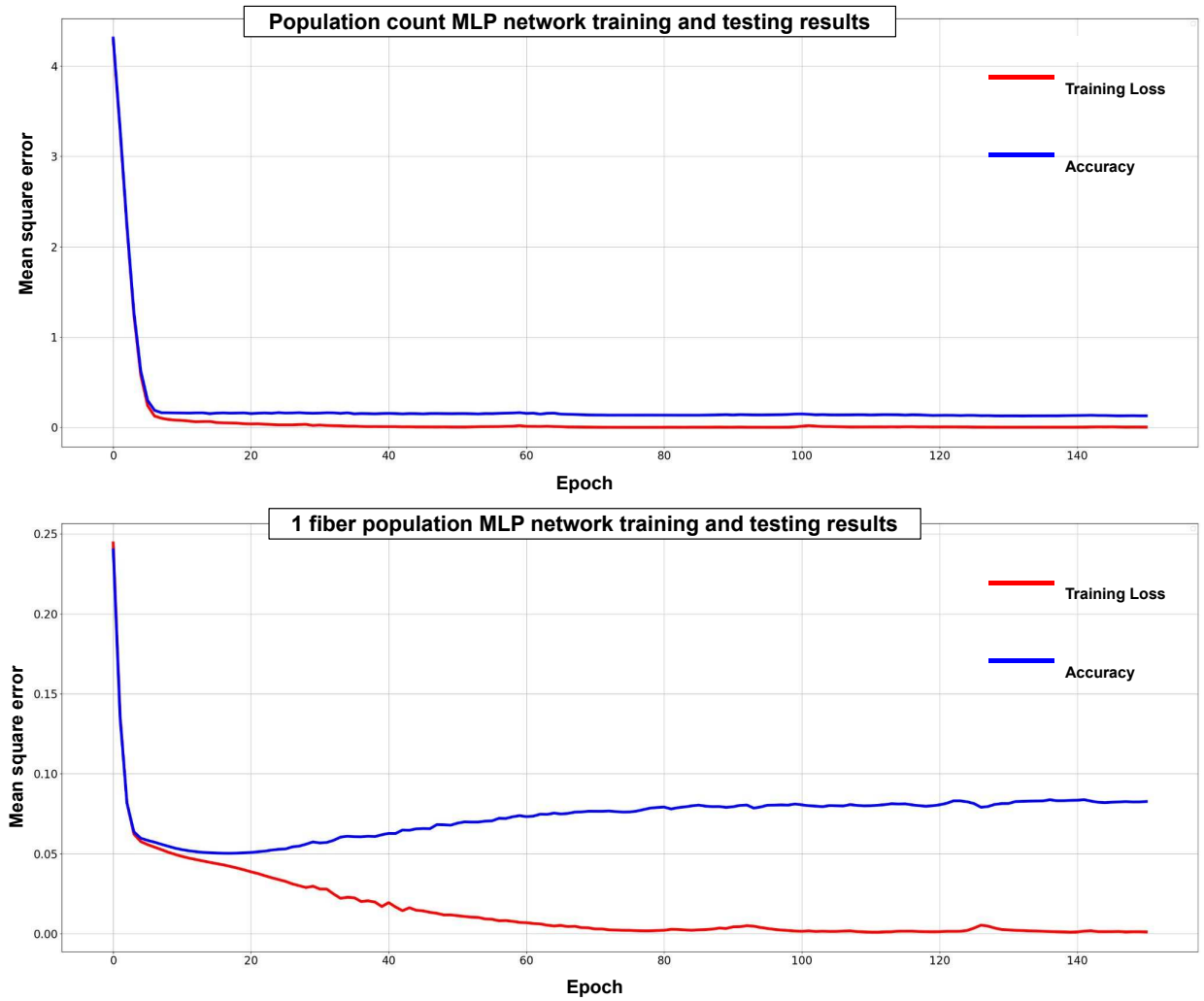


Figure 6.9: *Evaluation of the mean square error of the population count neural network predictor (top) and of the microstructure neural network predictor for 1 fiber population according to the epoch count.*

tested using 81 480, 102 700, 139 400, and 165 200 samples respectively.

Gaussian noise of approximately 10% was added corresponding to the typical noise level observed on acquisitions performed with a 3T Prisma MRI system. It would be interesting in the future to vary the noise standard deviation to better evaluate the robustness of our neural networks to noise.

[Figures 6.9, 6.10] show the training and testing error of the different models for each learning epoch. The training errors and accuracies presented correspond to the mean of the mean square errors of each parameter decoded by the models.

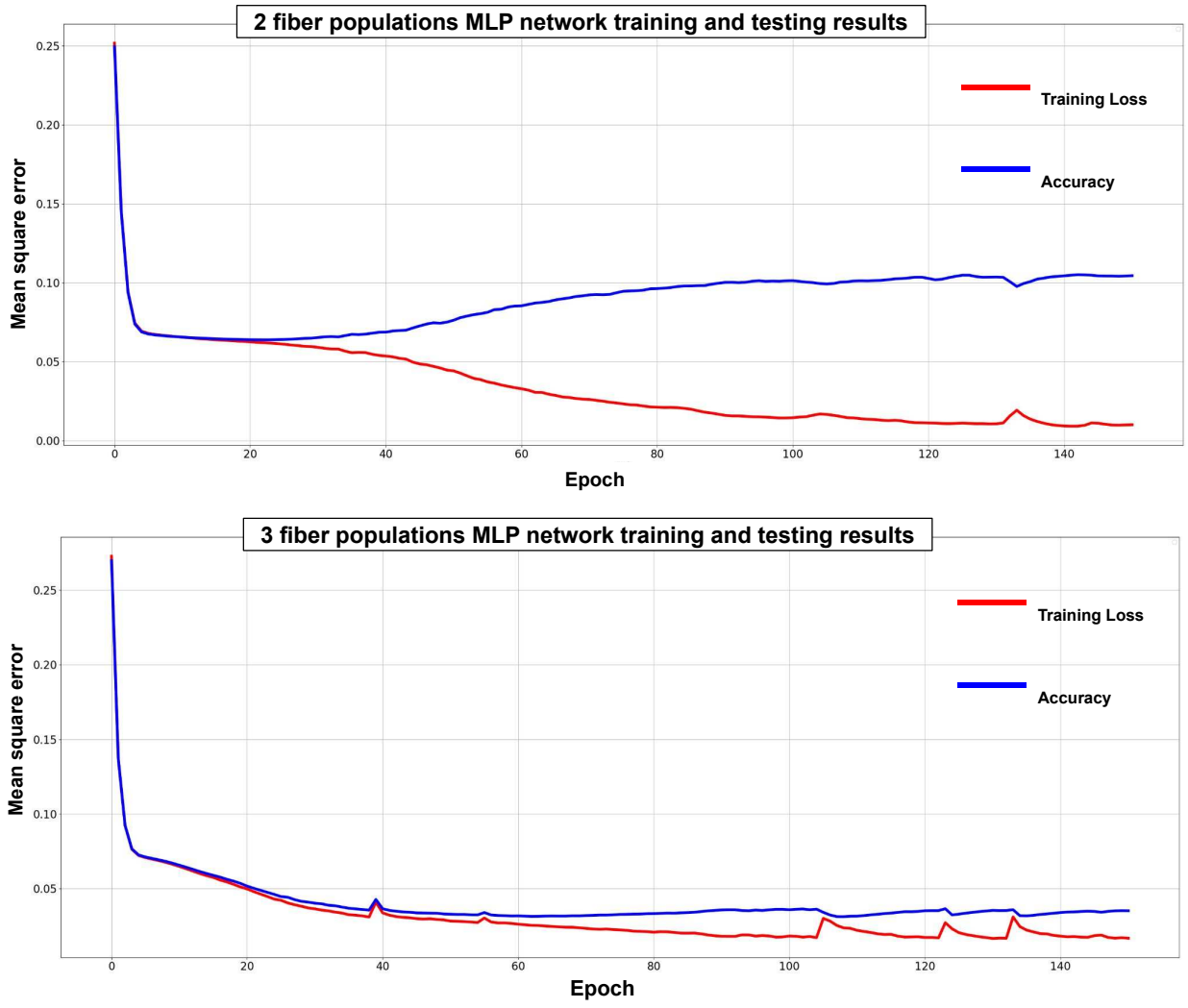


Figure 6.10: *Evaluation of the mean square error of the microstructure neural network predictor for 2 fiber population network predictor (top) and of the microstructure neural network predictor for 3 fiber population according to the epoch count.*

Neural nework		Population count	Axon diameter	Global angular dispersion	Volume fraction	Mean orientation X	Mean orientation Y	Mean orientation Z
Population count		**1.3** 10^{-1}	/	/	/	/	/	/
1 fiber population	**First population parameters**	/	**5.9** 10^{-3}	**4.0** 10^{-2}	**4.8** 10^{-3}	**1.7** 10^{-1}	**1.7** 10^{-1}	**1.1** 10^{-1}
2 fiber populations	**First population parameters**	/	**4.6** 10^{-2}	**9.7** 10^{-2}	**7.8** 10^{-3}	**1.7** 10^{-1}	**1.8** 10^{-1}	**1.2** 10^{-1}
	Second population parameters	/	**4.7** 10^{-2}	**1.1** 10^{-1}	**7.1** 10^{-3}	**1.7** 10^{-1}	**1.7** 10^{-1}	**1.2** 10^{-1}
3 fiber populations	**First population parameters**	/	**6.5** 10^{-2}	**8.7** 10^{-2}	**1.4** 10^{-2}	**1.4** 10^{-2}	**1.5** 10^{-2}	**1.2** 10^{-2}
	Second population parameters	/	**5.7** 10^{-2}	**1.2** 10^{-1}	**1.5** 10^{-2}	**1.4** 10^{-2}	**1.4** 10^{-2}	**1.2** 10^{-2}
	Third population parameters	/	**6.3** 10^{-2}	**8.7** 10^{-2}	**3.3** 10^{-3}	**1.4** 10^{-2}	**1.4** 10^{-3}	**1.4** 10^{-2}

Table 6.1: Accuracy of the predictions of the various networks after 150 epochs. The accuracies presented result from a calculation of the mean square error between the predicted values and the ground truth value of the test data set.

The range of mean square errors for the population count neural network predictor differs from the others because the population count was not normalized between [0, 1] like other parameters. The divergence of the training and testing error curves for the 1 and 2 population decoding models indicates an overfitting of the neural network after epoch 20. To complete the analysis of training results [table 6.1] shows the testing error obtained after 150 epochs for each parameter independently.

If we look more closely at the error stemming from the prediction of each parameter, we observe that the prediction of fiber orientations, whatever the number of populations, is insufficient compared to other parameters. Fiber's mean orientation direction variability explains these poor results. Indeed, fiber orientations, because they are described in Cartesian coordinates, retain information about their directions, yet vector direction has no impact on the diffusion phenomenon, it is therefore necessary to prevent the data from this variability to improve the performance of the predictive model. Nevertheless, the fact that the predictive models are also insensitive to the direction of the vectors in their prediction is surely proof of good learning of the diffusion phenomenon by the model, and therefore a first promising validation of the quality of the simulated diffusion-weighted signals provided.

A post-processing code was added to the training data. This code aims to force the alignment of fiber orientation vectors to an arbitrarily defined direction oriented along the upper half of the angular sphere of the orthonormal reference frame (X,Y,Z). To achieve this result, the coordinates (x,y,z) of each fiber population orientation vector were iteratively read to identify the orientations for which $z < 0$, in which case a scalar product (x,y,z)*(-1,-1,-1) was applied resulting in the standardization of all fiber orientation vector direction toward angular sphere upper half. [Figure 6.11] illustrates the application of this post-processing code in the case of a sample with two fiber populations.

Based on the rotated data-set, networks were trained once again, except for the one dedicated to decoding population numbers as this does not involve fiber orientation. [Figure 6.12] shows the evolution of the training and testing error networks decoding the microstructure of samples with 1, 2, and 3 populations in the case of a database with rotated orientation vectors.

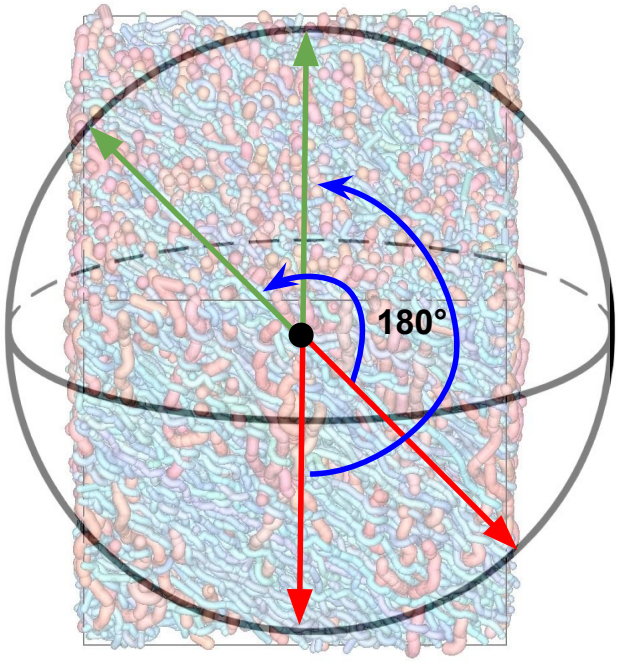


Figure 6.11: *2D Illustration of a two fiber populations simulated sample with it means orientation vector in red re-oriented to the top part of the surrounding angular sphere using rotation angle of 180°.*

As expected, new networks have better prediction results for fiber population orientations, with for instance a reduction of the testing error by a factor of 50 in the case of one population [table 6.2]. Overall, the accuracy of the models has been improved by the rotation of the data-set as the error from each parameter prediction decreased compared to the previous networks. Furthermore, the overfitting observed during the previous training session decreased as well, despite that, the models corresponding to 2 and 3 population decoding were trained again with fewer epochs to minimize the remaining overfitting, resulting in stopping the training after epochs 31 and 41 respectively for networks of 2 and 3 populations decoding.

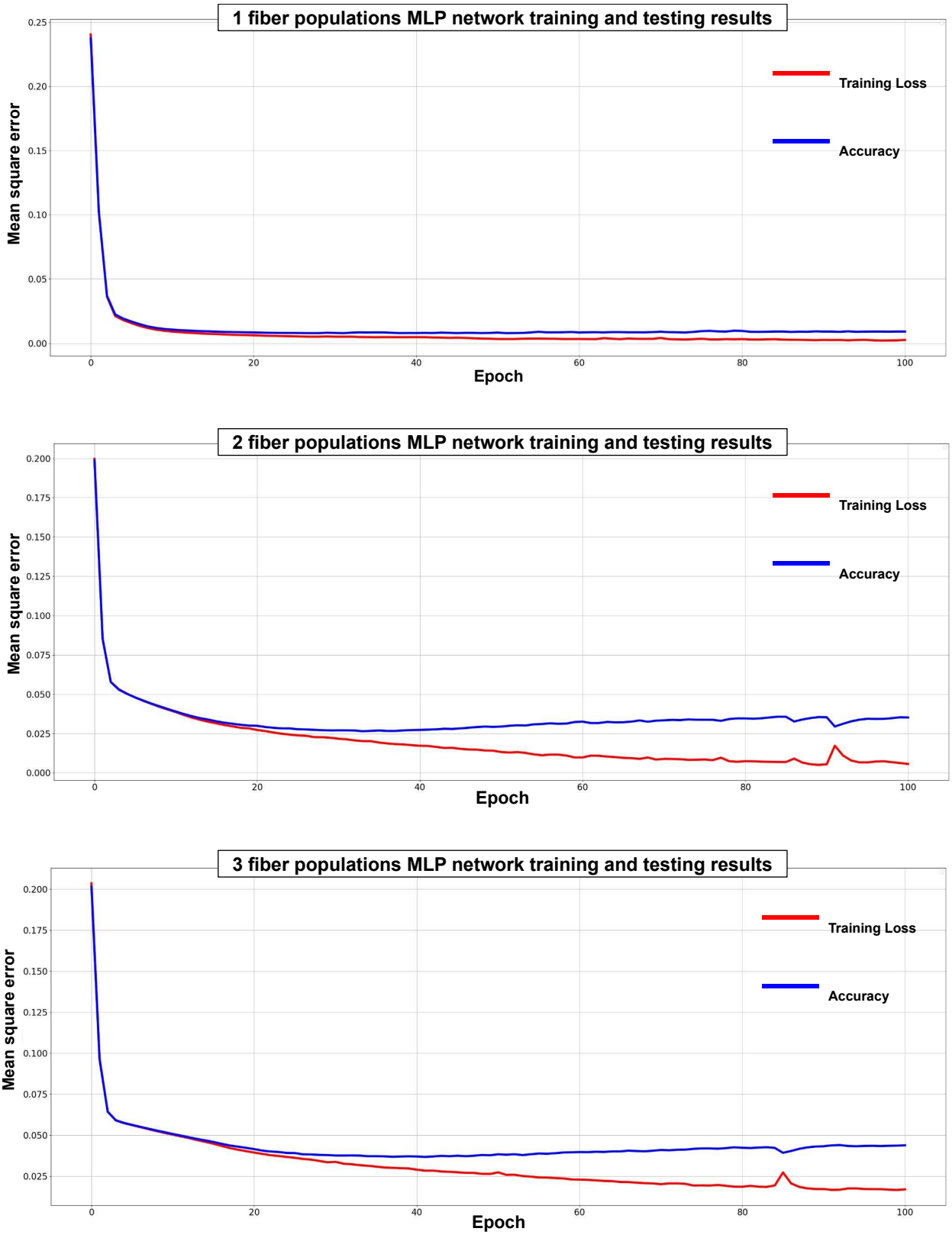


Figure 6.12: *Evaluation of the mean square error of the microstructure neural network predictor for 1, 2, and 3 fiber population network predictor after applying the rotation post-processing algorithm.*

Neural nework		Population count	Axon diameter	Global angular dispersion	Volume fraction	Mean orientation X	Mean orientation Y	Mean orientation Z
Population count		**1.7** 10^{-3}	/	/	/	/	/	/
1 fiber population	**First population parameters**	/	**4.1** 10^{-3}	**3.3** 10^{-2}	**3.2** 10^{-3}	**6.4** 10^{-3}	**7.2** 10^{-3}	**5.0** 10^{-3}
2 fiber populations	**First population parameters**	/	**4.2** 10^{-2}	**6.0** 10^{-2}	**1.4** 10^{-2}	**1.4** 10^{-2}	**1.5** 10^{-2}	**3.6** 10^{-3}
	Second population parameters	/	**4.2** 10^{-2}	**7.8** 10^{-2}	**1.3** 10^{-2}	**2.5** 10^{-2}	**2.5** 10^{-2}	**4.9** 10^{-3}
3 fiber populations	**First population parameters**	/	**5.8** 10^{-2}	**7.4** 10^{-2}	**1.8** 10^{-2}	**2.1** 10^{-2}	**2.2** 10^{-2}	**7.1** 10^{-3}
	Second population parameters	/	**5.9** 10^{-2}	**8.6** 10^{-2}	**2.0** 10^{-2}	**3.6** 10^{-2}	**3.6** 10^{-2}	**8.4** 10^{-3}
	Third population parameters	/	**6.0** 10^{-2}	**8.6** 10^{-2}	**1.1** 10^{-2}	**4.5** 10^{-2}	**4.6** 10^{-2}	**1.4** 10^{-2}

Table 6.2: Accuracy of the predictions of the various networks after 150 epochs. The accuracies presented result from a calculation of the mean square error between the predicted values and the ground truth value of the test data-set, after applying the rotation post-processing algorithm to the data-set.

[Figures 6.13, 6.14, 6.15] give a representation of the prediction versus ground-truth accuracy for the 1, 2, and 3 fiber populations computational models. These figures also feature the model's predictive results dedicated to decoding the number of fiber populations in the case of samples composed respectively of 1, 2, and 3 fiber populations exclusively.

In those figures, the axon diameter (AD), the global angular dispersion (GAD), and the volume fraction (VF) are distributed in lines perpendicular to the ground truth axis, unlike the mean orientations coordinates (x,y,z). This difference is due to the rotation augmentation algorithm applied to the dataset before the learning step that doesn't impact the AD, GAD, and VF, resulting in an oversampling of these parameters proportional to the rotation count factor. Nevertheless, this line structure makes it possible to identify model predictions' mean and standard deviation for a given ground truth.

Let's take a closer look at the results shown in [figure 6.13], dedicated to 1 fiber population MLP:

- The population count prediction is centered on the value 1.5. In our model, because of rounding, any prediction between]-inf, 1.5] is associated with samples composed of 1 fiber population. This limits the model's prediction error, although some values still lie outside this range. The same goes for the 2 and 3 fiber population count decoding, with variation ranges of [1.5,2.5] and [2.5,+inf[respectively.
- Axonal diameter prediction obtains a coefficient of determination of 0.95. It is a promising result, but there is a deterioration in the standard deviation of prediction as a function of axonal diameter. This observation counters our expectations, since in dMRI, large-diffusion elements have a greater impact on the signal than small ones, so it's the opposite trend we were expecting. Two elements can explain this anomaly: first, it could be attributed to network learning, which reduces its overall error by increasing the standard deviation of large axonal diameters. The second explanation could come from the dMRI sequences selected and derived from the iCORTEX protocol, to decode axonal diameters, as the iCORTEX protocol only proposes a single diffusion time for all sequences, or we know that this parameter is crucial if we want to correctly predict axonal diameter [Alexander 2010].
- Global angular dispersion prediction obtains a coefficient of determination of 0.62. This is the least convincing score among the predicted parameters. This can be explained by the more limited role of GAD on the dMRI signal.
- Volume fraction obtains a coefficient of determination of 0.94. The standard deviation of the predictions is narrowed and the accuracy is evenly distributed along the range [0.5, 0.9].
- Coordinates of mean fiber orientation X, Y, and Z obtain respectively 0.91, 0.92, and 0.97 scores. The prediction accuracy is evenly distributed along the range [-1.0, 1.0] for X and Y, and along [0.0, 1.0] for Z, this distinction results from the rotation of orientations along the upper half of the angular sphere of the orthonormal reference frame (X,Y,Z), making all z coordinates positive.

Results attributable to models decoding samples with 2 and 3 populations show similar trends to those observed for one population, with a few distinctions:

- The greater the number of fibers, the lower the accuracy. In particular, GAD predictions are considered null in the case of two and three populations with determination coefficients of 0.19 and 0.02.

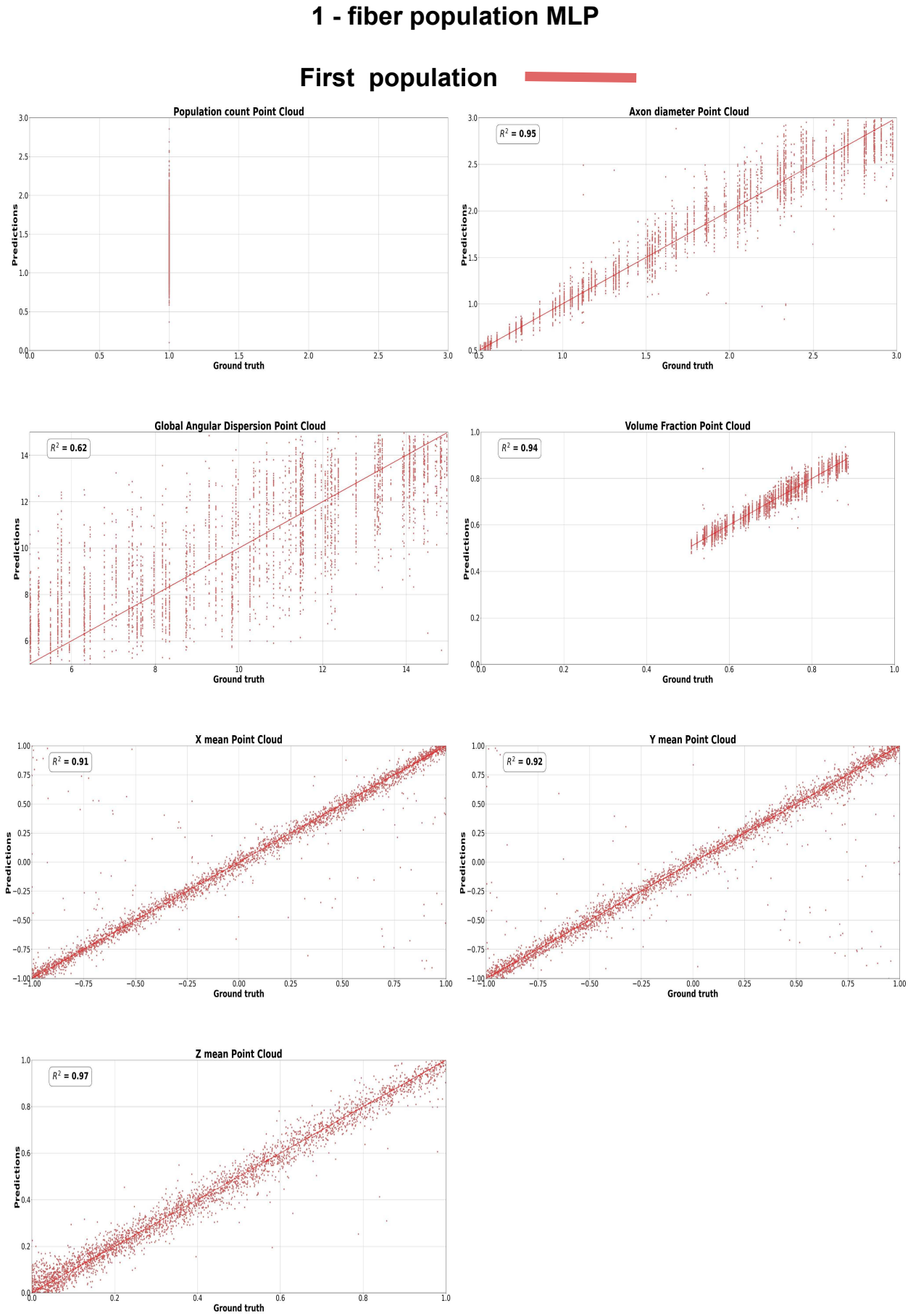


Figure 6.13: *Point cloud plots presenting the prediction capacities of the network dedicated to decoding 1 fiber population microstructure. The first plot exhibits the prediction results from the population count network. The point cloud is formalized as a color map to highlight the precision results from the prediction of each parameter. A determination coefficient R^2 is displayed for each predicted parameter.*

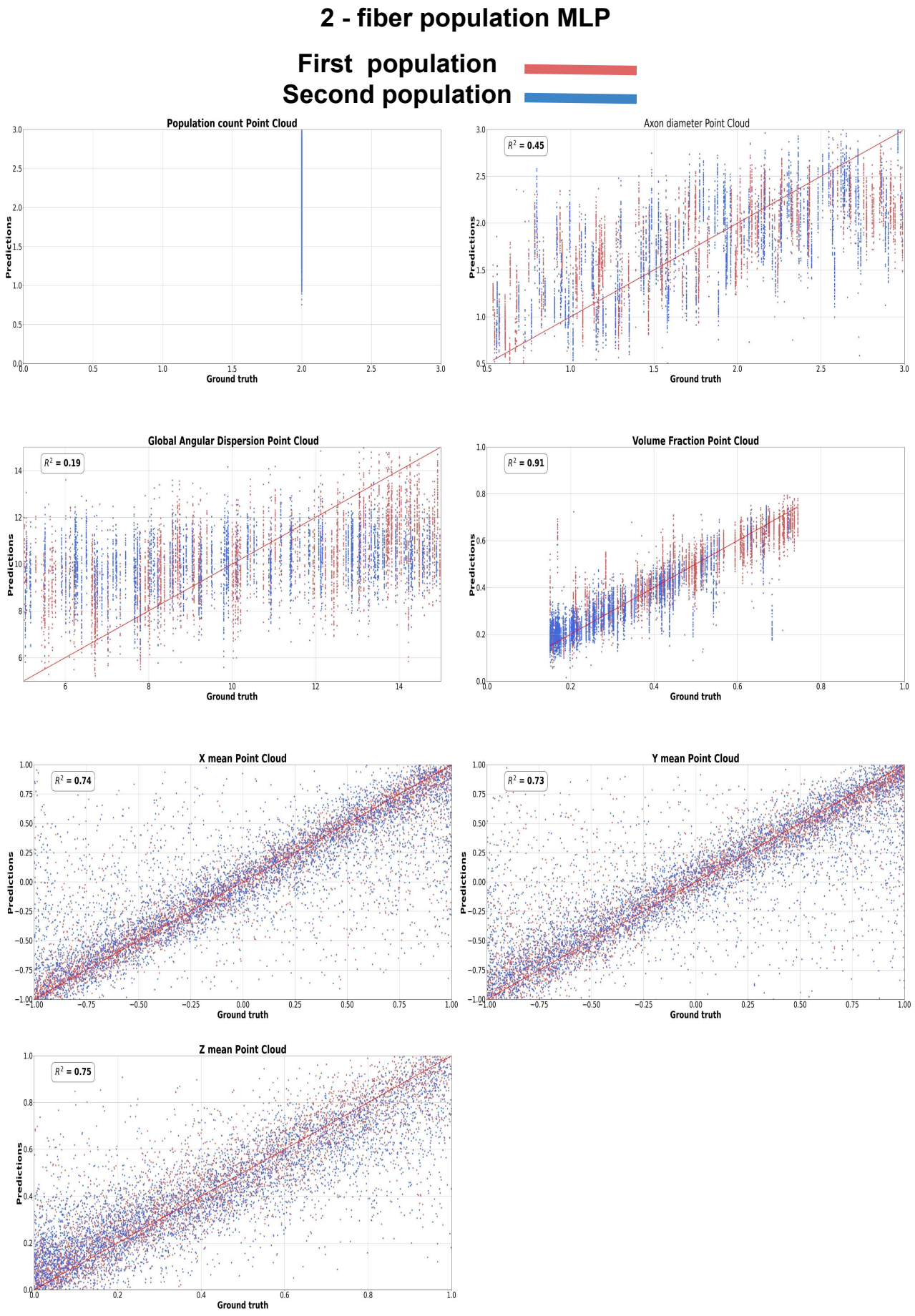


Figure 6.14: *Point cloud plots presenting the prediction capacities of the network dedicated to decoding 2 fiber population microstructure. The predicted parameters are separated by color depending on the fiber population they belong to within the sample. A determination coefficient R^2 is displayed for each predicted parameter.*

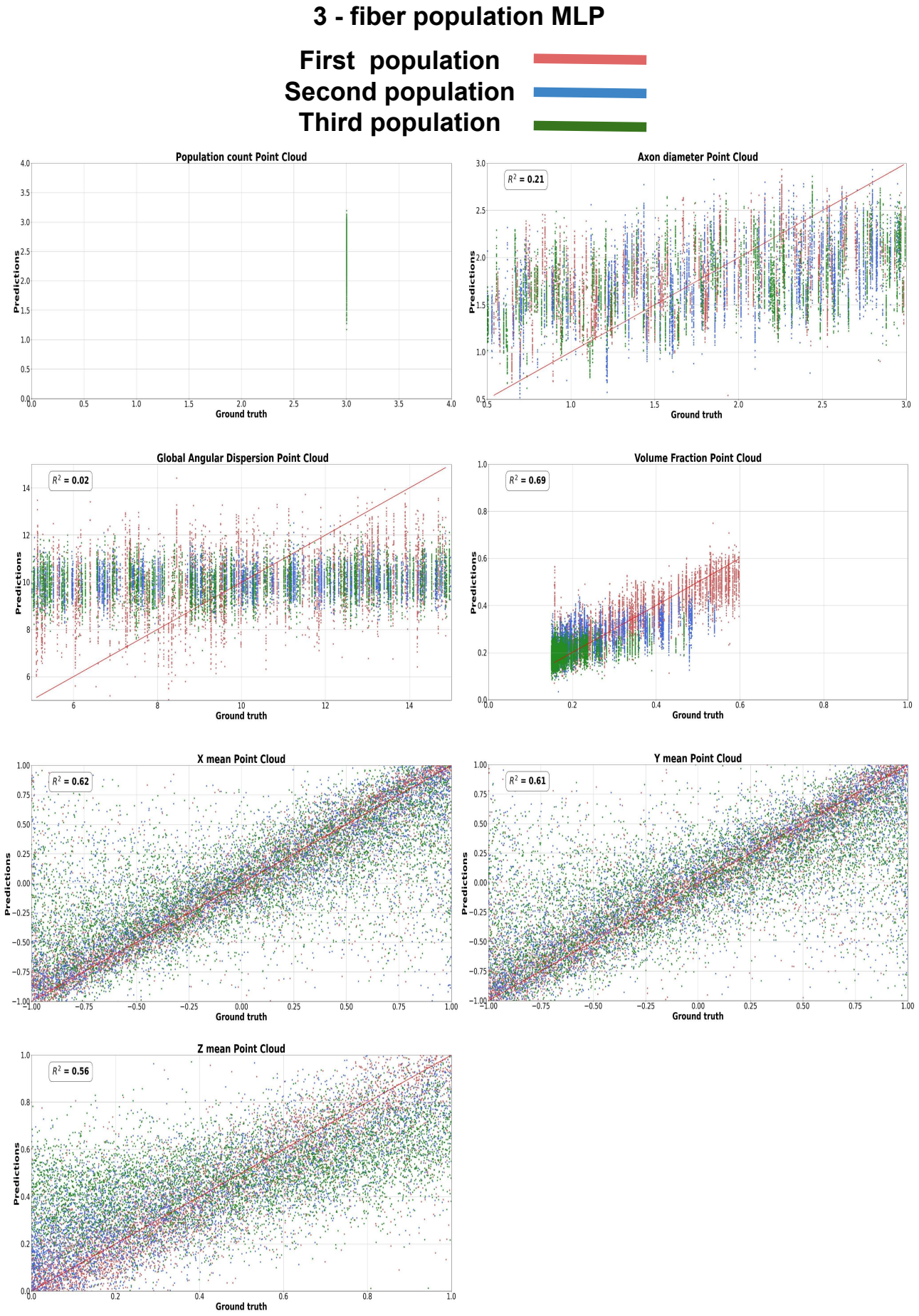


Figure 6.15: *Point cloud plots presenting the prediction capacities of the network dedicated to decoding 3 fiber population microstructure. The predicted parameters are separated by color depending on the fiber population they belong to within the sample. A determination coefficient R^2 is displayed for each predicted parameter.*

- Figures showing the distribution of volume fractions reveal an uneven distribution. This is due to the arbitrary choice made during the simulations to distribute the VFs of successive fiber populations according to the remaining volume, as the total VF from a sample cannot exceed 1.0. As a result, populations placed further downstream during generation are less likely to be able to obtain a large volume fraction.

- The belonging of a parameter to a given population does not seem to influence the prediction accuracy from a trend point of view. However, from a qualitative point of view, the parameters associated with fiber populations labeled 2 are less well predicted than those associated with fiber populations labeled 1, the same goes for fiber populations labeled 3 compared to 2 and 1. This is explainable by the distributions of VFs previously discussed. Indeed, if the volume fraction is low, so is the impact of the generative parameter on the overall geometry of the sample, and therefore the impact on the dMRI signal and hence on the ability to be decoded by the networks.

The prediction of the microstructure parameters in the case of one population is very satisfying with a high determination coefficient R^2 value for most of the generative parameters prediction. The number of populations in the sample has an impact on the learning quality of the model as predictions are much more accurate for 1 population than for 2 or 3 populations. Among the predicted parameters, the mean orientation of the fiber populations is the most reliable followed by their volume fractions, the axonal diameter, and finally the angular dispersion.

[Figures 6.16] present the feature importances from model prediction dedicated to 1 fiber population decoding. The importance value is computed using a permutation technique that measures the contribution of each feature by calculating the mean square error difference between the original prediction and the one after shuffling the model features [ScikitLearn 2024]. The features designate the 285 dMRI parameters used to train the decoding methods, however, the figures present only the FA and spherical harmonics (SH) stemming from the 15 iCORTEX PGSE sequences, and the 15 SHs of each sequence were summed to form a unique SH feature. The ADC, the axial and transversal diffusivity ($\lambda_\perp$,$\lambda_\parallel$) importance were not drawn because these features did not influence the training of any models, even worse it seems that these parameters are noising features as they importance score is sometimes negative. This was predictable as the dMRI parameters were consciously not normalized, the range from ADC $\lambda_\perp$,$\lambda_\parallel$ made them invisible for the model. Several observations can be made on the feature importances:

- The 15 summed spherical harmonics are more important than the FA in model predictions. More precisely, only the FAs of sequences with b values greater than or equal to 550 have an impact on learning.

- The importance of SH increases with bvalue, with a first characteristic step when passing from 6-direction sequences (bvalue = 300) to 30-direction sequences (bvalue = 550) and another step when passing from 30-directions sequences (bvalue = 1000) to 60-direction sequences (bvalue = 3200). These steps are positive in the case of AD, GAD, VF, and Z mean orientation coordinates, and negative in the case of X and Y mean orientation coordinates.

- The feature importances of X and Y mean orientation coordinates show the same trend, which is consistent with the fact that these two parameters simulate a similar geometry and have been

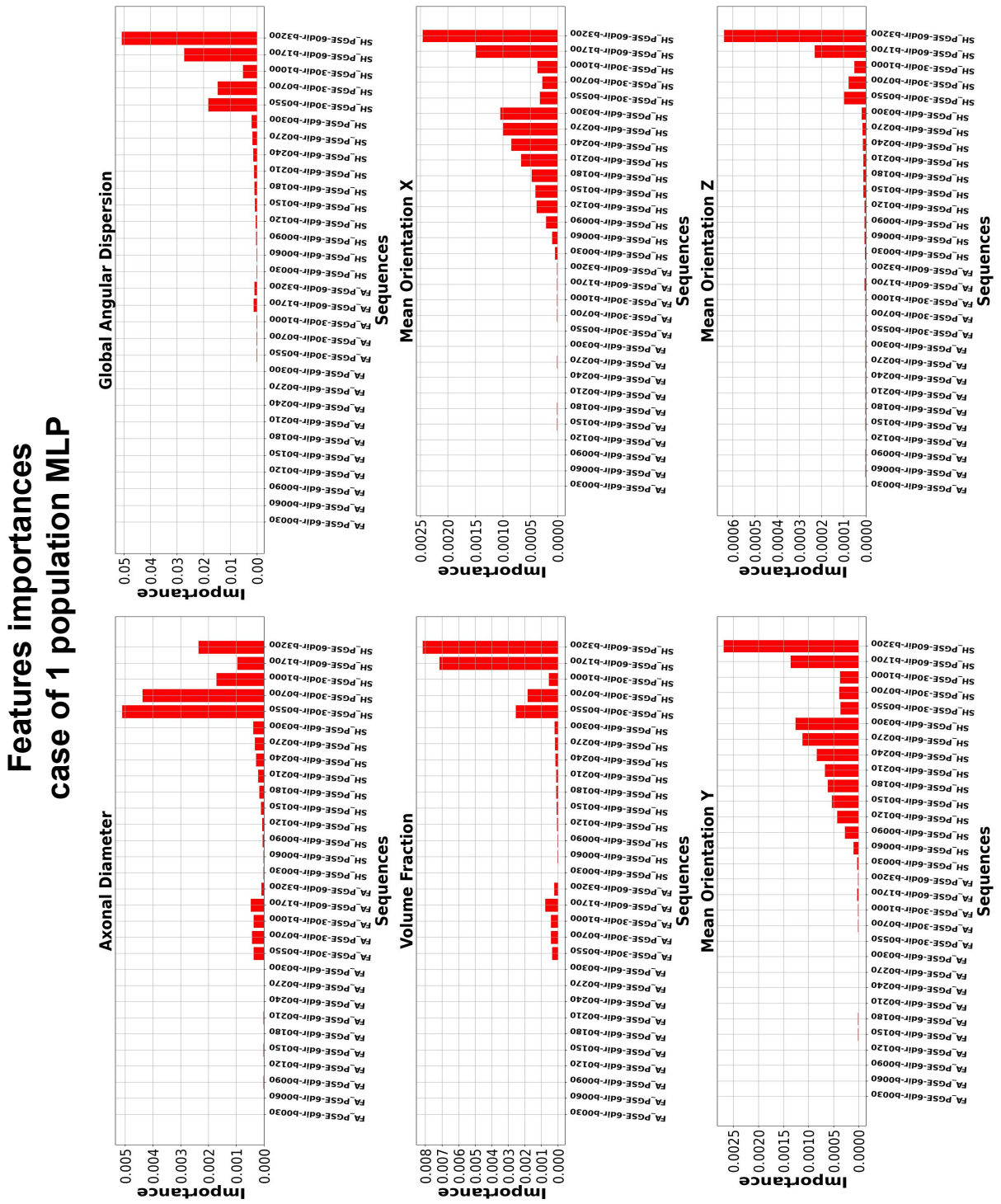


Figure 6.16: *Feature importances for each generative parameter predicted by the 1 fiber population MLP. The features are grouped according to the iCORTEX sequence they belong to, the importances of the 15 spherical harmonics obtained for each sequence are therefore added together. The importance value is computed using a features permutation technique.*

equally distributed. Z coordinates differ from X and Y, which can be explained by the different distributions implied by the rotation algorithm.

- Except from the AD, sequences with great b-value and numerous directions (bvalue = 1700 and bvalue = 3200, 60 directions) show the most significant importances overall.

The fact that the importance of features correlates with the sequences from which the different parameters stem is an expected result. DMRI signals from large b-value sequences are more sensitive to the simulated microstructure inside the samples. However, the relationship is not linear, as some inconsistencies in this rule can be observed, notably for AD or fiber orientation coordinates along X and Y. These inconsistencies may be explained by model optimization, which is difficult to pin down, or they may be due to the affinity of certain generative parameters, distributed in certain ranges, with specific sequences. This is the case with AD, which gives more importance to sequences with bvalue = 550 than to those with greater bvalues.

6.3.4 Discussion

The established computational model of white matter microstructure shows promising validation results. The capabilities and limitations of our model must now be discussed.

First, the results of our model are particularly efficient for decoding samples with a single fiber population. Different reasons can explain the correlation between decoding efficiency and the number of populations: from a computational point of view, 2 and 3 populations correspond to a larger number of parameters to decode, and despite feeding the models with more samples, the complexity of the regression is thus higher than with 1 fiber population. From a dMRI perspective, the signal used to decode the microstructure lacks information to correctly distinguish the generative parameters from the different populations.

Overall, we can state that the ability of the models to decode certain generative parameters better than others can be explained by the impact of these parameters on the diffusion signal. For example, the GAD is distributed between [5°, 15°], thus impacting the geometry less than the fiber population orientation parameter which variation contributes to the 3D rotation of the whole fiber population.

One limitation of our approach, highlighted by [figure 6.13, 6.14, 6.15] is that the AD, the GAD and the VF profited from less unique samples during the training than fiber orientation due to the augmentation of the data-set with the rotation and noising algorithm. More precisely there are 100 times more uniquely distributed orientations than AD, GAD, and VF. Reducing the rotation count might balance the accuracy of predictions.

To improve the precision of our predictions, the straightforward approach would be to feed our networks with more samples by launching larger simulation campaigns. Moreover, the simulated samples must benefit from PGSE sequences with high b-values and numerous orientations, as signal stemming from such sequences has more impact on the network learning. However, it should be noted that in the case of the current presented simulation campaign, the number of directions and the b-values were specifically designed to suit the iCORTEX protocol, this constraint was added to be able to test directly our model on acquisitions stemming from real hardware MRI scans.

This led us to a limit of our computational model based on MLP networks: it relies on the simulations of signal attenuations stemming from dMRI sequences that must suit hardware protocol, as our model only works for a specific set of dMRI data. This is limiting the potential application of a MEDUSA model to any kind of dMRI acquisition. A few years ago a direct solution to tackle this issue would have been to implement a "zero padding" method, consisting of replacing missing values from dMRI acquisitions with "0", but this solution is pretty limited [stackexchange 2016], the best would be to replace our current limited MLP network with recurrent neural networks (RNN), suitable for such a task. Even better, the use of "transformers" [Vaswani et al. 2017] might excel for such a task, as last year such technology tended to replace RNN architecture [Budu 2023]. Changing the type of neural network architecture might as well change our current model consisting of separating the different population decoding into different networks. Nevertheless, the use of MLP during this thesis was driven by the will to propose, at any cost, a proof of concept regarding the decoding of microstructure, without consideration of a multi-sequential application of the model. Furthermore, the use of MLP reduced the amount of data required as well as the complexity of both network design and learning.

The implementation of our rotation and noise algorithm can be replaced by neural networks directly implementing the rotational variation of the dMRI signal like graph convolutional neural networks [Chen et al. 2020] or spherical convolutional neural networks [T. S. Cohen et al. 2018].

Providing more samples, or changing the network architecture might not be the only solution to improve the precision of our model. As seen in [figures 6.16] parameters as ADC or axial and transversal diffusivity are irrelevant for the learning due to their lower distribution range. To avoid this issue it is necessary to normalize the dMRI parameters. The current models avoid the normalization of dMRI parameters as this solution proves itself to be the one providing the best results, but in the future, a suitable normalization scheme must be found to optimize the input dMRI parameter space. Furthermore, it could be worth removing some dMRI parameters from the input as some of them appear to contribute negatively to the training.

Increasing the learning data-set or the model's architecture might imply moving the model's training to a better computation hardware resource. Indeed, the results presented regarding the training of the network were performed on a local workstation with an NVIDIA® RTX A5000 GPU comprising 16Gb of RAM. This limitation restrained the number of neurons in our MLP networks to fit as close as possible to the RAM capacity. Launching our training on GPU-oriented HPC is doable considering the resources accessible in the laboratory notably the access to the "Jean Zay" supercomputer hosted in the IDRISS [IDRISS 2022] HPC facility. In the future, if the choice to change architecture is made, notably with the implementation of transformer networks within the model, it is strongly recommended to consider launching the training on available HPC or servers more suited to GPU-intensive tasks.

If we take a step back from the training of the neural network, we can discuss as well the choices made to define the framework of our simulations, in other words, the definition of the fiber populations within the samples. In the case of a single fiber population, the results are promising and in the future, other microstructural parameters such as tortuosity or Ranvier nodes could be studied. In addition, certain parameters such as AD or GAD could benefit from an extended range of variation, especially for GAD, whose contribution to the signal decrease appears to be erased compared with other decoded parameters such as fiber orientation. For cases with two and three populations, it is getting tricky, as the definition of what constitutes a population in the sense of MEDUSA does not

always make sense in the context of the simulated dMRI signals, and by extension, to dMRI signal stemming from hardware acquisitions. For example, within MEDUSA and the simulation framework, two populations of fibers within a sample can have similar geometric parameters, which creates a problem during decoding because the sample might most likely be decoded as a sample with a single population, which is true in term of dMRI analysis, but wrong in term of MEDUSA computation, leading to the computation of an error during training. To limit this phenomenon, the simulations currently presented have seen the orientation of their fibers separated by a minimum angle of 5° between the populations along the two axes of the spherical coordinates. This criterion might be too tight, in the future we recommend increasing this threshold angle, to differentiate populations more significantly from each other and thus better characterize fiber populations between each other.

6.3.5 Conclusion

The white matter microstructure decoding model presented here shows promising results in the decoding of samples composed of one fiber population. As the models were trained to fit iCORTEX acquisitions, a good proof of concept would be to apply MEDUSA models to an iCORTEX acquisition centered on the corpus callosum, with a priori on the number of fiber population in the area fixed at one.

The design of the decoding pipeline proved itself to involve a great complexity due to the number of sequences and the consideration of three different scenarios for the geometry (1, 2, and 3 fiber populations). To tackle this complexity, the coding of the pipeline beneficiate from an optimized and adaptative API, making further development on the Ginkgo framework faster.

Chapter 7

Conclusion

To conclude this thesis, the following sections will list the contribution and the milestones reached, present the prospect, and finally conclude.

7.1 Contributions

The following sections summarise the various contributions of this thesis work.

7.1.1 Optimization of MEDUSA simulator and new input user-friendly dictionary

From the very beginning, the focus was put on improving the MEDUSA simulator capacity to be HPC compliant. Consequently, the code of both the geometry and diffusion simulations was entirely re-designed to improve their computational efficiency and to allow its evolution using a generic coding approach. Moreover, its API was designed to be able to be used either with command lines or through the Ginkgo python module using a simple JSON config file. Two levels of input dictionaries were defined: one dedicated to the simulation of individual MEDUSA samples, and the other dedicated to the management of a full simulation campaign through the use of META-JSON. Additional monitoring tools were developed such as the computation of the total energy from the remove overlap process.

7.1.2 Validation of MEDUSA

To validate the consistency and accuracy of the MEDUSA simulations, several testing campaigns were launched to quantify the impact of different parameters on diffusion signal acquisition. Notably, the RSL, FOV, and remove overlap iteration count were investigated, resulting in optimal settings for the large-scale simulation campaigns.

7.1.3 Launching of a simulation campaign on HPC facility

A total of three simulation campaigns were launched during the frame of this thesis, approximately two months long each, and all carried on the TGCC HPC facility. The environment developed to allow

efficient use and exchange with the HPC, like dockers and job submission, now makes the MEDUSA simulation framework very convenient to deploy on supercomputers.

7.1.4 Implementing a neural network decoding the white matter microstructure

Four different MLP neural networks were designed and optimized to suit the regression problem brought by the decoding of white matter microstructure. The pre- and post-processing0 code allowing formatting of the data stemming from the simulation will benefit the future application of MEDUSA models. The analysis of the impact made by both generative and diffusion features on the learning capacity of the model will help future investigation and participate in a better sampling of the parameters from simulations.

7.2 Prospects

7.2.1 Web servibe MEDUSA

MEDUSA simulations are launched by inputting JSON dictionaries, which provide better visibility of the various simulation parameters, thereby increasing ease of use. The next step would be to offer a web service that combines the ergonomics of JSON dictionaries with the laboratory's access to a powerful computing infrastructure (TGCC and local server). In this way, users unaware of all the MEDUSA functionality or JSON would be invited to complete different online widgets, the ergonomics of which would be simplified and directly connected to fill in the various simulation JSONs. This project could democratize the use of MEDUSA to many people and thus contribute to its perenniality and visibility.

7.2.2 Implement a new neural network architecture

The current model suffers from its inability to adapt the input made of dMRI parameters stemming from PGSE sequences. This limitation is due to the choice of an MLP architecture for our networks. In the future, a model based on recurrent neural networks (RNN) or transformers is to be preferred to improve the efficiency of the model.

7.2.3 Sampling more geometry parameters

The simulation campaign proposed in this thesis proposed to distribute 4 different parameters of the geometry of fiber populations, namely: the axonal diameter, the volume fraction, the global angular dispersion and, the mean orientation. Additional parameters such as tortuosity, or Ranvier nodes could be distributed to extend the number of decoded WM microstructure parameters. Moreover, additional cell populations as glial cells, already existing and implemented within the MEDUSA framework, could be added to the simulations, thus extending the number of elements perceptible by the decoder.

7.2.4 Test more parameters from the simulation

The various test campaigns presented during this thesis have enabled us to clarify the optimal choice of certain parameters in terms of realism and computational time. We would like to take these test campaigns a step further, in particular by quantifying the impact of the presence of Ranvier nodes

within the simulation. A dedicated and simplified pipeline would be recommended for these tests, enabling simple adaptation and rapid transposition to an HPC facility.

7.2.5 Simulation of new white matter tissues

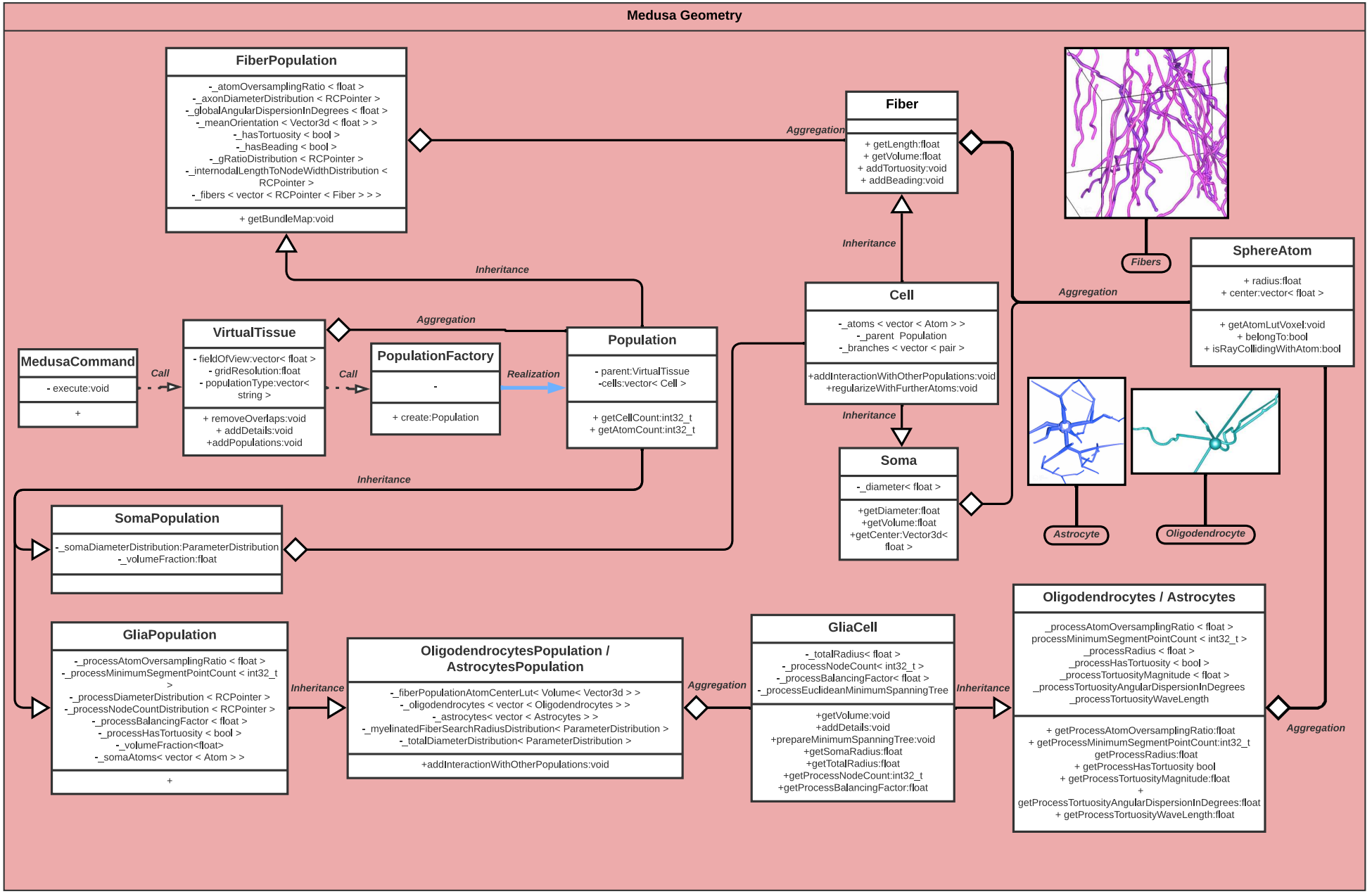
MEDUSA can currently simulate approximately over 90% of the cell volume fraction found in WM. However, the simulation of glial cells such as microglia or vascularisation is still lacking. Simulating micro-vascularity could have an impact on the simulation of the diffusion process, adding the necessity to simulate blood flow and its contribution to the signal.

7.3 Conclusion

The computational approach of a model for decoding the microstructure of white matter, supported by digitally synthesized samples, is promising and constitutes an alternative to analytical models. The work presented has numerous practical applications for patients, notably through the creation of an additional diagnostic tool for doctors. The topic of this thesis is multi-disciplinary, opening the way to numerous developments and future reflections, facilitated by the flexibility and portability of the Ginkgo MEDUSA software environment.

APPENDIX

Medusa geometry generation UML



Medusa remove overlapp UML

MedusaCommand
- execute:void
+

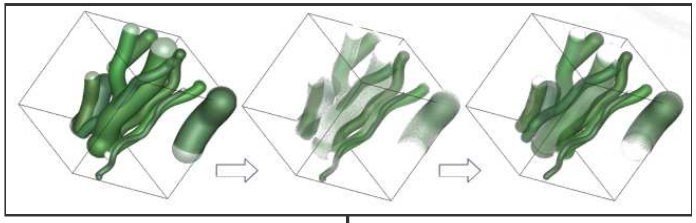
Medusa Geometry

VitualTissue
- fieldOfView:vector< float >
- gridResolution:float
- populationType:vector< string >
+ removeOverlaps:void
+ addDetails:void
+addPopulations:void

AtomColliderFactory
- _atomColliders < AtomCollider >
+ getAtomCollider:AtomCollider

AtomSmoother Factory
- _atomSmoothers < AtomSmoother >
+ getAtomSmoother:AtomSmoother

AtomForceApplierFactory
- _atomForceAppliers < AtomForceAppliers >
+ getAtomForceApplier:AtomForceApplier



Remove overlapp procedure

KOKKOS KERNELS

AtomOverlapSolver

struct RepulsionForceInitializer
- forces < Float3VectorView >
+ RepulsionForceInitializer

struct RepulsionForceComputer
- atomCount < int32_t >
- atomCompressedRepresentations < BoolVectorView >
- atomTypes UInt8VectorView
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64vectorView >
- populationIds < IntVectorView >
- cellIds < Int32VectorView >
- lower X/Y/Z < float >
- span X/Y/Z < float >
- lutSizeX/Y/Z < int32_t >
- lutSize XY < int32_t >
- lutSize XYZ < int32_t >
- lutAtomCountPerVoxel < Int32VectorView >
- atomIndexLut < Int32VectorView >
- neighborhoodHalfSize < int32_t >
- forces < Float3VectorView >
+ RepulsionForceComputer
+ theAtomCompressedRepresentations

struct TotalRepulsionForceNormComputer
- forces < Float3VectorView >
-value_type < float >
+ TotalRepulsionForceNormComputer

struct AtomSmoother
- atomCount < int32_t >
- atomTypes < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64VectorView >
- atomBranchLabels < UInt32VectorView >
+ AtomSmoother

struct RepulsionForceApplier
- atomTypes < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParametersOffsets < UInt64VectorView >
- forces < Float3VectorView >
- maximumForceNorm < float >
+ RepulsionForceApplier

AtomLut

struct AtomCountPerGridVoxelInitializer
- atomCountPerGridVoxel < Int32VectorView >
+ AtomCountPerGridVoxelInitializer

struct SpreadedAtomCountPerGridVoxelComputer
- atomType < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64vectorView >
- lower X/Y/Z < float >
- span X/Y/Z < float >
- lutSizeX/Y/Z < int32_t >
- lutSize XY < int32_t >
- atomCountPerGridVoxel < Int32VectorView >
+ SpreadedAtomCountPerGridVoxelComputer

struct AtomIndexLutComputer
- atomType < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64vectorView >
- lower X/Y/Z < float >
- span X/Y/Z < float >
- lutSizeX/Y/Z < int32_t >
- lutSize XY < int32_t >
- lutSize XYZ < int32_t >
- atomCountPerGridVoxel < Int32VectorView >
- atomOffsetVoxel < Int32VectorView >
- atomIndexLut < Int32VectorView >
+ AtomIndexLutComputer

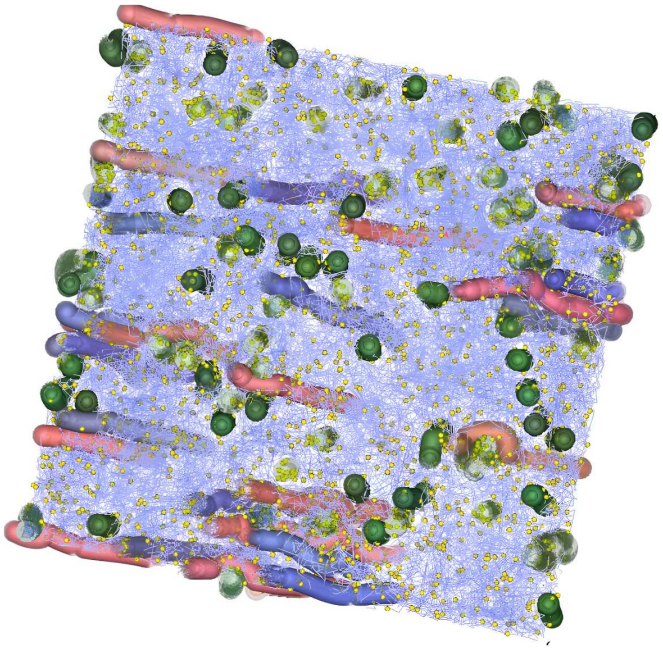
struct TotalAtomCountComputer
- lutAtomCountPerVoxel < Int32VectorView >
+ TotalAtomCountComputer

struct MaximumAtomCountPerGridVoxelComputer
- lutAtomCountPerVoxel < Int32VectorView >
+ MaximumAtomCountPerGridVoxelComputer

struct AtomCountPerGridVoxelComputer
- atomType < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64vectorView >
- lower X/Y/Z < float >
- span X/Y/Z < float >
- lutSizeX/Y/Z < int32_t >
- lutSize XY < int32_t >
- lutSize XYZ < int32_t >
- atomCountPerGridVoxel < Int32VectorView >
+ AtomCountPerGridVoxelComputer

struct SpreadedAtomIndexLutComputer
- atomType < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64vectorView >
- lower X/Y/Z < float >
- span X/Y/Z < float >
- lutSizeX/Y/Z < int32_t >
- lutSize XY < int32_t >
- lutSize XYZ < int32_t >
- atomCountPerGridVoxel < Int32VectorView >
- atomOffsetVoxel < Int32VectorView >
- atomIndexLut < Int32VectorView >
+ SpreadedAtomIndexLutComputer

Medusa diffusion UML



MedusaCommand

- execute:void

+

Aggregation

Medusa Geometry

VitualTissue

- fieldOfView:vector< float >
- gridResolution:float
- populationType:vector< string >

+ removeOverlaps:void
+ addDetails:void
+addPopulations:void

Medusa Diffusion

MonteCarloDiffusionProcess

- _virtualTissue < VirtualTissue >
- _particleInitializationType < Type >
- _particleToMembraneInteractionType < ParticleToMembraneInteractionType >
- _sessionCount < int32_t >
- _particleDiffusivityInUm2PerUs < float >
- _particleCount < int32_t >
- _gridResolutionInUm < Vector3d < double > >
- _timeStepInUs < float >
- _totalDurationInUs < float >
- _stepLength < float >
- *stepCount* < int32t >
- _particleBackupPercentage < float >
-_randomParticleSelection < vector < int32_t > >

+ simulate:void

call

KOKKOS KERNELS

kokkos

DiffusionProcessKernels

struct VirtualTissueKokkosContainer

- atomCount < int_32_t >
- atomTypes < UInt8VectorView >
- atomCompressedRepresentations < BoolVectorView >
- atomParameters < FloatVectorView >
- atomParameterOffsets < UInt64VectorView >
- atomPopulationIds < Int32VectorView >
- atomCellIds < Int32VectorView >
- atomCountPerGridVoxel < Int32VectorView >
- atomOffsetPerGridVoxel < Int32VectorView >
- atomIndexLut < Int32VectorView >
- fieldOfView < BoundingBox < float > >
- fieldOfViewLowerX/Y/Z < float >
- fieldOfViewSpanX/Y/Z < float >
- gridResolutionInUmX/Y/Z < float >
- atomLutSizeX/Y/Z < float >
- atomLutSizeXY < float >
- atomLutSizeXYZ < float >
- maximumAtomCountPerGridVoxel < int32_t >

+ getAtomLutIndices:void
+ belongToAtom:bool
+ inBox:bool
+ getDistanceToMebrane:float
+ getDistanceToMembrane:void
+ getElasticBounceDirection:void
+ isRayCollindingWithAtom

struct DiffusionProcessParticleInitializer

- virtualTissueKokkosContainer < RCPointer< VirtualTissueKokkosContainer > >
- particleCoordinates < Float3VectorView >
- particleIds < Int32VectorView >
randPool < DefaultExecutionSpace >

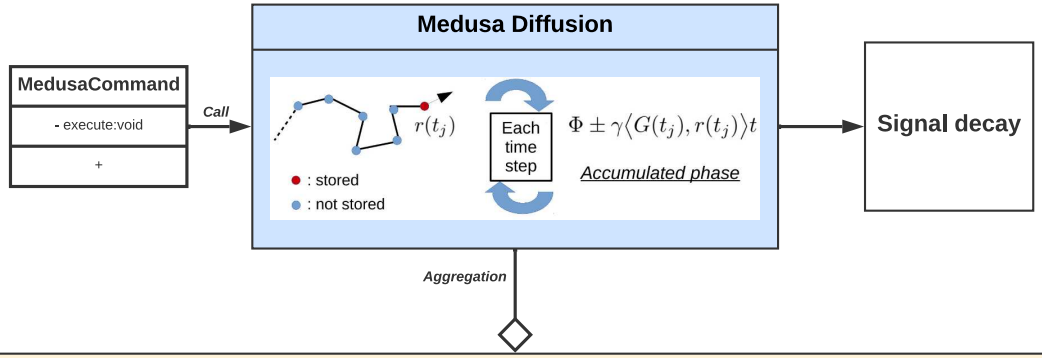
+ enum Type < Everywhere,
OnlyIntracellular,
OnlyExtracellular,
AllParticlesAtFieldOfViewCenter >

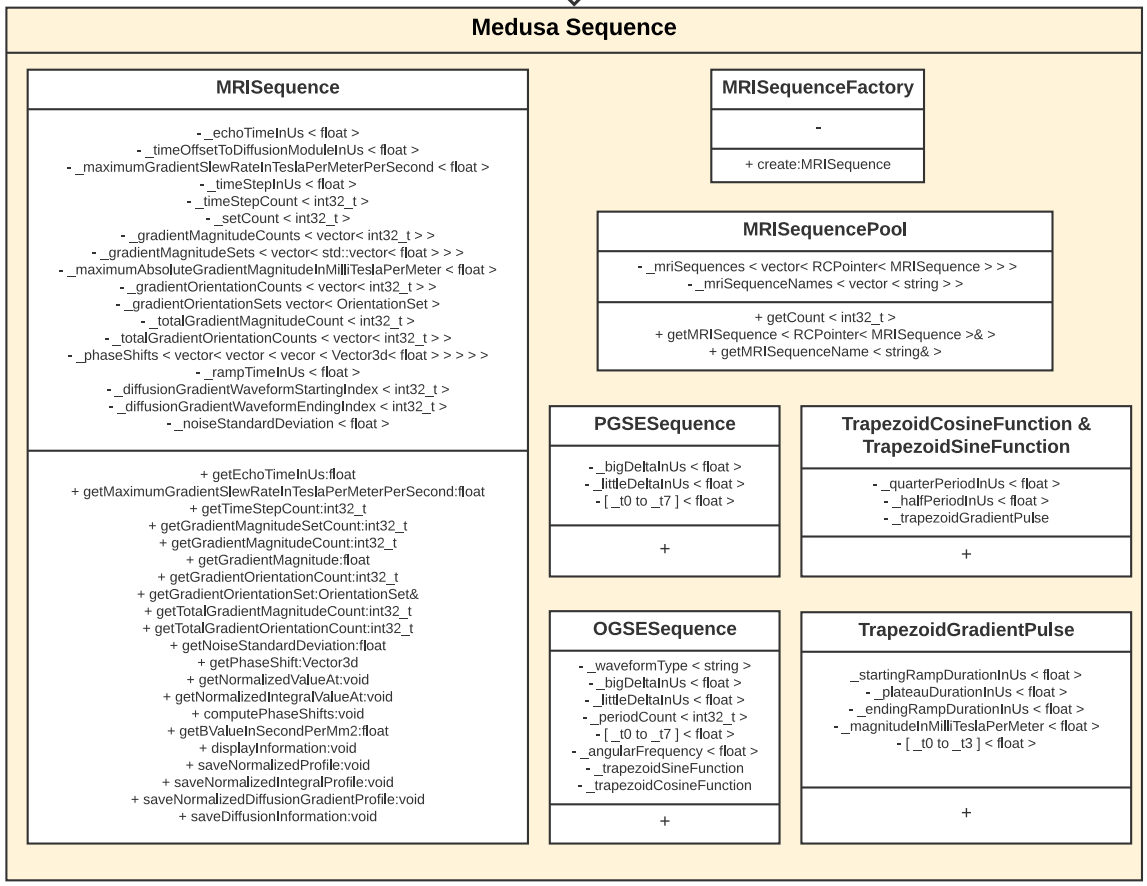
struct DiffusionProcessMonteCarloAndMRISimulator

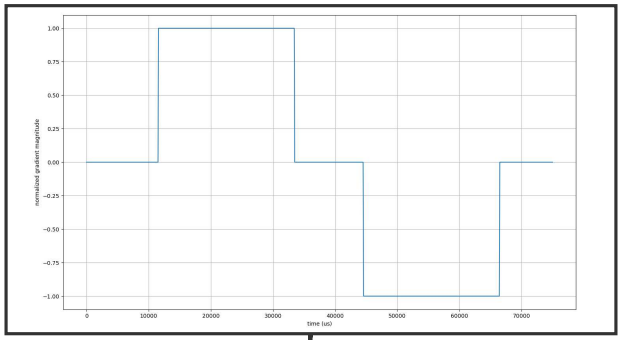
- virtualTissueKokkosContainer<RCPointer< VirtualTissueKokkosContainer > >
- cellPermeabilities < FloatVectorView >
- particleDiffusivityInUm2PerUs < float >
- stepLength < float >
- particleToMembraneInteractionType < ParticleToMembraneInteractionType >
- particleCount < int32_t >
- particleCoordinates < Float3VectorView >
- particleIds < Int32VectorView >
- randPool < DefaultExecutionSpace >
- mriSequenceCount < int32_t >
- totalDiffusionWeightedContrastCount < int32_t >
- allSequencesPhaseShifts < FloaVectorView >
- perContrastMriSequenceIndices < Int32VectorView >
- perSequenceTimeStepCounts < Int32VectorView >
- perSequencePhaseShiftStartingIndices < Int32VectorView >
- perSequenceDiffusionWeightedContrastCounts < Int32VectorView >
- formerAccumulatedDiffusionWeightedContrastCounts < Int32VectorView >
- allSequencesPhaseAccumulators < vector< FloatVectorView > >
- currentTimeStep < int32_t >
- twiceStepLengthOverThreeDiffusivity < float >

+ enum ParticleToMembraneInteractionType <
EqualStepLengthRandomLeap,
ElasticBouncing >
+ resetTimeStep:void
+ step:void
+ getPhaseShiftIndex:uint64_t
+ computeSubStepsWithElasticBouncing:bool
+ computeSubStepsWithEqualStepLengthRandomLeap:bool

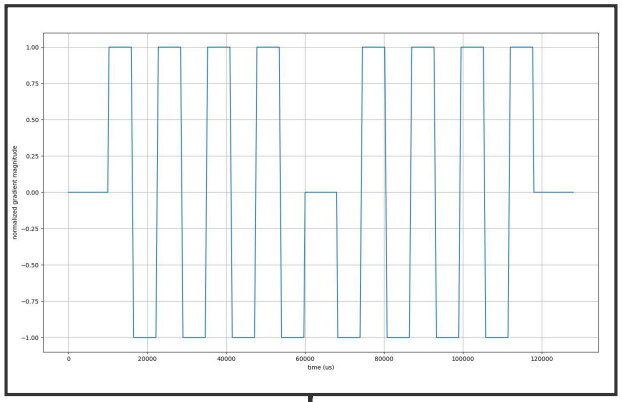
Medusa sequence UML



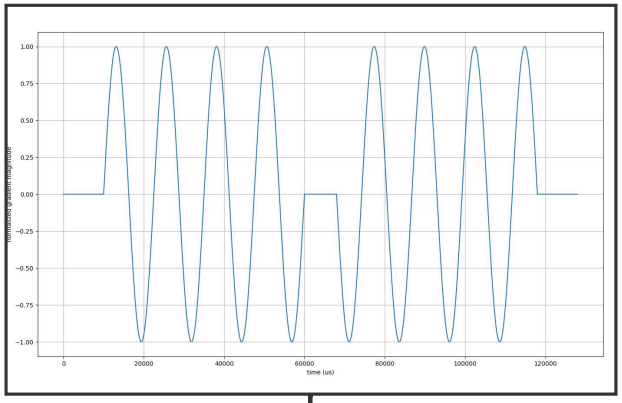




Sequence PGSE



Sequence OGSE trapezoid sine



Sequence OGSE sine

Bibliography

[Abdollahzadeh et al., 2017] Abdollahzadeh, Ali et al. (2017). 3D axonal morphometry of white matter. *bioRxiv* (cit. on p. 92).

[Aboitiz, R.S. Fisher, and E.Zaidel, 1992] Aboitiz, F., A.B. Scheibel ans R.S. Fisher, and E.Zaidel (1992). Fiber composition of the human corpus callosum. *Brain research* (cit. on pp. 17, 25, 63).

[Agarwal and Carare, 2021] Agarwal, Nivedita and Roxana Octavia Carare (2021). Cerebral vessels: an overview of anatomy, physiology, and role in the drainage of fluids and solutes. *Frontiers in neurology* 11, p. 611485 (cit. on p. 22).

[Agirre-Basurko, Ibarra-Berastegi, and Madariaga, 2006] Agirre-Basurko, Elena, Gabriel Ibarra-Berastegi, and I Madariaga (2006). Regression and multilayer perceptron-based models to forecast hourly O3 and NO2 levels in the Bilbao area. *Environmental Modelling & Software* 21.4, pp. 430–446 (cit. on p. 167).

[Alabdullatef, 2020] Alabdullatef, Layan (2020). *Complete Guide to Adam Optimization*. url: https://medium.com/@LayanSA/complete-guide-to-adam-optimization-1e5f29532c3d (cit. on p. 179).

[Alexander, 2008] Alexander (2008). A general framework for experiment design in diffusion MRI and its application in measuring direct tissue-microstructure features. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 60.2, pp. 439–448 (cit. on p. 64).

[Alexander, 2010] Alexander (2010). Orientationally invariant indices of axon diameter and density from diffusion MRI. *Neuroimage* 52.4, pp. 1374–1389 (cit. on pp. 64, 188).

[Aliyu, Mokhtar, and Hussin, 2022] Aliyu, Ruwaida, Ainul Akmar Mokhtar, and Hilmi Hussin (2022). "Classification of Pump Failure Using a Decision Tree Technique". In: *International Conference on Renewable Energy and E-mobility*. Springer, pp. 319–336 (cit. on p. 172).

[Amunts and Zilles, 2015] Amunts, Katrin and Karl Zilles (2015). Architectonic Mapping of the Human Brain beyond Brodmann. *Neuron* 88.6, pp. 1086–1107 (cit. on pp. 10, 23).

[Anatomie-amsterdam, 2024] Anatomie-amsterdam (2024). *Base ganglia*. url: http://www.anatomie-amsterdam.nl (cit. on p. 12).

[Andriezen, 1893] Andriezen (1893). The neuroglia elements in the human brain. *British medical journal* 2.1700, p. 227 (cit. on p. 19).

[Ascoli, Donohue, and Halavi, 2007] Ascoli, Giorgio A, Duncan E Donohue, and Maryam Halavi (2007). NeuroMorpho. Org: a central resource for neuronal morphologies. *Journal of Neuroscience* 27.35, pp. 9247–9251 (cit. on pp. 73, 81).

[Assaf and Basser, 2005] Assaf, Yaniv and Peter J Basser (2005). Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain. *Neuroimage* 27.1, pp. 48–58 (cit. on p. 63).

[Assaf, Blumenfeld-Katzir, et al., 2008] Assaf, Yaniv, Tamar Blumenfeld-Katzir, et al. (2008). AxCaliber: a method for measuring axon diameter distribution from diffusion MRI. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 59.6, pp. 1347–1354 (cit. on p. 63).

[Assaf and Y. Cohen, 1998] Assaf, Yaniv and Yoram Cohen (1998). Non-mono-exponential attenuation of water andn-acetyl aspartate signals due to diffusion in brain tissue. *Journal of Magnetic Resonance* 131.1, pp. 69–85 (cit. on p. 60).

[Assaf and Y. Cohen, 2000] Assaf, Yaniv and Yoram Cohen (2000). Assignment of the water slow-diffusing component in the central nervous system using q-space diffusion MRS: implications for fiber tract imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 43.2, pp. 191–199 (cit. on pp. 45, 50, 51, 60).

[Assaf, Kafri, et al., 2002] Assaf, Yaniv, Michal Kafri, et al. (2002). Changes in axonal morphology in experimental autoimmune neuritis as studied by high b-value q-space 1H and 2H DQF diffusion magnetic resonance spectroscopy. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 48.1, pp. 71–81 (cit. on p. 60).

[Avram, Assaf, and Y. Cohen, 2004] Avram, Liat, Yaniv Assaf, and Yoram Cohen (2004). The effect of rotational angle and experimental parameters on the diffraction patterns and micro-structural information obtained from q-space diffusion NMR: implication for diffusion in white matter fibers. *Journal of Magnetic Resonance* 169.1, pp. 30–38 (cit. on pp. 61, 63).

[Bacyinski et al., 2017] Bacyinski, A. et al. (2017). The paravascular pathway for brain waste clearance: current understanding, significance and controversy. *Frontiers in neuroanatomy* 11, pp. 101–1172 (cit. on p. 6).

[Baete et al., 2008] Baete, SH et al. (2008). Microstructural analysis of foam by use of NMR R2 dispersion. *Journal of Magnetic Resonance* 193.2, pp. 286–296 (cit. on p. 73).

[Balls and L. R. Frank, 2009] Balls, Gregory T and Lawrence R Frank (2009). A simulation environment for diffusion weighted MR experiments in complex media. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 62.3, pp. 771–778 (cit. on pp. 68–70, 75, 78).

[Bar-Shir et al., 2008] Bar-Shir, Amnon et al. (2008). The effect of the diffusion time and pulse gradient duration ratio on the diffraction pattern and the structural information estimated from q-space diffusion MR: experiments and simulations. *Journal of Magnetic Resonance* 194.2, pp. 230–236 (cit. on p. 50).

[Basser, Mattiello, and LeBihan, 1994] Basser, Peter J, James Mattiello, and Denis LeBihan (1994). MR diffusion tensor spectroscopy and imaging. *Biophysical journal* 66.1, pp. 259–267 (cit. on p. 47).

[Beaujoin, 2018] Beaujoin, Justine (2018). "Post mortem inference of the human brain microstructure using ultra-high field magnetic resonance imaging with strong gradients". PhD thesis. Université Paris Saclay (COmUE) (cit. on pp. 43, 44).

[Beaulieu, 2011] Beaulieu, Christian (2011). *Diffusion MRI: theory, methods, and application/edited by Derek K. Jones* (cit. on p. 44).

[Beaulieu and Allen, 1994] Beaulieu, Christian and Peter S Allen (1994). Determinants of anisotropic water diffusion in nerves. *Magnetic resonance in medicine* 31.4, pp. 394–400 (cit. on p. 58).

[T. E. J. Behrens et al., 2007] Behrens, T. E. J. et al. (2007). Probabilistic diffusion tractography with multiple fibre orientations: What can we gain? *NeuroImage* 34.1, pp. 144–145 (cit. on pp. 25, 49).

[T. E. Behrens et al., 2003] Behrens, Timothy EJ et al. (2003). Characterization and propagation of uncertainty in diffusion-weighted MR imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 50.5, pp. 1077–1088 (cit. on p. 63).

[Berman et al., 2008] Berman, Jeffrey I et al. (2008). Probabilistic streamline q-ball tractography using the residual bootstrap. *Neuroimage* 39.1, pp. 215–222 (cit. on p. 53).

[Berthold, I. Nilsson, and Rydmark, 1983] Berthold, Claes-Henric, I Nilsson, and M Rydmark (1983). Axon diameter and myelin sheath thickness in nerve fibres of the ventral spinal root of the seventh lumbar nerve of the adult and developing cat. *Journal of anatomy* 136.Pt 3, p. 483 (cit. on p. 17).

[Biendenbach, DeVito, and A. C. Brown, 1986] Biendenbach, M. A., J. L. DeVito, and A. C. Brown (1986). Pyramidal tract of the cat: axon size and morphology. *Experimental Brain Research* 61, pp. 303–310 (cit. on p. 25).

[Bloch, 1946] Bloch, Felix (1946). Nuclear induction. *Physical review* 70.7-8, p. 460 (cit. on pp. 30, 34).

[Borrell, 2018] Borrell, Víctor (2018). How cells fold the cerebral cortex. *Journal of Neuroscience* 38.4, pp. 776–783 (cit. on p. 11).

[Bossler and Valdez, 2018] Bossler, Kerry and Greg D Valdez (2018). *Comparison of Kokkos and CUDA Programming Models for Key Kernels in the Monte Carlo Transport Algorithm.* Tech. rep. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States) (cit. on p. 137).

[Bourne, 2012] Bourne, Geoffrey (2012). *The Structure and Function of Nervous Tissue V2: Structure I*. Elsevier (cit. on pp. 25, 75).

[Breneman, J. R. Purcell, and Burnett, 1987] Breneman, Bruce Charles, John Russell Purcell, and Sibley Charles Burnett (1987). *MRI patent*. url: https://patents.google.com/patent/EP0310212A2/en (cit. on p. 29).

[Brodmann, 1909] Brodmann, K. (1909). *Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues*. Leipzig : Barth (cit. on p. 23).

[Bronlee, 2019] Bronlee, Json (2019). *How to Configure the Number of Layers and Nodes in a Neural Network*. url: https://machinelearningmastery.com/how-to-configure-the-number-of-layers-and-nodes-in-a-neural-network/ (cit. on p. 174).

[R. Brown, 1827] Brown, Robert (1827). *Prodromus florae Novae Hollandiae et Insulae Van-Diemen: exhibens characteres plantarum*. Vol. 3. Sumtibus L. Schrag (cit. on pp. 41, 75).

[Bruner, 2022] Bruner, Emiliano (2022). A network approach to the topological organization of the Brodmann map. *The Anatomical Record* 305.12, pp. 3504–3515 (cit. on p. 23).

[Budde and J. A. Frank, 2010] Budde, Matthew D and Joseph A Frank (2010). Neurite beading is sufficient to decrease the apparent diffusion coefficient after ischemic stroke. *Proceedings of the National Academy of Sciences* 107.32, pp. 14472–14477 (cit. on pp. 71, 92).

[Budu, 2023] Budu, Emanuella (2023). *Random Forest vs. Extremely Randomized Trees*. url: https://www.baeldung.com/cs/random-forest-vs-extremely-randomized-trees (cit. on p. 195).

[Burcaw, Fieremans, and Novikov, 2015] Burcaw, Lauren M, Els Fieremans, and Dmitry S Novikov (2015). Mesoscopic structure of neuronal tracts from time-dependent diffusion. *NeuroImage* 114, pp. 18–37 (cit. on pp. 60, 69).

[Bystron, Blakemore, and Rakic, 2008] Bystron, Irina, Colin Blakemore, and Pasko Rakic (2008). Development of the human cerebral cortex: Boulder Committee revisited. *Nature Reviews Neuroscience* 9.2, pp. 110–122 (cit. on p. 22).

[P. Callaghan, 1995] Callaghan, P.T. (1995). Pulsed-Gradient Spin-Echo NMR for Planar, Cylindrical, and Spherical Pores under Conditions of Wall Relaxation. *Journal of Magnetic Resonance, Series A* 113.1, pp. 53–59 (cit. on p. 116).

[P. T. Callaghan et al., 1991] Callaghan, Paul T et al. (1991). Diffraction-like effects in NMR diffusion studies of fluids in porous solids. *Nature* 351.6326, pp. 467–469 (cit. on pp. 50, 51).

[R. Callaghan et al., 2020] Callaghan, Ross et al. (2020). ConFiG: Contextual Fibre Growth to generate realistic axonal packing for diffusion MRI simulation. *NeuroImage* 220, p. 117107 (cit. on pp. 71, 72, 75).

[Campbell et al., 2005] Campbell, Jennifer SW et al. (2005). Flow-based fiber tracking with diffusion tensor and q-ball data: validation and comparison to principal diffusion direction techniques. *NeuroImage* 27.4, pp. 725–736 (cit. on p. 53).

[Carlson et al., 2011] Carlson, Verginia C Cuzon et al. (2011). Subsets of spiny striosomal striatal neurons revealed in the Gad1-GFP BAC transgenic mouse. *Basal Ganglia* 1.4, pp. 201–211 (cit. on p. 18).

[Carr and E. M. Purcell, 1954] Carr, Herman Y and Edward M Purcell (1954). Effects of diffusion on free precession in nuclear magnetic resonance experiments. *Physical review* 94.3, p. 630 (cit. on p. 30).

[Chauvel et al., 2023] Chauvel, Maëlig et al. (2023). In vivo mapping of the deep and superficial white matter connectivity in the chimpanzee brain. *NeuroImage* 282, p. 120362 (cit. on pp. 15, 40).

[Chen et al., 2020] Chen, Geng et al. (2020). Estimating tissue microstructure with undersampled diffusion data via graph convolutional neural networks, pp. 280–290 (cit. on p. 195).

[Chenevert, Brunberg, and Pipe, 1990] Chenevert, Thomas L, James A Brunberg, and James G Pipe (1990). Anisotropic diffusion in human white matter: demonstration with MR techniques in vivo. *Radiology* 177.2, pp. 401–405 (cit. on p. 58).

[Chin et al., 2002] Chin, Chih-Liang et al. (2002). Biexponential diffusion attenuation in the rat spinal cord: computer simulations based on anatomic images of axonal architecture. *Magnetic Resonance*

in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine 47.3, pp. 455–460 (cit. on p. 73).

[Chomiak and Hu, 2009] Chomiak, T. and B. Hu (2009). What Is the Optimal Value of the g-Ratio for Myelinated Fibers in the Rat CNS? A Theoretical Approach. *PLOS ONE* 4. Cited on page 24, e7754 (cit. on p. 17).

[Cipolla, 2016] Cipolla, Marilyn J (2016). The cerebral circulation (cit. on p. 22).

[Clow and Young, 1978] Clow, H and IR Young (1978). Britain's brains produce first NMR scans. *New Scientist* 80.588, b26 (cit. on p. 31).

[T. S. Cohen et al., 2018] Cohen, Taco S et al. (2018). Spherical cnns. *arXiv preprint arXiv:1801.10130* (cit. on p. 195).

[Colgan et al., 2016] Colgan, Niall et al. (2016). Application of neurite orientation dispersion and density imaging (NODDI) to a tau pathology model of Alzheimer's disease. *Neuroimage* 125, pp. 739–744 (cit. on p. 65).

[CUDA, 2024] CUDA (2024). *CUDA*. url: https://developer.nvidia.com/cuda-toolkit (cit. on p. 137).

[CultureMath, 2024] CultureMath (2024). *CultureMath*. url: http://cm2.ens.fr/ (cit. on p. 41).

[Curran, Emsell, and Leemans, 2016] Curran, Kathleen M, Louise Emsell, and Alexander Leemans (2016). Quantitative DTI measures. *Diffusion tensor imaging: A practical handbook*, pp. 65–87 (cit. on p. 49).

[d2l, 2024] d2l (2024). *linear-regression*. url: https://d2l.ai/chapter_linear-regression/linear-regression.html (cit. on p. 163).

[data-mining, 2019] data-mining (2019). *data-mining*. url: https://data-mining.philippe-fournier-viger.com/too-many-machine-learning-papers/ (cit. on p. 162).

[dataversity, 2024] dataversity (2024). *dataversity*. url: https://www.dataversity.net/brief-history-deep-learning/ (cit. on p. 166).

[Del Bigio, 2010] Del Bigio, Marc R (2010). Ependymal cells: biology and pathology. *Acta neuropathologica* 119, pp. 55–73 (cit. on p. 22).

[Descoteaux, 2015] Descoteaux, Maxime (2015). High angular resolution diffusion imaging (HARDI). *Wiley encyclopedia of electrical and electronics engineering*, pp. 1–25 (cit. on p. 53).

[Descoteaux et al., 2007] Descoteaux, Maxime et al. (2007). Regularized, fast, and robust analytical Q-ball imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 58.3, pp. 497–510 (cit. on pp. 54, 149).

[DOCKER, 2024] DOCKER (2024). *DOCKER*. url: https://www.docker.com/ (cit. on p. 140).

[Drobnjak, Neher, et al., 2021] Drobnjak, Ivana, Peter Neher, et al. (2021). Physical and digital phantoms for validating tractography and assessing artifacts. *Neuroimage* 245, p. 118704 (cit. on p. 60).

[Drobnjak, H. Zhang, et al., 2016] Drobnjak, Ivana, Hui Zhang, et al. (2016). PGSE, OGSE, and sensitivity to axon diameter in diffusion MRI: Insight from a simulation study. *Magnetic resonance in medicine* 75.2, pp. 688–700 (cit. on p. 47).

[Dubois et al., 2006] Dubois, Jessica et al. (2006). Optimized diffusion gradient orientation schemes for corrupted clinical DTI data sets. *Magnetic Resonance Materials in Physics, Biology and Medicine* 19, pp. 134–143 (cit. on pp. 80, 106).

[Duh, Mohorič, and Stepišnik, 2001] Duh, Andrej, Aleš Mohorič, and Janez Stepišnik (2001). Computer simulation of the spin-echo spatial distribution in the case of restricted self-diffusion. *Journal of Magnetic Resonance* 148.2, pp. 257–266 (cit. on pp. 69, 75).

[Dunn, 2017] Dunn, Greg (2017). *Microgravure des neurones*. url: https://www.sciencesetavenir.fr/sante/cerveau-et-psy/diaporama-toute-la-magie-et-la-complexite-des-neurones-revelees-dans-une-microgravure_113817 (cit. on p. 3).

[Economo and Koskinas, 1925] Economo, C. v. and G. N. Koskinas (1925). *Die Cytoarchitektonik der Hirnrinde des erwachsenen Menschen*. J. Springer : Berlin (cit. on p. 23).

[Edwards et al., 2017] Edwards, Luke J et al. (2017). NODDI-DTI: estimating neurite orientation and dispersion parameters from a diffusion tensor in healthy white matter. *Frontiers in neuroscience* 11, p. 720 (cit. on p. 65).

[Eickhoff, Yeo, and Genon, 2018] Eickhoff, Simon B, BT Thomas Yeo, and Sarah Genon (2018). Imaging-based parcellations of the human brain. *Nature Reviews Neuroscience* 19.11, pp. 672–686 (cit. on p. 23).

[Einstein, 1905] Einstein, Albert (1905). Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der physik* 4 (cit. on pp. 41, 75).

[Essen, Donahue, and M. Glasser, 2018] Essen, D.C. Van, C.J Donahue, and M.F Glasser (Aug. 2018). Development and Evolution of Cerebral and Cerebellar Cortex. *Brain Behavior and Evolution* 91.3, pp. 158–169 (cit. on p. 23).

[Fang et al., 2023] Fang, Chengran et al. (2023). A simulation-driven supervised learning framework to estimate brain microstructure using diffusion MRI. *Medical Image Analysis* 90, p. 102979 (cit. on pp. 68, 73, 74, 80, 81, 178).

[Fei-Fei, Deng, and K. Li, 2009] Fei-Fei, Li, Jia Deng, and Kai Li (2009). ImageNet: Constructing a large-scale image database. *Journal of vision* 9.8, pp. 1037–1037 (cit. on p. 166).

[Fiala, Harris, and Spacek, 1999] Fiala, J.C., K.M. Harris, and K.H. Spacek (1999). Dendrite structure. *Oxford University Press Oxford* (cit. on p. 17).

[Fick, 1855] Fick, Adolph (1855). V. On liquid diffusion. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 10.63, pp. 30–39 (cit. on p. 41).

[Fieremans, De Deene, et al., 2008] Fieremans, Els, Yves De Deene, et al. (2008). Simulation and experimental verification of the diffusion in an anisotropic fiber phantom. *Journal of magnetic resonance* 190.2, pp. 189–199 (cit. on p. 60).

[Fieremans and Lee, 2018] Fieremans, Els and Hong-Hsi Lee (2018). Physical and numerical phantoms for the validation of brain microstructural MRI: A cookbook. *Neuroimage* 182, pp. 39–61 (cit. on pp. 77, 78, 84, 115).

[Fieremans, Novikov, et al., 2010] Fieremans, Els, Dmitry S Novikov, et al. (2010). Monte Carlo study of a two-compartment exchange model of diffusion. *NMR in Biomedicine* 23.7, pp. 711–724 (cit. on pp. 68, 70, 75, 78, 80).

[Fillard et al., 2011] Fillard, Pierre et al. (2011). Quantitative evaluation of 10 tractography algorithms on a realistic diffusion MR phantom. *Neuroimage* 56.1, pp. 220–234 (cit. on p. 62).

[Fischl and Dale, 2000] Fischl, Bruce and Anders M Dale (2000). Measuring the thickness of the human cerebral cortex from magnetic resonance images. *Proceedings of the National Academy of Sciences* 97.20, pp. 11050–11055 (cit. on p. 11).

[Fisher, Lewis, and Embleton, 1993] Fisher, Nicholas I, Toby Lewis, and Brian JJ Embleton (1993). *Statistical analysis of spherical data*. Cambridge university press (cit. on p. 89).

[Fourier, 1827] Fourier, Joseph (1827). Memoire sur la temperature du globe terrestre et espaces. *France. J. SCciene* 32, pp. 1–20 (cit. on p. 41).

[FRAMAGITMEDUSA, 2024] FRAMAGITMEDUSA (2024). *FRAMAGITMEDUSA*. url: https://framagit.org/cpoupon/gkg (cit. on p. 107).

[Fukushima, 1975] Fukushima, Kunihiko (1975). Cognitron: A self-organizing multilayered neural network. *Biological cybernetics* 20.3, pp. 121–136 (cit. on p. 165).

[Funk, 1915] Funk, Paul (1915). Über eine geometrische anwendung der abelschen integralgleichung. *Mathematische Annalen* 77.1, pp. 129–135 (cit. on p. 53).

[Gamma et al., 1999] Gamma, Erich et al. (1999). Design Patterns Catalogue de modèles de conception réutilisables (Traduction de Jean-Marie Lasvergères). *Editions Vuibert* (cit. on p. 86).

[Gardier et al., 2023] Gardier, Rémy et al. (2023). Cellular Exchange Imaging (CEXI): Evaluation of a diffusion model including water exchange in cells using numerical phantoms of permeable spheres. *Magnetic Resonance in Medicine* (cit. on p. 66).

[Gasser and Grundfest, 1939] Gasser, Herbert S and Harry Grundfest (1939). Axon diameters in relation to the spike dimensions and the conduction velocity in mammalian A fibers. *American Journal of Physiology-Legacy Content* 127.2, pp. 393–414 (cit. on p. 17).

[Gauss, 1809] Gauss, Carl Friedrich (1809). Theoria motus corporum coelestum. *Werke* (cit. on p. 163).

[GENCI, 2024] GENCI (2024). *GENCI*. url: https://genci.fr/ (cit. on pp. 135, 138).

[Gfg, 2023a] Gfg (2023a). *decision-tree*. url: https://www.geeksforgeeks.org/decision-tree/ (cit. on p. 164).

[Gfg, 2023b] Gfg (2023b). *kaiming-initialization-in-deep-learning*. url: https://www.geeksforgeeks.org/kaiming-initialization-in-deep-learning/ (cit. on p. 179).

[Gfg, 2023c] Gfg (2023c). *vanishing-and-exploding-gradients-problems-in-deep-learning*. url: https://www.geeksforgeeks.org/vanishing-and-exploding-gradients-problems-in-deep-learning/ (cit. on p. 179).

[Gfg, 2024] Gfg (2024). *k-nearest-neighbours*. url: https://www.geeksforgeeks.org/k-nearest-neighbours/ (cit. on p. 163).

[K. Ginsburger, 2019] Ginsburger, Kévin (2019). *Modeling and simulation of the diffusion MRI signal from human brain white matter to decode its microstructure and produce an anatomic atlas at high fields (3T)*. url: https://www.theses.fr/2019SACLS158 (cit. on pp. 71, 72, 85, 93, 95–97, 137, 138, 164, 165).

[K. Ginsburger et al., 2018] Ginsburger, Kévin et al. (2018). Improving the realism of white matter numerical phantoms: a step toward a better understanding of the influence of structural disorders in diffusion MRI. *Frontiers in Physics* 6, p. 12 (cit. on pp. 68, 69, 82, 84).

[Ginsburger et al., 2019] Ginsburger et al. (2019). MEDUSA: A GPU-based tool to create realistic phantoms of the brain microstructure using tiny spheres. *NeuroImage* 193, pp. 10–24 (cit. on pp. 68, 70, 71, 78, 80, 84, 85, 87, 92, 93, 100).

[Giuliodori and DiCarlo, 2004] Giuliodori, Mauricio J and Stephen E DiCarlo (2004). Myelinated vs. unmyelinated nerve conduction: a novel way of understanding the mechanisms. *Advances in physiology education* 28.2, pp. 80–81 (cit. on p. 97).

[M. F. Glasser et al., 2016] Glasser, Matthew F et al. (2016). A multi-modal parcellation of human cerebral cortex. *Nature* 536.7615, pp. 171–178 (cit. on p. 23).

[Golgi, 1885] Golgi, Camillo (1885). *Sulla fina anatomia degli organi centrali del sistema nervoso*. S. Calderini (cit. on p. 19).

[Goodfellow, Bengio, and Courville, 2016a] Goodfellow, Bengio, and Courville (2016a). *Deep learning*. MIT press (cit. on p. 173).

[Goodfellow, Bengio, and Courville, 2016b] Goodfellow, Bengio, and Courville (2016b). *Deep learning*. MIT press (cit. on p. 178).

[Gore et al., 2010] Gore, John C et al. (2010). Characterization of tissue structure at varying length scales using temporal diffusion spectroscopy. *NMR in Biomedicine* 23.7, pp. 745–756 (cit. on p. 46).

[Grebenkov, 2007] Grebenkov, Denis S (2007). NMR survey of reflected Brownian motion. *Reviews of Modern Physics* 79.3, p. 1077 (cit. on p. 49).

[Grebenkov, 2011] Grebenkov, Denis S (2011). A fast random walk algorithm for computing the pulsed-gradient spin-echo signal in multiscale porous media. *Journal of Magnetic Resonance* 208.2, pp. 243–255 (cit. on p. 77).

[Gschmeissner, 2024] Gschmeissner, Steve (2024). *Axones art photography*. url: https://theworldcloseup.com (cit. on p. 3).

[Guise et al., 2016] Guise, Catarina et al. (2016). Hollow polypropylene yarns as a biomimetic brain phantom for the validation of high-definition fiber tractography imaging. *ACS applied materials & interfaces* 8.44, pp. 29960–29967 (cit. on pp. 61, 62).

[Gupta, 2017] Gupta, D. (2017). Limbic Lobe - an overview. *Essential in neuroanesthesia* (cit. on p. 10).

[Hagen and Henkelman, 2002] Hagen, Elisabeth AH von dem and R Mark Henkelman (2002). Orientational diffusion reflects fiber structure within a voxel. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 48.3, pp. 454–459 (cit. on p. 60).

[Häggqvist and al, 1936] Häggqvist, G. and al (1936). Analyse der faserverteilung in einem rückenmarkquerschnitt. *Z Mikrosk Anat Forsch* (cit. on p. 17).

[Hahn, 1950] Hahn, Erwin L (1950). Nuclear induction due to free Larmor precession. *Physical Review* 77.2, p. 297 (cit. on p. 30).

[Matt G Hall, G. Nedjati-Gilani, and Daniel C Alexander, 2017] Hall, Matt G, Gemma Nedjati-Gilani, and Daniel C Alexander (2017). Realistic voxel sizes and reduced signal variation in Monte-Carlo simulation for diffusion MR data synthesis. *arXiv preprint arXiv:1701.03634* (cit. on p. 78).

[Matt G. Hall and Daniel C. Alexander, 2009] Hall, Matt G. and Daniel C. Alexander (2009). Convergence and Parameter Choice for Monte-Carlo Simulations of Diffusion MRI. *IEEE Transactions on Medical Imaging* 28.9, pp. 1354–1364 (cit. on pp. 68–71, 75, 77–80, 117, 118, 125).

[Harkins and Does, 2016] Harkins, Kevin D and Mark D Does (2016). Simulations on the influence of myelin water in diffusion-weighted imaging. *Physics in Medicine & Biology* 61.13, p. 4729 (cit. on pp. 68, 70).

[A. W. He, 2017] He, Archy Will (2017). *torch.nn*. url: https://stackoverflow.com/questions/2480650/what-is-the-role-of-the-bias-in-neural-networks (cit. on p. 169).

[K. He et al., 2015] He, Kaiming et al. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, pp. 1026–1034 (cit. on p. 179).

[Healthliteracyhub, 2024] Healthliteracyhub (2024). *Dure mere expliquée*. url: https://healthliteracyhub.com/fr/anatomie-101-blog/dure-mere-expliquee (cit. on p. 6).

[S. Herculano-Houzel, 2014] Herculano-Houzel, S. (2014). The glia/neuron ratio: how it varies uniformly across brain structures and species and what that means for brain physiology and evolution. *Glia* 62.9, pp. 1377–1391 (cit. on p. 25).

[Herrero, Barcia, and Navarro, 2002] Herrero, M., C. Barcia, and J. Navarro (Sept. 2002). Functional anatomy of thalamus and basal ganglia. *Child's nervous system : ChNS : official journal of the International Society for Pediatric Neurosurgery* 18, pp. 386–404 (cit. on p. 13).

[Hopf, 1951] Hopf, Adolf (1951). *Die myeloarchitektonik des isocortex temporalis beim menschen* (cit. on p. 22).

[Hotcore, 2024] Hotcore (2024). *MRI global functioning*. url: https://hotcore.info/babki/magnetic-resonance-imaging-diagram.html (cit. on p. 29).

[Hubbard et al., 2015] Hubbard, Penny L et al. (2015). Biomimetic phantom for the validation of diffusion magnetic resonance imaging. *Magnetic resonance in medicine* 73.1, pp. 299–305 (cit. on pp. 61, 62).

[Hutson, 2018] Hutson, Matthew (2018). *Has artificial intelligence become alchemy?* (Cit. on p. 162).

[IBM, 2023] IBM (2023). *What are SVMs?* url: https://www.ibm.com/topics/support-vector-machine (cit. on p. 164).

[IDRISS, 2022] IDRISS (2022). *IDRISS*. url: http://www.idris.fr/jean-zay/jean-zay-presentation.html, (cit. on pp. 137, 195).

[IMAIOS, 2024] IMAIOS (2024). *IMAIOS*. url: https://www.imaios.com/fr/e-mri (cit. on p. 36).

[InstitutDeRadiologie, 2024] InstitutDeRadiologie (2024). *Brain vasculature, angiography*. url: https://www.institut-de-radiologie.com (cit. on p. 16).

[Ioffe and Szegedy, 2015] Ioffe, Sergey and Christian Szegedy (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International conference on machine learning*. pmlr, pp. 448–456 (cit. on p. 180).

[IRB, n.d.] IRB (n.d.). *IRB*. url: https://www.irbpty.com/dyneema-sk78-and-sk99-1 (cit. on p. 62).

[Jansons and Daniel C Alexander, 2003] Jansons, Kalvis M and Daniel C Alexander (2003). Persistent angular structure: new insights from diffusion magnetic resonance imaging data. *Inverse problems* 19.5, p. 1031 (cit. on p. 54).

[Jelescu et al., 2016] Jelescu, Ileana O et al. (2016). Degeneracy in model parameter estimation for multi-compartmental diffusion in neuronal tissue. *NMR in Biomedicine* 29.1, pp. 33–47 (cit. on pp. 65, 67).

[Jian and Vemuri, 2007] Jian, Bing and Baba C Vemuri (2007). "Multi-fiber reconstruction from diffusion MRI using mixture of Wisharts and sparse deconvolution". In: *Biennial international conference on information processing in medical imaging*. Springer, pp. 384–395 (cit. on p. 49).

[Johansen-Berg and T. E. Behrens, 2013] Johansen-Berg, Heidi and Timothy EJ Behrens (2013). *Diffusion MRI: from quantitative measurement to in vivo neuroanatomy*. Academic Press (cit. on p. 40).

[Jones, 2010] Jones, Derek K (2010). *Diffusion mri*. Oxford University Press (cit. on pp. 40, 42, 49, 54).

[Karbowski, 2011] Karbowski, Jan (2011). Scaling of brain metabolism and blood flow in relation to capillary and neural scaling. *PloS one* 6.10, e26709 (cit. on p. 22).

[Kärger and Heink, 1983] Kärger, Jörg and Wilfried Heink (1983). The propagator representation of molecular transport in microporous crystallites. *Journal of Magnetic Resonance (1969)* 51.1, pp. 1–7 (cit. on p. 50).

[Kastler and Vetter, 2018] Kastler, Bruno and Daniel Vetter (2018). *Comprendre l'IRM: manuel d'auto-apprentissage*. Elsevier Health Sciences (cit. on p. 38).

[Keras, 2024] Keras (2024). *Adam*. url: `https://keras.io/api/optimizers/adam/` (cit. on p. 179).

[Kerkelä et al., 2020] Kerkelä, Leevi et al. (2020). Disimpy: A massively parallel Monte Carlo simulator for generating diffusion-weighted MRI data in Python. *Journal of Open Source Software* 5.52, p. 2527 (cit. on pp. 68, 73).

[Kettenmann and Verkhratsky, 2008] Kettenmann, Helmut and Alexei Verkhratsky (2008). Neuroglia: the 150 years after. *Trends in neurosciences* 31.12, pp. 653–659 (cit. on p. 19).

[King et al., 1994] King, Martin D et al. (1994). q-Space imaging of the brain. *Magnetic resonance in medicine* 32.6, pp. 707–713 (cit. on pp. 50, 60).

[Kingma and Ba, 2014] Kingma, Diederik P and Jimmy Ba (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (cit. on p. 178).

[Kodiweera et al., 2016] Kodiweera, Chandana et al. (2016). Age effects and sex differences in human brain white matter of young to middle-aged adults: A DTI, NODDI, and q-space study. *Neuroimage* 128, pp. 180–192 (cit. on p. 65).

[Kölliker, 1889] Kölliker (1889). *Handbuch der gewebelehre des menschen* (cit. on p. 19).

[Kuchel, Lennon, and Durrant, 1996] Kuchel, Philip W, Alison J Lennon, and Christopher Durrant (1996). Analytical solutions and simulations for spin-echo measurements of diffusion of spins in a sphere with surface and bulk relaxation. *Journal of Magnetic Resonance, Series B* 112.1, pp. 1–17 (cit. on pp. 69, 75).

[Landman et al., 2010] Landman, Bennett A et al. (2010). Complex geometric models of diffusion and relaxation in healthy and damaged white matter. *NMR in Biomedicine: An International Journal Devoted to the Development and Application of Magnetic Resonance In vivo* 23.2, pp. 152–162 (cit. on pp. 71, 78, 80).

[Lauterbur, 1973] Lauterbur, Paul C (1973). Image formation by induced local interactions: examples employing nuclear magnetic resonance. *nature* 242.5394, pp. 190–191 (cit. on p. 31).

[Lazar et al., 2014] Lazar, M et al. (2014). *Axonal deficits in young adults with high functioning autism and their impact on processing speed. Neuroimage Clin 4, 417–425* (cit. on p. 59).

[Le, 2013] Le, Quoc V (2013). "Building high-level features using large scale unsupervised learning". In: *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, pp. 8595–8598 (cit. on p. 167).

[Le Bihan, 2003] Le Bihan, Denis (2003). Looking into the functional architecture of the brain with diffusion MRI. *Nature reviews neuroscience* 4.6, pp. 469–480 (cit. on p. 59).

[Le Bihan et al., 1986] Le Bihan, Denis et al. (1986). MR imaging of intravoxel incoherent motions: application to diffusion and perfusion in neurologic disorders. *Radiology* 161.2, pp. 401–407 (cit. on p. 47).

[LeCun et al., 1988] LeCun, Yann et al. (1988). "A theoretical framework for back-propagation". In: *Proceedings of the 1988 connectionist models summer school*. Vol. 1, pp. 21–28 (cit. on pp. 165, 170).

[Lee, Fieremans, and Novikov, 2021] Lee, Hong-Hsi, Els Fieremans, and Dmitry S Novikov (2021). Realistic Microstructure Simulator (RMS): Monte Carlo simulations of diffusion in three-dimensional cell segmentations of microscopy images. *Journal of neuroscience methods* 350, p. 109018 (cit. on pp. 73, 95).

[Lee, Jespersen, et al., 2020] Lee, Hong-Hsi, Sune N Jespersen, et al. (2020). The impact of realistic axonal shape on axon diameter estimation using diffusion MRI. *Neuroimage* 223, p. 117228 (cit. on pp. 71, 73, 74).

[Lee, Yaros, et al., 2018] Lee, Hong-Hsi, Katarina Yaros, et al. (2018). Electron microscopy 3-dimensional segmentation and quantification of axonal dispersion and diameter distribution in mouse brain corpus callosum. *bioRxiv*, p. 357491 (cit. on p. 92).

[Legendre, 1805] Legendre, Adrien Marie (1805). *Mémoire sur les opérations trigonométriques: dont les résultats dépendent de la figure de la terre*. 1. F. Didot (cit. on p. 163).

[Lenhossék, 1893] Lenhossék, Mihály (1893). *Der feinere Bau des Nervensystems im Lichte neuester Forschungen*. Fischer (cit. on p. 19).

[J.-R. Li, Calhoun, et al., 2013] Li, Jing-Rebecca, Donna Calhoun, et al. (2013). Numerical simulation of diffusion MRI signals using an adaptive time-stepping method. *Physics in Medicine & Biology* 59.2, p. 441 (cit. on p. 80).

[J.-R. Li, V.-D. Nguyen, et al., 2019] Li, Jing-Rebecca, Van-Dang Nguyen, et al. (2019). SpinDoctor: A MATLAB toolbox for diffusion MRI simulation. *NeuroImage* 202, p. 116120 (cit. on pp. 68, 70, 73, 74).

[Liewald et al., 2014] Liewald, D. et al. (2014). Distri- bution of axon diameters in cortical white matter: an electron-microscopic study on three human brains and a macaque. *Biological cybernetics* (cit. on pp. 17, 25).

[C.-P. Lin et al., 2003] Lin, Ching-Po et al. (2003). Validation of diffusion spectrum magnetic resonance imaging with manganese-enhanced rat optic tracts and ex vivo phantoms. *Neuroimage* 19.3, pp. 482–495 (cit. on p. 60).

[M. Lin et al., 2016] Lin, Mu et al. (2016). Simulation of changes in diffusion related to different pathologies at cellular level after traumatic brain injury. *Magnetic resonance in medicine* 76.1, pp. 290–300 (cit. on pp. 69, 71).

[Lipinski, 1990] Lipinski, Hans-G (1990). Monte Carlo simulation of extracellular diffusion in brain tissues. *Physics in Medicine & Biology* 35.3, p. 441 (cit. on pp. 75, 77).

[Liu et al., 2004] Liu, Chunlei et al. (2004). Characterizing non-Gaussian diffusion by using generalized diffusion tensors. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 51.5, pp. 924–937 (cit. on pp. 68, 69, 75, 77).

[Lundgaard et al., 2014] Lundgaard, I. et al. (2014). White matter astrocytes in health and disease. *Neuroscience,* 276, pp. 161–173 (cit. on p. 19).

[MacWilliams, 2003] MacWilliams, Bryon (2003). Russian claims first in magnetic imaging. *Nature* 426.6965, pp. 375–376 (cit. on pp. 30, 31).

[Mansfield, 1977] Mansfield, Peter (1977). Multi-planar image formation using NMR spin echoes. *Journal of Physics C: Solid State Physics* 10.3, p. L55 (cit. on p. 31).

[Marco Palombo, 2019] Marco Palombo Daniel Alexander, Hui Zhang (2019). A generative model of realistic brain cells with application to numerical simulation of the diffusion-weighted MR signal. *NeuroImage* 188, pp. 391–402 (cit. on pp. 71, 73, 74, 95).

[Mardia and Jupp, 2009] Mardia, Kanti V and Peter E Jupp (2009). *Directional statistics*. John Wiley & Sons (cit. on p. 64).

[Maubon et al., 1999] Maubon, Antoine J et al. (1999). Effect of field strength on MR images: comparison of the same subject at 0.5, 1.0, and 1.5 T. *Radiographics* 19.4, pp. 1057–1067 (cit. on p. 31).

[McCall, Douglass, and Anderson, 1963] McCall, David W, Dean C Douglass, and Ernest W Anderson (1963). Self-diffusion studies by means of nuclear magnetic resonance spin-echo techniques. *Berichte der Bunsengesellschaft für physikalische Chemie* 67.3, pp. 336–340 (cit. on p. 43).

[McCulloch and Pitts, 1943] McCulloch, Warren S and Walter Pitts (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics* 5, pp. 115–133 (cit. on p. 165).

[McGraw et al., 2006] McGraw, Tim et al. (2006). "Von Mises-Fisher mixture model of the diffusion ODF". In: *3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2006.* IEEE, pp. 65–68 (cit. on p. 49).

[Meier, Dreher, and Leibfritz, 2003] Meier, Christian, Wolfgang Dreher, and Dieter Leibfritz (2003). Diffusion in compartmental systems. II. Diffusion-weighted measurements of rat brain tissue in vivo and postmortem at very large b-values. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 50.3, pp. 510–514 (cit. on p. 60).

[Meynert, 1885] Meynert, T. (1885). *Psychiatry: A Clinical Treatise on Diseases of the Fore-brain...* (Cit. on p. 14).

[Minsky and Papert, 1969] Minsky, Marvin and Seymour Papert (1969). Perceptrons cambridge. *MA: MIT Press. zbMATH* (cit. on p. 165).

[Mitra and Halperin, 1995] Mitra, Partha P and Bertrand I Halperin (1995). Effects of finite gradient-pulse widths in pulsed-field-gradient diffusion measurements. *Journal of Magnetic Resonance, Series A* 113.1, pp. 94–101 (cit. on p. 51).

[Mollink et al., 2017] Mollink, Jeroen et al. (2017). Evaluating fibre orientation dispersion in white matter: Comparison of diffusion MRI, histology and polarized light imaging. *Neuroimage* 157, pp. 561–574 (cit. on p. 64).

[Mori et al., 1999] Mori, Susumu et al. (1999). Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Annals of Neurology: Official Journal of the American Neurological Association and the Child Neurology Society* 45.2, pp. 265–269 (cit. on p. 58).

[Mottershead et al., 2003] Mottershead, J. P. et al. (2003). High field MRI correlates of myelincontent and axonal density in multiple sclerosis. *Journal of Neurology* 250.11, pp. 1293–1301 (cit. on pp. 25, 140).

[Moussavi-Biugui et al., 2011] Moussavi-Biugui, Amir et al. (2011). Novel spherical phantoms for Q-ball imaging under in vivo conditions. *Magnetic resonance in medicine* 65.1, pp. 190–194 (cit. on p. 60).

[Mozumder et al., 2019] Mozumder, Meghdoot et al. (2019). Population-based Bayesian regularization for microstructural diffusion MRI with NODDIDA. *Magnetic resonance in medicine* 82.4, pp. 1553–1565 (cit. on p. 65).

[Mulkern et al., 1999] Mulkern, Robert V et al. (1999). Multi-component apparent diffusion coefficients in human brain. *NMR in Biomedicine: An International Journal Devoted to the Development and Application of Magnetic Resonance In Vivo* 12.1, pp. 51–62 (cit. on p. 60).

[Murtagh, 1991] Murtagh, Fionn (1991). Multilayer perceptrons for classification and regression. *Neurocomputing* 2.5-6, pp. 183–197 (cit. on p. 167).

[Nassif, Ho, and Capretz, 2013] Nassif, Ali Bou, Danny Ho, and Luiz Fernando Capretz (2013). Towards an early software estimation using log-linear regression and a multilayer perceptron model. *Journal of Systems and Software* 86.1, pp. 144–160 (cit. on p. 167).

[Nedjati-Gilani, 2017] Nedjati-Gilani (2017). Machine learning based compartment models with permeability for white matter microstructure imaging. *NeuroImage* 150, pp. 119–135 (cit. on pp. 81, 164).

[Neuman, 1974] Neuman, CH (1974). Spin echo of spins diffusing in a bounded medium. *The Journal of Chemical Physics* 60.11, pp. 4508–4511 (cit. on pp. 49, 66, 80).

[Neupsykey, 2024] Neupsykey (2024). *Brain vasculature*. url: `https://neupsykey.com` (cit. on p. 16).

[D. V. Nguyen et al., 2014] Nguyen, Dang Van et al. (2014). A finite elements method to solve the Bloch–Torrey equation applied to diffusion magnetic resonance imaging. *Journal of Computational Physics* 263, pp. 283–302 (cit. on p. 69).

[K.-V. Nguyen et al., 2018] Nguyen, Khieu-Van et al. (2018). Efficient GPU-based Monte-Carlo simulation of diffusion in real astrocytes reconstructed from confocal microscopy. *Journal of Magnetic Resonance* 296, pp. 188–199 (cit. on pp. 73, 74).

[Niendorf et al., 1996] Niendorf, Thoralf et al. (1996). Biexponential diffusion attenuation in various states of brain tissue: implications for diffusion-weighted imaging. *Magnetic Resonance in Medicine* 36.6, pp. 847–857 (cit. on p. 60).

[Nieuwenhuys, 2013] Nieuwenhuys, Rudolf (2013). The myeloarchitectonic studies on the human cerebral cortex of the Vogt-Vogt school, and their significance for the interpretation of functional neuroimaging data. *Microstructural Parcellation of the Human Cerebral Cortex: From Brodmann's Post-Mortem Map to in Vivo Mapping with High-Field Magnetic Resonance Imaging*, pp. 55–125 (cit. on pp. 22, 24, 26).

[Nieuwenhuys and Broere, 2020] Nieuwenhuys, Rudolf and Cees AJ Broere (2020). A detailed comparison of the cytoarchitectonic and myeloarchitectonic maps of the human neocortex produced by the Vogt–Vogt school. *Brain Structure and Function* 225.9, pp. 2717–2733 (cit. on p. 22).

[Nieuwenhuys, Broere, and Cerliani, 2015a] Nieuwenhuys, Rudolf, Cees AJ Broere, and Leonardo Cerliani (2015a). A new myeloarchitectonic map of the human neocortex based on data from the Vogt–Vogt school. *Brain Structure and Function* 220, pp. 2551–2573 (cit. on p. 22).

[Nieuwenhuys, Broere, and Cerliani, 2015b] Nieuwenhuys, Rudolf, Cees AJ Broere, and Leonardo Cerliani (2015b). A new myeloarchitectonic map of the human neocortex based on data from the Vogt–Vogt school. *Brain Structure and Function* 220, pp. 2551–2573 (cit. on p. 22).

[M. Nilsson et al., 2010] Nilsson, M. et al. (2010). A mechanism for exchange between intraaxonal and extracellular water: Permeable nodes of Ranvier (cit. on pp. 80, 81).

[Markus Nilsson et al., 2017] Nilsson, Markus et al. (2017). Resolution limit of cylinder diameter estimation by diffusion MRI: The impact of gradient waveform and orientation dispersion. *NMR in Biomedicine* 30.7, e3711 (cit. on p. 64).

[Nissl, 1904] Nissl, F. (1904). *Zur Histopathologie der paralytischen Rindenerkrankung.* (Cit. on pp. 22, 23).

[Novikov, Fieremans, et al., 2019] Novikov, Dmitry S, Els Fieremans, et al. (2019). Quantifying brain microstructure with diffusion MRI: Theory and parameter estimation. *NMR in Biomedicine* 32.4, e3998 (cit. on p. 151).

[Novikov, Veraart, et al., 2018] Novikov, Dmitry S, Jelle Veraart, et al. (2018). Rotationally-invariant mapping of scalar and orientational metrics of neuronal microstructure with diffusion MRI. *NeuroImage* 174, pp. 518–538 (cit. on p. 65).

[Nowogrodzki, 2018] Nowogrodzki, Anna (2018). The world's strongest MRI machines are pushing human imaging to new limits. *Nature* 563.7732, pp. 24–27 (cit. on p. 31).

[N. Oberheim et al., 2009] Oberheim, N.A. et al. (2009). Uniquely Hominid Features of Adult Human Astrocytes. *Journal of Neuroscience* 29.6, pp. 3276–3287 (cit. on pp. 19, 97).

[N. A. Oberheim, Goldman, and Nedergaard, 2012] Oberheim, Nancy Ann, Steven A Goldman, and Maiken Nedergaard (2012). Heterogeneity of astrocytic form and function. *Astrocytes: Methods and protocols*, pp. 23–45 (cit. on p. 19).

[Olaf, 2002] Olaf, Dietrich (2002). *Diffusion Coefficients of Water*. url: https://dtrx.de/od/diff (cit. on p. 117).

[openai, 2024] openai (2024). *dall-e*. url: https://openai.com/dall-e-3 (cit. on p. 161).

[OpenMP, 2024] OpenMP (2024). *OpenMP*. url: https://www.openmp.org/ (cit. on p. 137).

[Oracle, 2021] Oracle (2021). *The Java™ Tutorials: Learning the Java Language: Interfaces and Inheritance*. url: https://docs.oracle.com/javase/tutorial/java/IandI/polymorphism.html (cit. on p. 86).

[Özarslan, Koay, and Basser, 2013] Özarslan, Evren, Cheng Guan Koay, and Peter J Basser (2013). Simple harmonic oscillator based reconstruction and estimation for one-dimensional q-space magnetic resonance (1D-SHORE). *Excursions in Harmonic Analysis, Volume 2: The February Fourier Talks at the Norbert Wiener Center*, pp. 373–399 (cit. on p. 151).

[Özarslan, Shepherd, et al., 2005] Özarslan, Evren, Timothy M Shepherd, et al. (2005). Fast orientation mapping from HARDI, pp. 156–163 (cit. on p. 54).

[Packard, 1946] Packard, F Bloch WW Hansen M (1946). The nuclear induction experiment. *Physical Review* 70.7-8, pp. 474–485 (cit. on p. 30).

[Paladino, 1892] Paladino, G. (1892). De la continuation de la névroglie dans le squelette myelinique des fibres nerveuses et de la constitution pluricellulaire du cylindraxe, pp. 26–32 (cit. on p. 20).

[Palombo, 2020] Palombo (2020). SANDI: a compartment-based model for non-invasive apparent soma and neurite imaging by diffusion MRI. *Neuroimage* 215, p. 116835 (cit. on pp. 66, 81).

[M. Palombo et al., 2018] Palombo, Marco et al. (2018). Can we detect the effect of spines and leaflets on the diffusion of brain intracellular metabolites? *Neuroimage* 182, pp. 283–293 (cit. on pp. 71, 72).

[Panagiotaki et al., 2010] Panagiotaki, Eleftheria et al. (2010). "High-fidelity meshes from tissue samples for diffusion MRI simulations". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010: 13th International Conference, Beijing, China, September 20-24, 2010, Proceedings, Part II 13*. Springer, pp. 404–411 (cit. on p. 73).

[Patel, 2022] Patel, Devanshi (2022). *data-normalization-techniques*. url: https://medium.com/codex/data-normalization-techniques-4148b69876b0 (cit. on p. 180).

[PEACOCK, 2024] PEACOCK (2024). *PEACOCK*. url: https://www-hpc.cea.fr/tgcc-public/en/html/toc/fulldoc/Virtualization.html (cit. on p. 140).

[Pedergnana, García, et al., 2016] Pedergnana, Mattia, Sebastián Gimeno García, et al. (2016). Smart sampling and incremental function learning for very large high dimensional data. *Neural Networks* 78, pp. 75–87 (cit. on p. 81).

[Perge et al., 2009] Perge, J. A. et al. (2009). How the Optic Nerve Allocates Space, Energy Capacity, and Information. *Journal of Neuroscience* 29.24. Cited on page 25, pp. 7917–7928 (cit. on pp. 25, 140).

[Perrin et al., 2005] Perrin, Muriel et al. (2005). Validation of q-ball imaging with a diffusion fibre-crossing phantom on a clinical scanner. *Philosophical Transactions of the Royal Society B: Biological Sciences* 360.1457, pp. 881–891 (cit. on p. 53).

[Pierpaoli and Basser, 1996] Pierpaoli, Carlo and Peter J Basser (1996). Toward a quantitative assessment of diffusion anisotropy. *Magnetic resonance in Medicine* 36.6, pp. 893–906 (cit. on p. 58).

[Postel et al., 2019] Postel, Charlotte et al. (2019). Hippocampal subfields alterations in adolescents with post-traumatic stress disorder. *Human Brain Mapping* 40.4, pp. 1244–1252 (cit. on p. 31).

[Poupon, 1999] Poupon, Cyril (1999). “Détection des faisceaux de fibres de la substance blanche pour l’étude de la connectivité anatomique cérébrale”. PhD thesis. Paris, ENST (cit. on pp. 58, 59).

[Poupon et al., 2008] Poupon, Cyril et al. (2008). New diffusion phantoms dedicated to the study and validation of high-angular-resolution diffusion imaging (HARDI) models. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 60.6, pp. 1276–1283 (cit. on p. 60).

[PRACE, 2024] PRACE (2024). *PRACE*. url: https://prace-ri.eu/ (cit. on p. 138).

[Puiseux et al., 2021] Puiseux, Thomas et al. (2021). Numerical simulation of time-resolved 3D phase-contrast magnetic resonance imaging. *PLoS One* 16.3, e0248816 (cit. on p. 34).

[E. M. Purcell, Henry Cutler Torrey, and Pound, 1946] Purcell, Edward M, Henry Cutler Torrey, and Robert V Pound (1946). Resonance absorption by nuclear magnetic moments in a solid. *Physical review* 69.1-2, p. 37 (cit. on p. 30).

[Pytorch, 2024] Pytorch (2024). *torch.nn*. url: https://pytorch.org/docs/stable/nn.html (cit. on pp. 169, 174).

[Quizlet, 2024] Quizlet (2024). *Ventricule from the human brain*. url: https://quizlet.com/239148256 (cit. on p. 7).

[Rabi et al., 1938] Rabi, Isidor Isaac et al. (1938). A new method of measuring nuclear magnetic moment. *Physical review* 53.4, p. 318 (cit. on p. 30).

[Radiopaedia, 2005-2024] Radiopaedia (2005-2024). *Radiopaedia*. url: https://radiopaedia.org/articles/apparent-diffusion-coefficient-1 (cit. on p. 153).

[Rafael-Patino et al., 2020] Rafael-Patino et al. (2020). Robust Monte-Carlo Simulations in Diffusion-MRI: Effect of the Substrate Complexity and Parameter Choice on the Reproducibility of Results. *Frontiers in neuroinformatics* 14.8 (cit. on pp. 118, 119).

[Rio-Hortega, 1928] Rio-Hortega, P.del (1928). Tercera aportacion al conocimiento morfologico e interpretacion funcional de la oligodendroglia (cit. on pp. 19, 21).

[Robotics, 2020] Robotics, Sally (2020). *Backpropagation and its Alternatives*. url: https://medium.com/@sallyrobotics.blog/backpropagation-and-its-alternatives-c09d306aae4c (cit. on p. 170).

[Ronen et al., 2014] Ronen, I. et al. (2014). Microstructural organization of axons in the human corpus cal- losum quantified by diffusion-weighted magnetic resonance spectroscopy of N- acetylaspartate and post-mortem histology. *Brain Structure and Function* 219.5. Cited on page 25, 64, 163 and 164, pp. 1773–1785 (cit. on pp. 27, 64).

[Rosenblatt, 1958] Rosenblatt, Frank (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* 65.6, p. 386 (cit. on p. 165).

[Rosenkrantz et al., 2015] Rosenkrantz, Andrew B et al. (2015). Body diffusion kurtosis imaging: basic principles, applications, and considerations for clinical practice. *Journal of Magnetic Resonance Imaging* 42.5, pp. 1190–1202 (cit. on p. 45).

[Ruck, Rogers, and Kabrisky, 1990] Ruck, Dennis W, Steven K Rogers, and Matthew Kabrisky (1990). Feature selection using a multilayer perceptron. *Journal of neural network computing* 2.2, pp. 40–48 (cit. on p. 168).

[Rumelhart, Hinton, and Williams, 1986] Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams (1986). Learning representations by back-propagating errors. *nature* 323.6088, pp. 533–536 (cit. on pp. 165, 170).

[Rushton, 1951] Rushton, W. A. H. (1951). A theory of the effects of fibre size in medullated nerve. *The Journal of Physiology* 115.1, pp. 101–122 (cit. on p. 17).

[Salzer, 1997] Salzer, J.L. (1997). Clustering Sodium Channels at the Node of Ranvier: Close Encounters of the Axon-Glia Kind. *Neuron* 18.6. Cited on page 24, pp. 843–846 (cit. on pp. 17, 22).

[Samuel, 1959] Samuel, Arthur L (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development* 3.3, pp. 210–229 (cit. on p. 163).

[Sanides, 2013] Sanides, Friedrich (2013). *Die Architektonik des menschlichen Stirnhirns: zugleich eine Darstellung der Prinzipien seiner Gestaltung als Spiegel der stammesgeschichtlichen Differenzierung der Grosshirnrinde*. Vol. 98. Springer-Verlag (cit. on p. 22).

[Santurkar et al., 2018] Santurkar, Shibani et al. (2018). How does batch normalization help optimization? *Advances in neural information processing systems* 31 (cit. on p. 180).

[Sarkisov, Filimonoff, and Preobrashenskaya, 1949] Sarkisov, SA, IN Filimonoff, and NS Preobrashenskaya (1949). *Cytoarchitecture of the human cortex cerebri (Russ.) Medgiz* (cit. on p. 23).

[Sarnat, 1992] Sarnat, Harvey B (1992). Role of human fetal ependyma. *Pediatric neurology* 8.3, pp. 163–178 (cit. on p. 22).

[Schmolesky, 2016] Schmolesky, Matthew (2016). *The Primary Visual Cortex by Matthew Schmolesky* (cit. on p. 95).

[Schüz and Braitenberg, 2002] Schüz, A. and V. Braitenberg (2002). The human cortical white matter: quantitative aspects of cortico-cortical longrange connectivity. *Cortical areas: Unity and diversity* 21, pp. 377–385 (cit. on p. 17).

[ScienceSparks, 2024] ScienceSparks (2024). *Diffusion phenomenon*. url: https://www.science-sparks.com/ (cit. on p. 41).

[ScikitLearn, 2024] ScikitLearn (2024). $permutation_importance$. url: https://scikit-learn.org/stable/modules/permutation_importance.html (cit. on p. 192).

[Shapson-Coe et al., 2021] Shapson-Coe, Alexander et al. (2021). A connectomic study of a petascale fragment of human cerebral cortex. *BioRxiv*, pp. 2021–05 (cit. on p. 114).

[ShutterStock, 2024] ShutterStock (2024). *Central nervous system*. url: https://www.shutterstock.com/search/cns?image_type=vector (cit. on p. 5).

[stackexchange, 2016] stackexchange (2016). *how-can-neural-networks-deal-with-varying-input-sizes*. url: https://ai.stackexchange.com/questions/2008/how-can-neural-networks-deal-with-varying-input-sizes (cit. on p. 195).

[Staff, 2014] Staff, B. (2014). Medical gallery of Blausen Medical 2014. *WikiJournal of Medicine* 1.2 (cit. on pp. 7, 9).

[Stanisz et al., 1997] Stanisz, Greg J et al. (1997). An analytical model of restricted diffusion in bovine optic nerve. *Magnetic Resonance in Medicine* 37.1, pp. 103–111 (cit. on pp. 61, 66).

[Stejskal and Tanner, 1965] Stejskal, Edward O and John E Tanner (1965). Spin diffusion measurements: spin echoes in the presence of a time-dependent field gradient. *The journal of chemical physics* 42.1, pp. 288–292 (cit. on pp. 43, 50).

[Stikov et al., 2015] Stikov, N. et al. (2015). In vivo histology of the myelin g- ratio with magnetic resonance imaging. *NeuroImage* 118, pp. 397–405 (cit. on pp. 17, 25, 140).

[Strasburger, 1937] Strasburger, Eduard H (1937). Die myeloarchitektonische gliederung des stirnhirns beim menschen und schimpansen. *J. Psychol. Neurol* 47, pp. 565–606 (cit. on p. 22).

[Su et al., 2019] Su, Jason H et al. (2019). Thalamus Optimized Multi Atlas Segmentation (THOMAS): fast, fully automated segmentation of thalamic nuclei from structural MRI. *Neuroimage* 194, pp. 272–282 (cit. on p. 31).

[Sweeney, Sagare, and Zlokovic, 2015] Sweeney, Melanie D, Abhay P Sagare, and Berislav V Zlokovic (2015). Cerebrospinal fluid biomarkers of neurovascular dysfunction in mild dementia and Alzheimer's disease. *Journal of Cerebral Blood Flow & Metabolism* 35.7, pp. 1055–1068 (cit. on p. 22).

[Szafer, Zhong, and Gore, 1995] Szafer, Aaron, Jianhui Zhong, and John C Gore (1995). Theoretical model for water diffusion in tissues. *Magnetic resonance in medicine* 33.5, pp. 697–712 (cit. on pp. 66, 68, 75).

[Taigman et al., 2014] Taigman, Yaniv et al. (2014). Deepface: Closing the gap to human-level performance in face verification, pp. 1701–1708 (cit. on p. 167).

[Tariq et al., 2016] Tariq, Maira et al. (2016). Bingham–NODDI: Mapping anisotropic orientation dispersion of neurites using diffusion MRI. *Neuroimage* 133, pp. 207–223 (cit. on p. 65).

[Thirunavukarasu et al., 2023] Thirunavukarasu, Arun James et al. (2023). Large language models in medicine. *Nature medicine* 29.8, pp. 1930–1940 (cit. on p. 170).

[Tian et al., 2022] Tian, Qiyuan et al. (2022). Comprehensive diffusion MRI dataset for in vivo human brain microstructure mapping using 300 mT/m gradients. *Scientific Data* 9.1, p. 7 (cit. on p. 81).

[Tomassy et al., 2014] Tomassy, Giulio Srubek et al. (2014). Distinct profiles of myelin distribution along single axons of pyramidal neurons in the neocortex. *Science* 344.6181, pp. 319–324 (cit. on p. 17).

[Henry C Torrey, 1956] Torrey, Henry C (1956). Bloch equations with diffusion terms. *Physical review* 104.3, p. 563 (cit. on pp. 80, 106).

[Tournier et al., 2008] Tournier, J-Donald et al. (2008). Resolving crossing fibres using constrained spherical deconvolution: validation using diffusion-weighted imaging phantom data. *Neuroimage* 42.2, pp. 617–625 (cit. on pp. 54, 60).

[towardsdatascience, 2024] towardsdatascience (2024). *Dropout*. url: https://towardsdatascience.com/dropout-in-neural-networks-47a162d621d9 (cit. on p. 179).

[Traiffort et al., 2020] Traiffort, Elisabeth et al. (2020). Astrocytes and Microglia as Major Players of Myelin Production in Normal and Pathological Conditions. *Frontiers in Cellular Neuroscience* 14 (cit. on p. 20).

[Tremblay et al., 2011] Tremblay, P et al. (2011). Brain mapping: From neural basis of cognition to surgical applications (cit. on p. 23).

[Trott et al., 2022] Trott, Christian R. et al. (2022). Kokkos 3: Programming Model Extensions for the Exascale Era. *IEEE Transactions on Parallel and Distributed Systems* 33.4, pp. 805–817 (cit. on p. 137).

[Tuch et al., 2002] Tuch et al. (2002). "Diffusion MRI of complex tissue structure". PhD thesis. Massachusetts Institute of Technology (cit. on pp. 49, 51).

[Tuch et al., 2003] Tuch et al. (2003). Diffusion MRI of complex neural architecture. *Neuron* 40.5, pp. 885–895 (cit. on p. 51).

[D. S. Tuch, 2004] Tuch, David S (2004). Q-ball imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 52.6, pp. 1358–1372 (cit. on p. 53).

[Turing, 2024] Turing (2024). *What Is the Necessity of Bias in Neural Networks?* url: https://www.turing.com/kb/necessity-of-bias-in-neural-networks#what-is-bias-in-a-neural-network? (cit. on p. 169).

[Unil, 2024] Unil (2024). *IMAGERIE PAR RÉSONANCE MAGNÉTIQUE*. url: https://wp.unil.ch/cvmr/files/2021/01/2021_FMH_IRM_Lecon_1.pdf (cit. on p. 38).

[Van Nguyen et al., 2014] Van Nguyen, Dang et al. (2014). A finite elements method to solve the Bloch–Torrey equation applied to diffusion magnetic resonance imaging. *Journal of Computational Physics* 263, pp. 283–302 (cit. on pp. 47, 80).

[Vapnik, Golowich, and Smola, 1996] Vapnik, Vladimir, Steven Golowich, and Alex Smola (1996). Support vector method for function approximation, regression estimation and signal processing. *Advances in neural information processing systems* 9 (cit. on p. 164).

[Vaswani et al., 2017] Vaswani, Ashish et al. (2017). Attention is all you need. *Advances in neural information processing systems* 30 (cit. on pp. 167, 195).

[Verkhratsky and Kirchhoff, 2007] Verkhratsky, Alexei and Frank Kirchhoff (2007). NMDA receptors in glia. *The Neuroscientist* 13.1, pp. 28–37 (cit. on p. 20).

[Viard, Eustache, and Segobin, 2021] Viard, Armelle, Francis Eustache, and Shailendra Segobin (2021). History of Magnetic Resonance Imaging: A Trip Down Memory Lane. *Neuroscience* 474. Brain imaging, pp. 3–13 (cit. on pp. 30, 31).

[Villarreal-Haro et al., 2023] Villarreal-Haro, Juan Luis et al. (2023). CACTUS: a computational framework for generating realistic white matter microstructure substrates. *Frontiers in Neuroinformatics* 17, p. 1208073 (cit. on pp. 68, 70–72, 75).

[Vinod, 2020] Vinod, Robin (2020). *batch-normalisation-explained*. url: https://towardsdatascience.com/batch-normalisation-explained-5f4bd9de5feb (cit. on p. 180).

[Virchow, 1858] Virchow, Rudolf (1858). *Die Cellularpathologie in ihrer Begründung auf physiologische und pathologische Gewebelehre*. Good Press (cit. on p. 19).

[Vogt, 1910] Vogt, Oskar (1910). Die Myeloarchitektonische Felderung Des Menschlichen. *Journal für Psychologie und Neurologie* 15.4/5, pp. 221–232 (cit. on p. 22).

[Von Bartheld, Bahney, and Suzana Herculano-Houzel, 2016] Von Bartheld, Christopher S, Jami Bahney, and Suzana Herculano-Houzel (2016). The search for true numbers of neurons and glial cells in the human brain: A review of 150 years of cell counting. *Journal of Comparative Neurology* 524.18, pp. 3865–3895 (cit. on p. 16).

[Voříšek et al., 2002] Voříšek, I et al. (2002). Water ADC, extracellular space volume, and tortuosity in the rat cortex after traumatic injury. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 48.6, pp. 994–1003 (cit. on pp. 25, 75).

[Wedeen et al., 2005] Wedeen, Van J et al. (2005). Mapping complex tissue architecture with diffusion spectrum magnetic resonance imaging. *Magnetic resonance in medicine* 54.6, pp. 1377–1386 (cit. on pp. 50, 52).

[Wikimedia, 2020] Wikimedia (2020). *Human skull*. url: https://commons.wikimedia.org/wiki/File:Human_skull-bg.svg (cit. on pp. 6, 29).

[Xing et al., 2013] Xing et al. (2013). Investigation of Different Boundary Treatment Methods in Monte-Carlo Simulations of Diffusion NMR. *Magnetic Resonance in Medecine* 70.4, pp. 1167–1172 (cit. on pp. 76–79, 117, 118, 125, 126).

[J. Xu, Does, and Gore, 2009] Xu, Junzhong, Mark D Does, and John C Gore (2009). Sensitivity of MR diffusion measurements to variations in intracellular structure: effects of nuclear size. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 61.4, pp. 828–833 (cit. on p. 106).

[T. Xu et al., 2018] Xu, Tianyou et al. (2018). The effect of realistic geometries on the susceptibility-weighted MR signal in white matter. *Magnetic resonance in medicine* 79.1, pp. 489–500 (cit. on pp. 73, 74).

[Yeh et al., 2013] Yeh, Chun-Hung et al. (2013). Diffusion microscopist simulator: a general Monte Carlo simulation system for diffusion magnetic resonance imaging. *PloS one* 8.10, e76626 (cit. on pp. 68–71, 78, 84, 106).

[Zhang, 2011] Zhang (2011). Axon diameter mapping in the presence of orientation dispersion with diffusion MRI. *Neuroimage* 56.3, pp. 1301–1315 (cit. on p. 64).

[Zhang, 2012] Zhang (2012). NODDI: practical in vivo neurite orientation dispersion and density imaging of the human brain. *Neuroimage* 61.4, pp. 1000–1016 (cit. on pp. 61, 64–66).

[Zucchelli, Descoteaux, and Menegaz, 2017] Zucchelli, Mauro, Maxime Descoteaux, and Gloria Menegaz (2017). NODDI-SH: a computational efficient NODDI extension for fODF estimation in diffusion MRI. *arXiv preprint arXiv:1708.08999* (cit. on p. 65).