

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 LE PROBLÈME DIRECT ENTRE LES SOURCES ET LES CAPTEURS ...	11
1.1 Introduction.....	11
1.2 Espace des sources.....	11
1.2.1 Définition d'un dipôle.....	11
1.2.2 Modèle dipolaire et Modèle distribué.....	13
1.3 Mesures.....	13
1.4 Matrice de gain	15
1.5 OpenMEEG pour la modélisation du problème direct	17
1.6 Un peu de théorie sur l'approche de OpenMEEG dans la modélisation du problème direct.	18
1.6.1 Matrice de Gain EEG.....	21
1.6.2 Matrice de Gain MEG.....	21
1.7 Le logiciel OpenMEEG	22
1.8 Description des répertoires de données et des types de fichiers de données	23
1.8.1 Contenu du répertoire HeadModels.....	23
1.8.2 Contenu du répertoire 'Computations'	27
1.8.3 Contenu du répertoire 'IntermediateFiles'.....	28
1.9 Maillages et position des électrodes.	28
1.10 Résumé sur OpenMEEG.....	29
1.11 Conclusion sur le chapitre 1.....	30
CHAPITRE 2 LES LOGICIELS BRAINVISA ET BRAINSUITE.....	31
2.1 Introduction.....	31
2.2 Présentation du logiciel BrainVISA.....	32
2.3 Mise en contexte	33
2.4 Utilisation de la librairie "Création d'un modèle EEG"	34
2.4.1 Étape 0 : Préparation des données	35
2.4.2 Étape 1 : Segmentation de la surface corticale	35
2.4.3 Étape 2 : Le recalage des électrodes	36
2.4.4 Étape 3 : Modèle distribué, suppression d'électrodes, matrice de Gain et génération de fichiers pour BrainEntropy.....	39

2.5	Résumé de l'utilisation de BrainVISA	44
2.6	BrainSuite2	46
2.6.1	Présentation du logiciel BrainSuite2.....	46
2.6.2	Lecture et visualisation du volume IRM.....	46
2.6.3	Orientation du volume	47
2.6.4	Segmentation de surfaces.....	47
2.7	Conclusion sur le chapitre 2.....	53
CHAPITRE 3 PROBLÈME INVERSE ET LES MÉTHODES FOCCUS		54
3.1	Complémentarité EEG et MEG	54
3.2	La Formulation Bayésienne du problème inverse	56
3.3	Résolution linéaire	59
3.4	Résolution non linéaire	60
3.4.1	Minimisation de la norme L_p	61
3.4.2	Entropie de Shannon.....	64
3.4.3	Convergence de l'entropie de Shannon	64
3.5	Conclusion sur le chapitre 3.....	66
CHAPITRE 4 BRAINENTROPY : UN OUTIL DE SIMULATION ET DE RÉSOLUTION DU PROBLÈME INVERSE EN IMAGERIE EEG - MEG.....		67
4.1	Préparation des données : Maillages et recalage des électrodes.....	68
4.2	Création d'un Modèle	73
4.3	Description des classes et des variables principales	75
4.4	Simulation d'activité cérébrale et validation des matrices de gain.....	78
4.5	Exemples de visualisation.....	80
4.6	Est-il judicieux d'utiliser le problème inverse à partir de données simulées ?.....	83
4.7	Analyse ROC	84
4.8	Comparaison et quantification de l'efficacité de différentes méthodes de résolution du problème inverse.....	86
4.9	Application avec des données réelles	87
4.10	Conclusion sur le chapitre 4.....	93
CONCLUSION.....		95
ANNEXE I INSTALLATION DE OPENMEEG.....		99
4.11	Installation sous Linux.....	99
4.12	Installation sous windows.....	101
4.13	3 Commandes pour Linux.....	106
ANNEXE II INSTALLATION DE BRAINVISA		108
4.14	1 Installation et configuration de BrainVISA	108

ANNEXE III CLASSES ET ATTRIBUTS DE BRAINENTROPY115
BIBLIOGRAPHIE.....120

LISTE DES FIGURES

	Page
Figure 1.1	Le regroupement de million de neurones perpendiculaires à la surface de la matière blanche est modélisé par un dipôle de courant dont sa direction est perpendiculaire à la surface corticale.....12
Figure 1.2	A) La disposition topographique des électrodes en vue de dessus; B) Électroencéphalogramme d’une activité épileptique. Chaque ligne représente la différence de potentiel entre deux électrodes en fonction du temps.....14
Figure 1.3	La tête est modélisée par plusieurs régions $\Omega_1, \dots, \Omega_{N+1}$ avec des conductivités constantes $\sigma_1, \dots, \sigma_{N+1}$ et séparées par les interfaces S_1, \dots, S_N . Les flèches indiquent la direction normale vers l’extérieur.....18
Figure 1.4	Correspondance entre les éléments d’une colonne de la matrice V et les valeurs aux nœuds du maillage des surfaces qui constituent le modèle de la tête.....20
Figure 1.5	Visualisation du fichier sujet.cond qui définit, dans OpenMEEG, la conductivité normalisée de l’air, de la peau, du crane et de la matière grise.....24
Figure 1.6	Position des sources en fonction du rapport de conductivité os/tissus-mous. La source dipolaire est montrée en violet. On observe que plus la conductivité du crâne est élevée plus la position d’une source obtenue est profonde. Tiré de Sylvain Vallaghé (2006).25
Figure 1.7	Édition d’un fichier d’extension .geom définissant les chemins des surfaces et leur ordre dans le modèle BEM.26
Figure 1.8	Schéma du modèle de la tête en plusieurs couches imbriquées.....27
Figure 1.9	De gauche à droite : vues axiale, sagittale et coronale de la surface corticale (rouge) imbriquée dans le cortex (jaune), la surface externe du crâne (vert), et la peau (bleu). Visualisation de la position des électrodes recalées selon la méthode expliquée dans le chapitre 4.28
Figure 1.10	Organigramme des fonctions d’OpenMEEG et de leurs entrées (sources, surfaces et position des capteurs).....30
Figure 2.1	Hiérarchie du package « création d’un modèle EEG ».....34

Figure 2.2	Résultat de la segmentation de l'hémisphère gauche pour obtenir le support anatomique des sources gauches. La segmentation de la peau à partir d'une IRM peut servir au recalage des électrodes ou au modèle BEM.	37
Figure 2.3	Interface de commande du recalage des électrodes permettant de placer la position des électrodes d'un repère relatif vers le repère volumique de l'IRM anatomique.	38
Figure 2.4	Étape 2 : Électrodes recalées sur la surface de la peau. La méthode utilisée minimise l'écart entre trois électrodes de référence (boules jaunes) appartenant au système d'électrodes et trois points de référence (boules roses) positionnées sur le nasion et aux points péri auriculaires.	39
Figure 2.5	Étape 3 : Module de BrainVISA permettant d'obtenir le maillage des sources, les informations des capteurs et la matrice de Gain EEG.	40
Figure 2.6	Fenêtre de suppression des électrodes non désirables du modèle. Leurs signaux ne seront pas pris en compte lors de la localisation.	41
Figure 2.7	Structures des fichiers de sortie de l'outil qui construit un modèle EEG: le maillage de surface corticale, les capteurs, les mesures, la matrice de gain EEG.	44
Figure 2.8	Organigramme des différents modules de l'outil « création d'un modèle EEG ». À partir d'une IRM anatomique le processus consiste à segmenter et décimer les maillages du cortex et de la peau. À partir des électrodes qui mesurent l'activité cérébrale, l'outil recale ces électrodes vers le repère de l'IRM permettant ainsi le calcul d'une matrice de gain.	45
Figure 2.9	Illustration du « skull stripping » qui est la première étape du logiciel pour obtenir les surfaces de tissus anatomiques.	48
Figure 2.10	« Skull stripping » effectué : le crane et la peau ont été enlevés.	49
Figure 2.11	Fenêtre «Skull and Scalp».	50
Figure 2.12	« Skull stripping » accompli.	51
Figure 2.13	La fenêtre « Surface View Properties » permet de contrôler la visualisation des différentes surfaces et de les enregistrer avec l'extension .dfs pour s'ouvrir avec BrainSuite2.	52
Figure 2.14	Affichage des quatre surfaces générées : Peau, crâne extérieur, crâne intérieur, cortex.	53

Figure 3.1	Équation matricielle du problème direct. Les mesures (M) en EEG et en MEG sont expliquées, via la matrice de gain (G), par l'activité des dipôles représentée par la matrice des sources (S) et par un bruit physiologique (N).56
Figure 3.2	Courbe en L de Pareto représentant les erreurs des sources/modèle et les erreurs des sources/mesures pour différent λ compris entre 0 et 2. La valeur de λ optimale est celle qui est la plus proche de l'origine.....60
Figure 4.1	Illustration du triangle du maillage de la peau (T) traversé par une droite (R). P_0 est au centre de la tête. Le point d'intersection (P_I) permet d'éloigner une électrode de la peau vers l'extérieur le long de cette droite.....70
Figure 4.2	Préparation des données pour créer une matrice de gain BEM avec BrainEntropy. Ici on visualise toutes les électrodes distantes de 2mm de la surface de la tête. Le calcul de la matrice de gain s'exécute en cliquant sur le bouton "calculer la matrice de gain".73
Figure 4.3	Interface IHM pour la création d'un modèle BrainEntropy. Le modèle pourra ensuite être chargé directement et il contient au moins les informations nécessaires à une simulation.75
Figure 4.4	Organigramme de la structure des classes de BrainEntropy.....77
Figure 4.5	Illustration des potentiels électriques et des champs magnétiques gravitant autour d'un dipôle de courant. Tiré de: http://lsiit-miv.u-strasbg.fr/contenu/fichiers_page/MEG-EEG-online.pdf79
Figure 4.6	Simulations du problème direct. Activation des sources représentées en rouge dans les régions frontale (A), pariétale (B), temporale (C), occipitale (D) et mésiale interhémisphérique (E). Les mesures sur les capteurs en EEG et en MEG sont dans la deuxième et dans la troisième colonne respectivement. On montre les mesures projetées sur le scalp ainsi que leurs interpolations sur les électrodes.81
Figure 4.7	Projection des champs magnétique et électrique sur scalp afin de vérifier la validité des matrices de gain générées par OpenMEEG et BrainVISA dans le cas de sources radiales et tangentielles actives.82
Figure 4.8	Tableau définissant les paramètres d'une analyse ROC qui permet d'évaluer la performance d'un algorithme de détection.85
Figure 4.9	Contrôle des modalités, de la fenêtre de temps, et de la méthode de résolution du problème inverse.88
Figure 4.10	Simulateur d'activité bioélectrique dans le logiciel BrainEntropy.....89

- Figure 4.11 Comparaison de méthodes de localisation de sources générées par les mêmes mesures. De gauche à droite, les méthodes sont : Norme L_p , Entropie de Shannon, Total Variation, Heat et « Minimum Norm ». L'aire sous la courbe ROC permet de quantifier la performance d'une méthode à détecter un objet « actif ». Ainsi, la méthode L_p en EEG (première ligne) se démarque des autres méthodes.....90
- Figure 4.12 Cartographie (en haut) en fonction du temps d'une localisation L_p obtenue à partir d'un signal EEG avec des pointes épileptique moyennées (en bas).91
- Figure 4.13 Identification d'une réaction bilatérale en EEG mais non présente en MEG lors de l'activation du foyer épileptique situé dans le lobe temporal gauche (LTG). La combinaison EEG et MEG permet d'atténuer le signal de la région non épileptique située dans le lobe temporal droit (LTD).93
- Figure 4.14 Cette figure montre l'organisation des principales contributions du travail effectué pendant cette maîtrise. Les contributions personnelles sont représentées par des boîtes jaunes et l'utilisation d'outils déjà existants par des boîtes bleues. On note un premier travail (à gauche) qui consiste à résoudre le problème direct le plus réaliste possible, et un second travail (à droite) qui contient des outils pour résoudre et vérifier le problème inverse dans l'environnement du logiciel BrainEntropy.97

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ACV	Accidents Vasculaires Cérébraux
BEM	Boundary Element Methods
BFC	Bias Field Corrector
CBLAS	C Basic Linear Algebra Subprograms
CNRS	Centre National de la Recherche Scientifique
CERNEC	Centre de Recherche en Neuropsychologie et Cognition
DICOM	Digital Imaging and Communications in Medecine
EEG	ElectroEncephaloGraphie
ENPC	École Nationale des Ponts et Chaussées
ÉTS	École de Technologies Supérieures
FDG	FluoroDeoxyGlucose
INRIA	Institut de Recherche en Informatique et Systèmes Aléatoires
IRF49	Institut Fédératif de Recherche n°49
IRM	Imagerie par Résonance Magnétique
IRMf	Imagerie par Résonance Magnétique fonctionnelle
ITK	Insight Toolkit
LAPACK	Linear Algebra PACKage
LATIS	LABoratoire de Traitement de l'Information et du Signal
MAP	Maximum A Posteriori
MB	Matière Blanche
MBA	Master

MEG	MagnétoEncéphaloGraphie
MNI	Montreal Neurological Institute
MEEG	Magneto-ElectroEncephaloGraphy
MEM	Méthode par Maximum d'Entropie
MG	Matière Grise
OS	Operating System
PA	Potentiel d'Action
RPM	Red Hat Package Manager
RSB	Rapport Signal sur Bruit
SPM	Statistical Parametric Mapping
SVN	SubVersioN
UdM	Université de Montreal
URL	Uniform Resource Locator

INTRODUCTION

Grâce aux plus récentes technologies en imagerie cérébrale, telle que l'imagerie par résonance magnétique (IRM), les cliniciens ont accès à l'anatomie du cerveau de façon non invasive sans ouvrir le crâne. Que ce soit en recherche médicale ou dans le cadre clinique des neurosciences, nous avons de nos jours accès à des instruments pour mesurer certains phénomènes cérébraux. En utilisant les mesures issues des technologies qui permettent de connaître l'anatomie de la tête ou de mesurer l'activité électromagnétique du cerveau, ce mémoire apporte une formule logicielle pour modéliser le milieu électrique de la tête et localiser l'activité cérébrale. Cette formule regroupe, d'une part, l'utilisation de logiciels déjà existants pour extraire des informations issues des acquisitions, et d'autre part, un outil développé dans le cadre de ce mémoire qui manipule les informations afin de localiser le lieu de l'activité cérébrale mesurée. La réalisation d'un tel outil informatique est importante pour la communauté des neurosciences car il permet de répondre aux questions où et quand une activité cérébrale d'intérêt se manifeste. Par exemple, avec cet outil les psychologues peuvent mieux étudier la cognition en connaissant les régions du cerveau qui s'activent face à des stimulations. De même, cet outil peut aider les neurologues et les neurochirurgiens à mieux appréhender certaines maladies neurodégénératives ou mieux préparer un processus préopératoire.

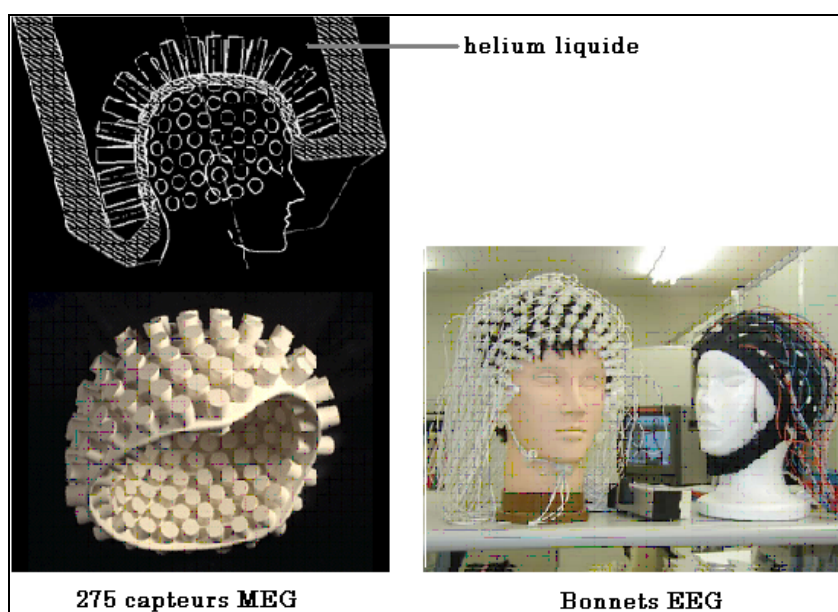
Les techniques d'imagerie utilisées dans ce mémoire sont l'électroencéphalographie (EEG) et la magnétoencéphalographie (MEG) qui sont deux modalités mesurant l'activité bioélectrique du cerveau. L'EEG mesure des potentiels électriques à la surface du scalp à l'aide d'électrodes placées sur la peau. Conçue en 1929, essentiellement pour l'étude des rythmes cérébraux, l'EEG détecte par exemple le rythme alpha¹ présent dans la région occipitale² du cerveau lorsque l'on ferme les yeux. Pour avoir un ordre de grandeur, le nombre de capteurs EEG peut varier d'une cinquantaine à une centaine. La MEG, quant à elle, vu la difficulté de mesurer des champs magnétiques très faibles de l'ordre de 10^{-15}

¹ Rythme cérébral dont la fréquence oscille entre 8 et 12Hz

² La région occipitale se situe à l'arrière de la tête et est associée au traitement de l'information visuelle.

Tesla³, ne fit son apparition que dans les années 1970. On compte environ 300 capteurs pour les équipements MEG les plus récents.

La localisation des sources d'activité cérébrale, dans le cadre de cette maîtrise, consiste à estimer le lieu et l'intensité de l'activité bioélectrique dans le cerveau des sujets ou des patients qui ont participé à des acquisitions EEG et/ou MEG.



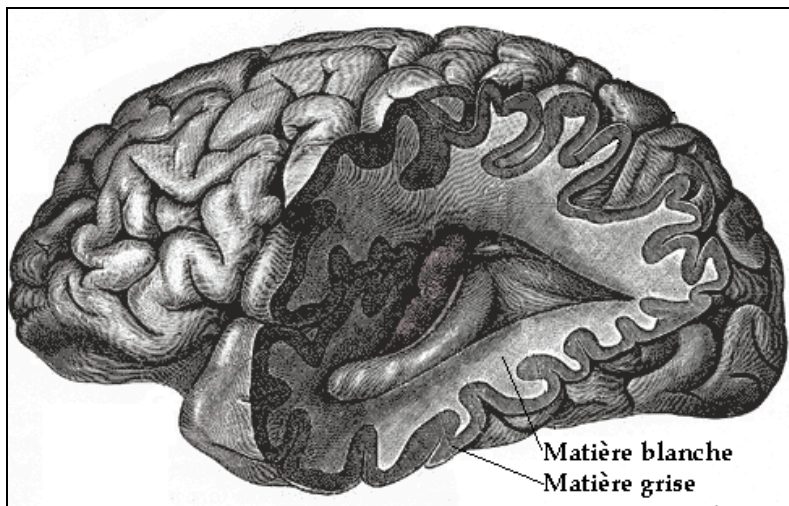
Photographies d'un casque MEG (à gauche) et de bonnets EEG (à droite) contenant respectivement les bobines et les électrodes. Tiré de : <http://pages-perso.esil.univmed.fr/~coulon.o/cours/MSISPro-2004-2005-Modalites.pdf>

Dans ce mémoire nous regroupons les mesures des acquisitions MEG et EEG en même temps de façon à réunir le plus d'informations possibles et de les utiliser conjointement pour mieux détecter l'endroit de l'activité cérébral. La combinaison des deux modalités dans la localisation d'une zone cérébrale active reste encore du domaine de la recherche, nous verrons de quelle manière ce mémoire traite ce sujet.

Quelle est l'origine physiologique des mesures EEG et MEG?

³ Le champ magnétique terrestre est de l'ordre de 10^{-5} Tesla

Le cerveau est principalement constitué de deux composantes : la matière grise (cortex) et la matière blanche. La matière grise, essentiellement constituée de cellules nerveuses pyramidales, forme un liseret d'environ 4-5mm d'épaisseur enveloppant la matière blanche.



Représentation d'un cerveau coupé pour illustrer la matière blanche et la matière grise.

La physiologie, telle que détaillée plus précisément dans [5], nous montre que les cellules sont toutes perpendiculaires à la surface corticale, que ce soit sur un gyrus (surface la plus visible du cerveau), ou dans un sillon (repli de la surface du cerveau). Les cellules pyramidales comportent un noyau avec, d'un côté, un long filament appelé dendrite apicale, se terminant par une multitude de ramifications de dendrites proches de la surface externe du cortex. Ces dernières permettent les communications inter-neurales. De l'autre côté du noyau cellulaire, on observe l'axone du neurone qui se prolonge dans la matière blanche en se ramifiant. Grâce à ce réseau de connexion complexe et très ramifié, l'activité cérébrale se manifeste par une activation bioélectrique synchrone de plusieurs milliers de cellules pyramidales et génère, localement, un grand nombre de petits courants. Le regroupement de cellules pyramidales dans une région du cortex forme ainsi à une échelle macroscopique un dipôle de courant comme l'illustre la figure de la prochaine page.

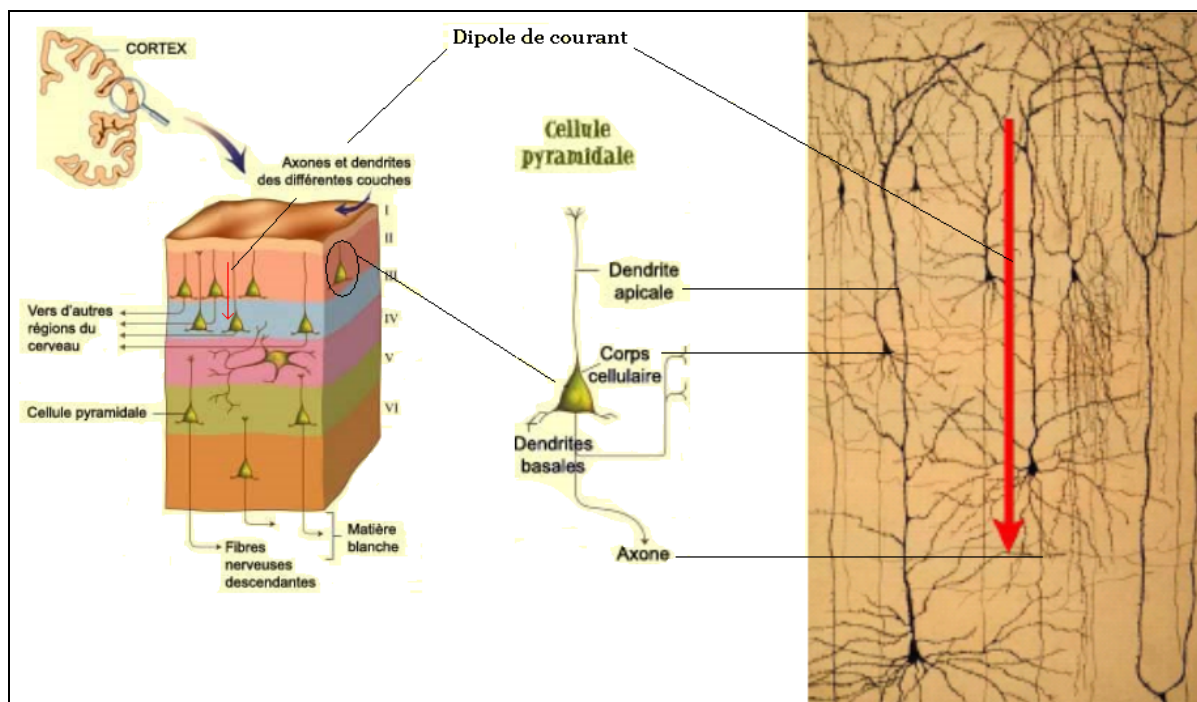


Illustration d'un dipôle de courant modélisant un grand ensemble de cellules pyramidales. La concentration de millions de neurones forme un dipôle de courant (flèche rouge). Tiré et modifié de : Neurosciences, À la découverte du cerveau 2^{ème} édition Mark F. Bear, Barry W. Connors, Michael A. Paradise; Édition Pradel, 2002.

Appelés courants primaires, ces courants circulent dans les dendrites et les axones et génèrent un champ magnétique mesuré par des bobines supraconductrices en MEG. De plus, les courants primaires chargent négativement le milieu extracellulaire avoisinant. Il apparaît alors de faibles courants secondaires qui se propagent dans les tissus biologiques de la tête jusqu'à la peau où ils sont mesurables par la différence de potentiels qu'ils engendrent entre les électrodes de l'EEG. Ces phénomènes bioélectromagnétiques sont décrits avec beaucoup plus d'informations dans [4].

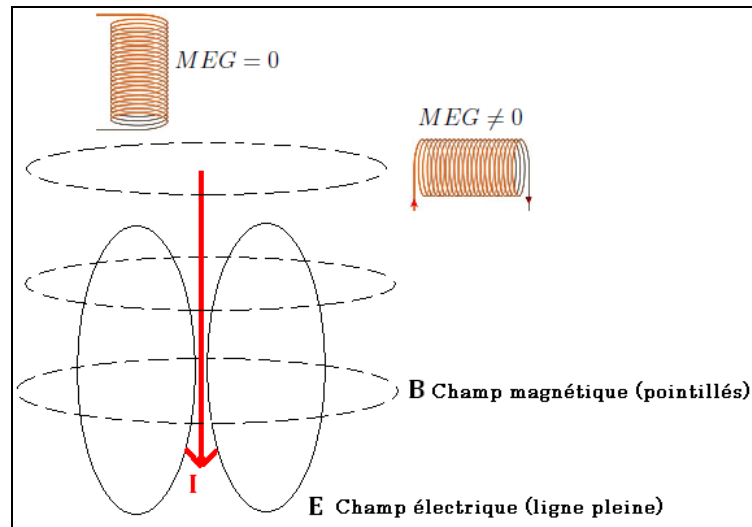
Dans ce mémoire nous allons utiliser une technique qui consiste à modéliser l'activité cérébrale comme l'amplitude de milliers de dipôles de courant situés le long de l'interface matière blanche / matière grise. Cette technique, appelée modèle distribué, à le bénéfice d'observer l'activité du cerveau avec une grande précision spatiale (mm^3). L'utilisation de cette méthode requiert d'avoir une IRM anatomique pour en extraire la surface géométrique

du ruban corticale. Il existe une autre méthode en usage pour représenter les sources, appelée le modèle dipolaire. Elle repose sur l'estimation de trois paramètres libres de très peu de sources (4 au maximum) qui sont la position, l'orientation et l'amplitude. Cette méthode nécessite de connaître a priori le nombre de régions actives dans le cerveau et elle ne correspond pas à un usage général tel que le fait le modèle distribué.

L'EEG est la technique la plus courante pour mesurer l'ensemble des processus bioélectriques du cerveau. Elle n'utilise pas un équipement coûteux comme la MEG et elle s'adapte aux mouvements de tête du sujet. Toutefois, en EEG la modélisation des courants secondaires se propageant dans la tête reste complexe. Ceci est dû essentiellement à la difficulté d'obtenir un modèle proche de l'anatomie, aux valeurs des conductivités des tissus biologiques humain utilisées dans le modèle et à l'anisotropie⁴ des tissus biologiques. Au contraire, en MEG il est plus facile de modéliser la propagation magnétique car les tissus biologiques interfèrent de façon négligeable sur le champs magnétique produit par les cellules pyramidales. Cependant, la MEG est très sensible à l'orientation des sources.

En effet, les cellules pyramidales situées au fait d'un gyrus produisent un champs magnétique perpendiculaire à l'orientation des capteurs MEG. Ce champs magnétique ne traverse alors pas les bobines MEG et les sources situées au sommet d'un gyrus ne sont pas détectées.

⁴ Propagation des courants dans plusieurs directions différentes.



L'intensité du champs magnétique mesurée dépend de l'orientation de la source par rapport à celle des capteurs MEG. Le champs est mesurable si le dipôle de courant est parallèle à une bobine. Si l'orientation du dipôle est perpendiculaire à une bobine celle-ci ne le mesure pas.

Le choix de ces deux technologies pour "écouter" l'activité cérébrale n'est pas anodin. La MEG et l'EEG sont les deux modalités qui ont la meilleure résolution temporelle. De plus, leur utilisation est non-invasive. C'est à dire que ce sont des mesures passives sans interaction directe avec les organes internes (pas d'effraction de la peau, ou d'injection d'agent radioactif comme en TEP⁵).

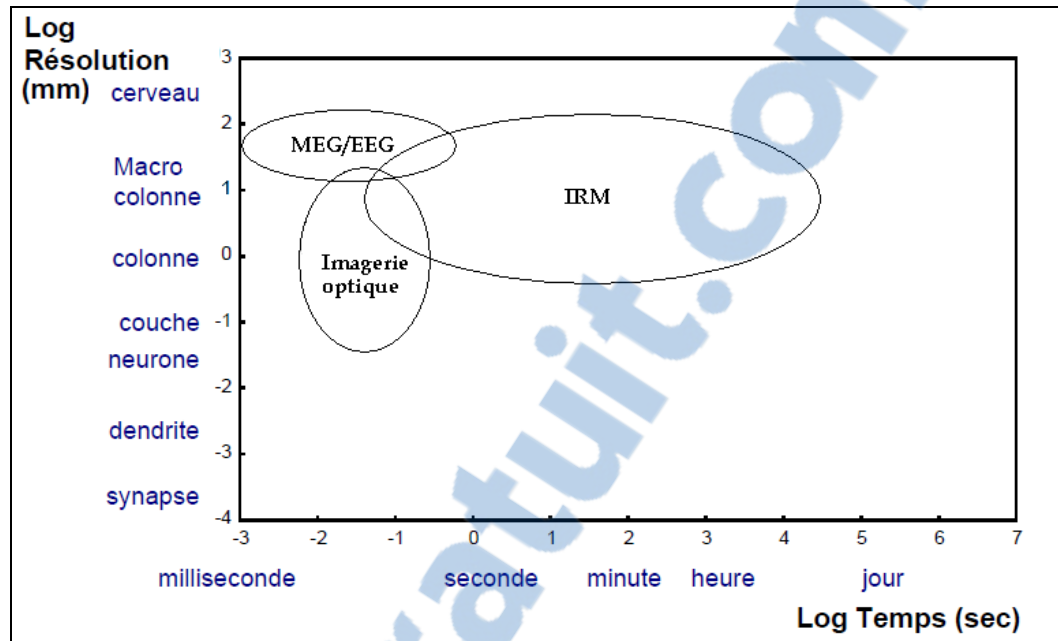
Le support anatomique de l'activité cérébrale est une information géométrique qui doit être disponible. On peut utiliser des gabarits de cortex qui existent déjà (pour les enfants par exemple), ou bien obtenir cette information à partir d'une acquisition IRM anatomique. Sommairement, l'IRM utilise un champs magnétique de forte intensité (1.5 à 3 Tesla) pour stimuler les propriétés paramagnétiques de la matière biologique. Plus précisément, cette technologie perçoit, tranche par tranche, la concentration locale de molécule d'eau, élément présent en abondance dans les tissus biologiques. L'image 3d, résultante d'une acquisition IRM, est constituée d'environ 255*255*180 voxels. Un voxel mesure en général 1 millimètre cube (résolution spatiale). Ils ont une intensité proportionnelle à la concentration locale

⁵ TEP : Tomographie par émission de positron. Technique d'imagerie nucléaire pour visualiser la diffusion d'un agent faiblement radioactif dans les organes.

d'hydrogène. Étant donné que les tissus biologiques n'ont pas tous la même concentration d'eau, il est possible de distinguer un contraste entre eux dans l'image IRM. Ce contraste est utilisé dans ce mémoire par le logiciel BrainSuite 2 pour obtenir la séparation entre les tissus biologiques. Il est notamment possible d'extraire l'interface matière blanche/matière grise pour obtenir le support géométrique de l'activité cérébrale.

En modifiant certains paramètres de son utilisation, l'IRM permet aussi de mesurer le fonctionnement hémodynamique du cerveau. On l'appelle alors l'imagerie par résonance magnétique fonctionnelle (IRMf). Le principe est le suivant. Lorsqu'une zone du cerveau s'active elle a besoin de sang oxygéné. Une petite augmentation de la consommation d'oxygène par les neurones est surcompensée par une large augmentation de flux sanguin. Il en résulte une diminution de la concentration de désoxyhémoglobine. Vu les propriétés paramagnétiques de cette molécule, le signal IRM augmente légèrement pendant les périodes d'activation dans le voxel où celle-ci a lieu. Toutefois nous n'utiliserons pas cette technique d'imagerie cérébrale dans ce travail pour deux raisons. Premièrement, l'IRMf permet de déduire indirectement l'activité cérébrale à travers l'activité hémodynamique, et deuxièmement, sa résolution temporelle de l'ordre de quelques secondes n'est pas assez fine.

Parmi les autres techniques d'investigation cérébrale que nous n'utiliserons pas dans ce travail, l'imagerie optique mérite d'être soulignée. Ses résolutions spatiale et temporelle sont compétitives face aux autres modalités, comme le montre la figure suivante.



Résolution spatio-temporelle et comparaison de différentes modalités d'imagerie cérébrale. La MEG et l'EEG ont de très bonnes résolutions spatiale et temporelle comparativement à l'IRM.

L'imagerie optique mesure aussi de l'activité hémodynamique comme l'IRMf. On fixe sur la tête des fibres optiques. Certaines fibres injectent de la lumière dans la tête (à travers la peau et l'os du crâne) et d'autres la récupèrent à la surface de la peau. En utilisant les propriétés d'absorption distinctes de l'oxy-hémoglobine et de la désoxyhémoglobine et en diffusant dans la tête deux longueurs d'ondes⁶ nous pouvons dissocier les deux types de molécules. C'est la variation des concentrations en hémoglobines qui est mesurée grâce à leur capacité d'absorption de la lumière. Ultimement, on sera intéressé à fusionner EEG, MEG et imagerie optique, mais à l'heure actuelle la modélisation de l'imagerie optique ne permet pas encore cette fusion avec autant de précision qu'en EEG et MEG.

La modélisation en EEG et en MEG du processus bioélectromagnétique de la tête fait la relation entre l'activité cérébrale et les signaux mesurés par les capteurs. Cette modélisation s'appelle le problème direct et il est représenté mathématiquement dans ce que l'on nomme une matrice de gain. L'utilisation du logiciel OpenMEEG (développé par le laboratoire

⁶ Une longueur d'onde dans le spectre visible et une autre proche de l'infrarouge.

Odysse de l'INRIA) permet d'obtenir des matrices de gain EEG et MEG de grande qualité. En effet nous verrons que ce logiciel prend en considération l'aspect anatomique et la conductivité électrique des tissus biologiques de la tête dans le calcul des matrices de gain EEG et MEG. Ceci représente un défi d'utiliser un modèle réaliste de la tête. En effet l'une des méthodes clinique d'usage consiste à modéliser la tête comme un milieu sphérique homogène. Dans ce type de modèle des courants secondaires ne produit aucun champ magnétique mesurable.

Avec un modèle réaliste de la tête et un modèle distribué de sources en main, on peut estimer des configurations spatiales de l'activité cérébrale produisant les mesures acquises. D'un point de vue mathématique, le fait de localiser des sources en fonction d'un signal reçu à distance s'appelle résoudre un problème inverse. Il existe énormément de solutions qui résolvent un problème inverse. Autrement dit, il existe plusieurs configurations des sources qui expliquent les mesures. Ceci vient du fait qu'il existe peu de capteurs pour beaucoup de sources. Le ratio du nombre de capteurs sur le nombre de neurones est très faible. On dit alors que le problème inverse est mal posé. En ajoutant de l'information a priori dans la méthode de résolution d'un problème inverse, il est tout de même possible de converger vers une estimation de l'intensité des sources réaliste.

Dans le but de modéliser le problème direct le plus réaliste possible, ce projet de maîtrise utilise la combinaison de logiciels déjà existants (BrainVISA, BrainSuite2 et OpenMEEG). Ainsi nous verrons comment il est possible de modéliser le processus bioélectromagnétique, mesuré en EEG et en MEG, à partir d'une image IRM et de la position des capteurs. L'association de ces logiciels nous permet d'obtenir des matrices de gain de grande qualité pour effectuer la résolution du problème inverse.

La principale contribution du projet de maîtrise présenté ici est le développement d'un environnement informatique (BrainEntropy), qui utilise en entrée les informations du problème direct, pour effectuer la validation de ce problème et la résolution du problème inverse (en utilisant deux approches).

Plus spécifiquement, les étapes et les contributions de ce mémoire sont :

a) La modélisation des processus bioélectromagnétiques (Chapitre 1) :

- description de la méthode mathématique utilisée dans le logiciel OpenMEEG.

b) Prétraitement des données EEG et MEG (Chapitre 2) :

- élaboration d'un module de prétraitement des données (BrainVISA)⁷;
- utilisation d'un outil de segmentation des interfaces des tissus biologiques de la tête (BrainSuite 2)⁸;

c) La localisation de l'activité cérébrale (Chapitre 3) :

- application de deux méthodes de résolution du problème inverse (minimisation de la norme L_p et minimisation de l'Entropie de Shannon);

d) Développement du logiciel BrainEntropy⁹ (Chapitre 4) :

- mise au point d'outils de recalage des électrodes EEG sur l'anatomie;
- intégration d'OpenMEEG pour le calcul des matrices de gain en EEG et en MEG;
- simulateur et validation des méthodes;
- localisation et visualisation de l'activité cérébrale en EEG, en MEG et en MEG/EEG combinées.

Ce mémoire dispose de trois annexes. Elles constituent un complément pour une compréhension complète et cohérente du contenu de ce mémoire. Elles présentent des détails techniques pour rendre utile ce document dans l'utilisation des outils mis au point dans ce travail. Enfin ces annexes permettent un développement futur des méthodologies et des outils élaborés dans le cadre de cette maîtrise.

⁷ BrainVISA est développé en grande partie par l'IFR49 : Imagerie Neurofonctionnelle, France

⁸ BrainSuite 2 est développé par David Shattuck, LA

⁹ Outil informatique qui fait le sujet de cette maîtrise.

CHAPITRE 1

LE PROBLÈME DIRECT ENTRE LES SOURCES ET LES CAPTEURS

1.1 Introduction

Le problème direct consiste à relier **les mesures** (signal reçu par les électrodes ou les capteurs) avec l'activation bioélectrique cérébrale. Il s'agit donc d'exprimer ici le modèle électromagnétique entre **les sources** et les mesures de potentiel ou magnétique. On peut alors dissocier deux espaces importants :

- L'espace des sources : Il représente l'activité cérébrale. Il existe deux principales modélisations de cet espace : une modélisation avec des sources distribuées sur un maillage spatial; ou une modélisation avec peu de sources "mobiles"¹⁰.
- L'espace de mesures : Les signaux et la position des capteurs EEG et MEG constituent l'espace des mesures.

Ce chapitre a pour but d'exprimer la relation entre ces deux espaces en modélisant le processus bioélectromagnétique du cerveau dans le problème direct en EEG et MEG.

1.2 Espace des sources

L'espace des sources est constitué de **dipôles** et peut se modéliser de deux façons : avec un modèle dipolaire ou un modèle distribué.

1.2.1 Définition d'un dipôle

Par abus de langage on mélange parfois sources et dipôles mais c'est l'ensemble des amplitudes des dipôles qui forme l'espace des sources. L'intensité d'un dipôle est celle d'un groupement de macro colonnes (cellules pyramidales).

¹⁰ Méthode dipolaire.

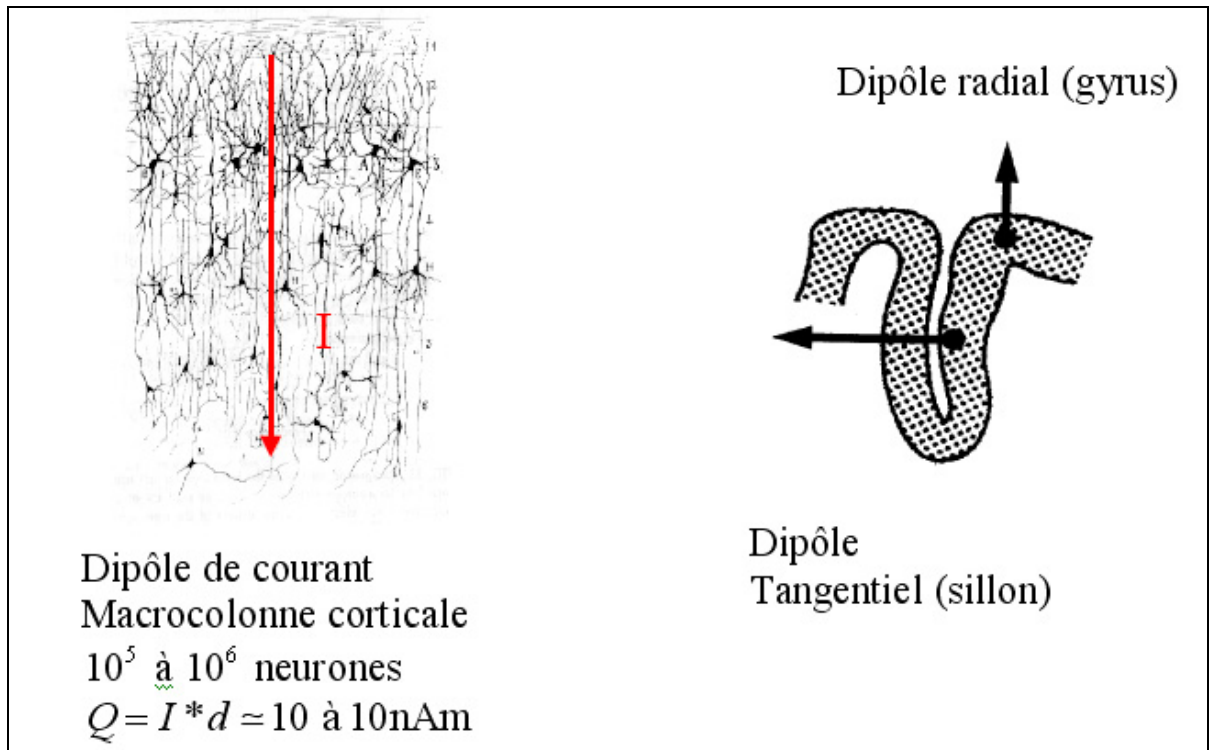


Figure 1.1 Le regroupement de million de neurones perpendiculaires à la surface de la matière blanche est modélisé par un dipôle de courant dont sa direction est perpendiculaire à la surface corticale.

Le moment dipolaire \vec{q} est une grandeur qui caractérise un dipôle électrostatique, il est donné par l'équation suivante :

$$\vec{q} = q \cdot \overline{NP} \quad (1.1)$$

avec le point P ayant une charge +q positive et le point N ayant une charge -q négative. Le moment dipolaire est exprimé en coulomb-mètre (Cm). Le moment du dipôle est souvent décomposé dans la base canonique de $R^3(e_x; e_y; e_z)$ de sorte qu'on a

$$\vec{q} = \overline{q_x} e_x + \overline{q_y} e_y + \overline{q_z} e_z \quad (1.2)$$

Lorsque les sources sont modélisées à l'aide de dipôles, le principe de superposition de l'électromagnétisme montre que le problème direct est linéaire par rapport au moment \vec{q} du dipôle.

1.2.2 Modèle dipolaire et Modèle distribué

Il existe deux types de représentation de l'espace des sources. Nous présentons ici leurs caractéristiques.

Le modèle dipolaire : Dans le cadre de la localisation de l'activité cérébrale, on utilise parfois les méthodes dipolaires pour optimiser la position et le moment (amplitude et orientation) d'un petit nombre de sources (3 ou 4) afin de minimiser l'erreur entre les mesures qu'elles produisent et les vraies mesures. Le nombre restreint de sources fournit des solutions très focales, ce qui est peu probable dans la majorité des cas. L'optimisation adéquate dépendra fortement des conditions initiales, autrement dit la solution risque de ne pas être unique.

Le modèle distribué : Les approches distribuées font l'hypothèse de plusieurs milliers de sources définies avec des positions et des orientations fixées sur l'interface matière blanche matière grise. Lors de la localisation de l'activité cérébrale, on estime les amplitudes de toutes les sources expliquant au mieux les mesures. Les approches distribuées ont le désavantage du très grand nombre de degrés de liberté à gérer. Par contre, la modélisation est plus réaliste et permet d'exprimer explicitement des contraintes (hypothèses), c'est pour cela qu'on choisit ici cette modélisation.

1.3 Mesures

Un neurone seul génère une activité électromagnétique indétectable. Pour qu'un signal soit mesuré, il est nécessaire que l'activité électrique provienne de plusieurs milliers de neurones locaux et synchrones. De plus, afin d'obtenir une réponse observable par un stimulus (réponse évoquée), il est souvent nécessaire de calculer la moyenne de plusieurs itérations de ce stimulus afin de réduire la contribution du bruit.

Nous distinguons les deux types de mesures électrophysiologiques, celle de l'EEG et celle de la MEG.

Les mesures EEG : Les électrodes en EEG sont appliquées à l'aide d'un gel conducteur assurant une bonne impédance entre la tête et les électrodes. Ce gel permet de mieux mesurer les potentiels à la surface de la tête. Dans la résolution du problème inverse, la position des électrodes sera cruciale. Ainsi avant de procéder à l'enregistrement des potentiels, il faut numériser les positions des électrodes relativement à la tête du patient¹¹. Les électrodes sont étiquetées en fonction de leur emplacement (souvent régit par le système 10-20) sur la tête du sujet (voir figure 1.2 A). Le nom qu'elles portent fait référence à la zone de mesure. Par exemple, les électrodes étiquetées de la lettre 'O' sont placées en arrière de la tête sur la région occipitale, celles avec la lettre 'F' sont placées en avant de la tête sur la région frontale, etc ... Dans la figure 1.2 B, on visualise des signaux provenant de la différence de potentiels mesurée par des paires d'électrodes. Les différences de potentiels mesurées sont de l'ordre du micro-volt.

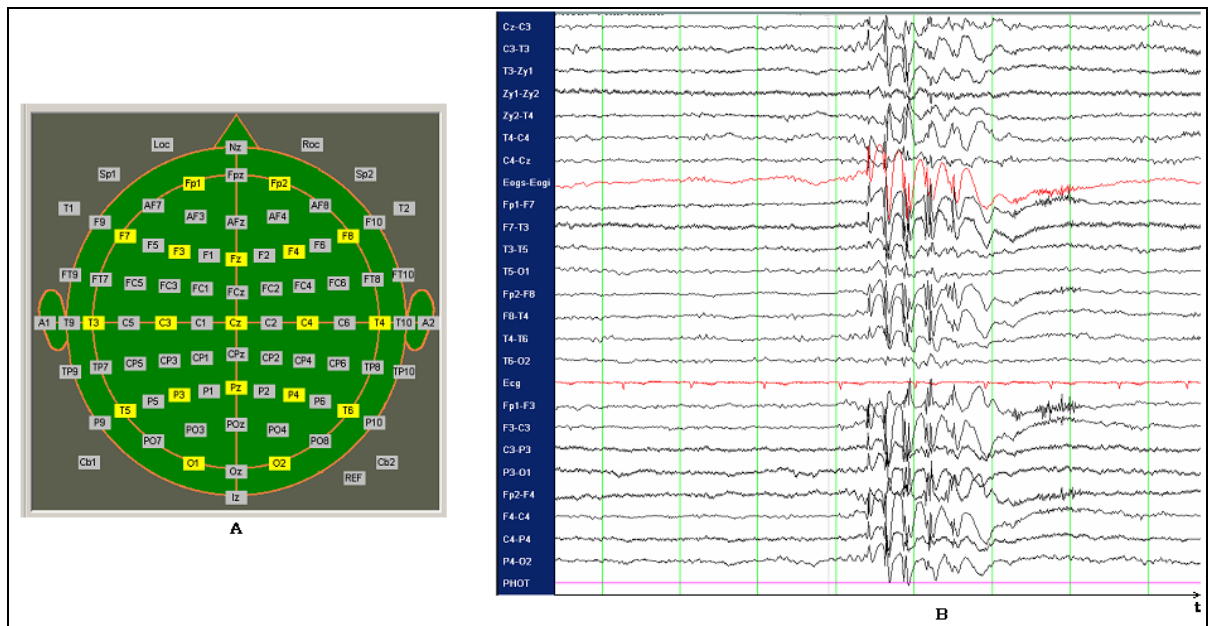


Figure 1.2 A) La disposition topographique des électrodes en vue de dessus; B) Electroencéphalogramme d'une activité épileptique. Chaque ligne représente la différence de potentiel entre deux électrodes en fonction du temps.

¹¹ Le système Polhemus permet de numériser des positions 3d de toutes les électrodes une fois posées.

Les mesures MEG : L'enregistrement de l'activité cérébrale en MEG consiste à placer le sujet dans un instrument qui mesure des champs magnétiques induits par l'activité électrique des neurones du cerveau. Les champs magnétiques mesurés étant extrêmement faibles (de l'ordre de quelques femto-Teslas), la MEG utilise une électronique basée sur des bobines supraconductrices (SQUIDS) refroidies à l'hélium liquide. Pour s'affranchir de toutes perturbations magnétiques, l'appareil est placé dans une pièce isolée magnétiquement par du mu-métal.

Dans le cadre d'une localisation de source (par exemple détection d'un foyer épileptogène), nous avons vu que le patient doit passer un examen IRM pour connaître l'anatomie de son cerveau. Or l'IRM magnétise fortement le patient et par effet d'hystérésis il reste magnétisé quelques jours. Dans cette perspective, les mesures en MEG doivent précéder l'acquisition en IRM.

1.4 Matrice de gain

La matrice de gain exprime la relation entre les sources et les mesures. Ici, on considère l'aspect mathématique du problème direct.

Pour un dipôle situé au point $p \in \mathbb{R}^3$ possédant un moment $\vec{q} \in \mathbb{R}^3$ et un point d'observation $r \in \mathbb{R}^3$, il existe un vecteur "ligne" $g(r, p)^t$ tel que la mesure en r s'exprime de la façon suivante

$$m(r) = g(r, p)^t \vec{q}$$

$g(r, p)^t$ est appelé vecteur de "Lead Field" associé à la source au point p et au capteur à la position r . Ce vecteur contient la modélisation électromagnétique de la propagation du potentiel et du champs magnétique, de la source vers les capteurs.

Les trois composantes du vecteur $g(r, p)$ sont calculées en solutionnant le problème direct (MEG ou EEG) sur les trois dipôles placés au point p et de moments unitaires respectifs e_x, e_y , et e_z .

Dans le cas des méthodes dipolaire avec plusieurs dipôles situés en $\overline{p_1}, \dots, \overline{p_p}$ et de moments respectifs $\overline{q_1}, \dots, \overline{q_p}$ il suffit de sommer les contributions individuelles de chaque dipôle. Dès lors, les mesures $m(r_k)$ pour $1 \leq k \leq m$ peuvent donc se mettre sous la forme

$$m(r_k) = \sum_{j=1}^p g(r_k, p_j)^t \overline{q_j}$$

On peut mettre cette écriture sous forme matricielle pour obtenir :

$$\underbrace{\begin{bmatrix} m(r_1) \\ \vdots \\ m(r_m) \end{bmatrix}}_m = \underbrace{\begin{bmatrix} g(r_1, p_1)^t & \cdots & g(r_1, p_p)^t \\ \vdots & \ddots & \vdots \\ g(r_m, p_1)^t & \cdots & g(r_m, p_p)^t \end{bmatrix}}_G \underbrace{\begin{bmatrix} \overline{q_1} \\ \vdots \\ \overline{q_p} \end{bmatrix}}_{\hat{q}} \quad (1.3)$$

Nous pouvons considérer le moment \vec{q} comme une intensité de source $J \in \mathbb{R}$ sur un axe orienté \hat{q} . Dans le contexte particulier de dipôles disposés sur la surface corticale, la normale au cortex fixe la direction des dipôles. Cette modélisation est justifiée par l'activité des cellules pyramidales dans les colonnes corticales qui sont anatomiquement perpendiculaire à la surface corticale. On contraint alors l'orientation des dipôles \hat{q} perpendiculairement à la surface du cortex pour une modélisation plus réaliste.

$$\vec{q} = \hat{q}J$$

Il est alors possible de réécrire la formulation de plusieurs dipôles situés en p_1, \dots, p_p avec cet espace de source particulier :

$$m(r) = \sum_{i=1}^p g(r, p_i)^t \hat{q}_i J_i$$

Ou encore sous forme matricielle

$$m = \begin{bmatrix} m(r_1) \\ \vdots \\ m(r_m) \end{bmatrix} = \begin{bmatrix} g(r_1, p_1)^t \hat{q}_1 & \cdots & g(r_1, p_p)^t \hat{q}_p \\ \vdots & \ddots & \vdots \\ g(r_m, p_1)^t \hat{q}_1 & \cdots & g(r_m, p_p)^t \hat{q}_p \end{bmatrix} \begin{bmatrix} J_1 \\ \vdots \\ J_p \end{bmatrix} \quad (1.4)$$

La matrice de gain G de dimension nombre de capteur par nombre de sources, est donc spécifique à la surface de source utilisée ici :

$$\begin{bmatrix} g(r_1, p_1)^t \hat{q}_1 & \cdots & g(r_1, p_p)^t \hat{q}_p \\ \vdots & \ddots & \vdots \\ g(r_m, p_1)^t \hat{q}_1 & \cdots & g(r_m, p_p)^t \hat{q}_p \end{bmatrix} \quad (1.5)$$

Cette matrice relie les intensités J des p dipôles du modèle aux m mesures (MEG ou EEG) dans une équation algébrique.

$$m = GJ \quad (1.6)$$

Notons que G est une matrice $m * p$ qui représente l'électromagnétisme dans le milieu biologique, entre les sources bioélectriques (positions p_i) et les capteurs (positions r_i). Dit autrement, une colonne de G (appelée aussi Forward Field) représente "l'effet" d'une source dipolaire sur l'ensemble des capteurs. De même, une ligne de la matrice G représente le facteur de gain de toutes les sources sur un capteur.

1.5 OpenMEEG pour la modélisation du problème direct

L'objectif de l'utilisation du logiciel OpenMEEG est le calcul des matrices de gain EEG et MEG. La principale caractéristique d'OpenMEEG est que ce logiciel prend en considération les interfaces de différents tissus biologiques de la tête. OpenMEEG permet aussi de prendre en compte les conductivités électriques des tissus situés entre les interfaces. Ainsi, ce logiciel modélise le problème direct en utilisant la méthode BEM¹² (Boundary Element Method). La fin de ce chapitre 1 est consacrée à l'explication et la mise en œuvre d'OpenMEEG afin d'obtenir des matrices de gain EEG et MEG modélisant au mieux les processus

¹² utilisation de surfaces délimitant les tissus biologiques.

bioélectromagnétiques de la tête. Ces matrices de gain nous permettrons plus tard d'utiliser des méthodes de localisation de sources.

1.6 Un peu de théorie sur l'approche de OpenMEEG dans la modélisation du problème direct.

La méthode BEM (Boundary Élément Method) modélise la tête comme un ensemble de surfaces emboîtées, délimitant des milieux (tissus biologiques) de conductivité différente. La figure 1.3 généralise cette méthode avec N surfaces et $N+1$ milieux. La surface la plus profonde, S_1 , est segmentée comme étant l'interface entre la matière blanche Ω_1 et la matière grise Ω_2 . Comme c'est dans cette région que se produisent les événements bioélectromagnétiques, elle sert de support géométrique aux dipôles de courants.

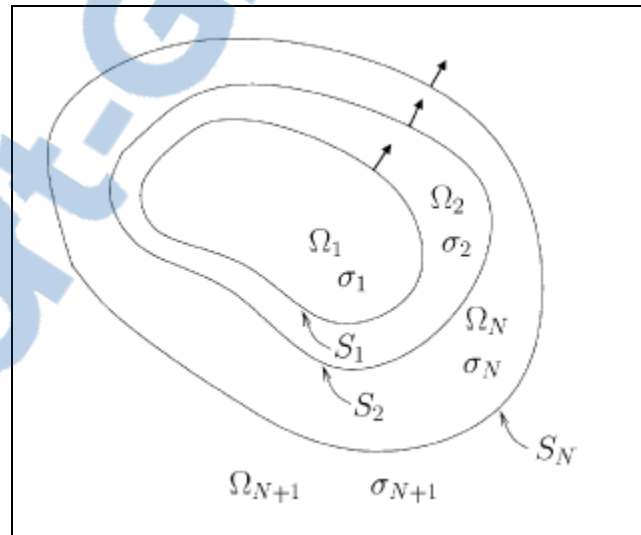


Figure 1.3 La tête est modélisée par plusieurs régions $\Omega_1, \dots, \Omega_{N+1}$ avec des conductivités constantes $\sigma_1, \dots, \sigma_{N+1}$ et séparées par les interfaces S_1, \dots, S_N . Les flèches indiquent la direction normale vers l'extérieur.

Un dipôle orienté perpendiculairement à la surface du cortex et positionné au point p_0 , avec son moment q , est défini par $J_{dip}(p) = q\delta_{p_0}(p)$.

Avec

$$\delta_{p_0}(p) = \begin{cases} 1 & \text{Si } p = p_0 \\ 0 & \text{Ailleurs} \end{cases}$$

L'équation suivante exprime le potentiel créé par le dipôle au point r :

$$U_{dip}(r) = \frac{1}{4\pi} \frac{q \cdot (r - p_0)}{\|r - p_0\|^3} \quad (1.7)$$

Pour une source située au point p , appartenant au cortex (maillage S_1), et perpendiculaire à celui-ci selon la norme $n_{S_1}(p)$, on exprime son intensité par $J_{surf} = j(p)n_{S_1}(p)\delta_{p_0}(p)$ avec sa densité de courant $j(p)$. Le potentiel électrique sur la surface du cortex $U_{surf}(r)$, produit par la source située au point p , est calculé en intégrant sur les points p_0 appartenant au maillage S_1 .

$$U_{surf}(r) = \frac{1}{4\pi} \int_{p \in S_1} \frac{n_{S_1}(p) \cdot (r - p)}{\|r - p\|^3} j(p) dp \quad (1.8)$$

On définit la densité de courant dans une surface en prenant en compte le champs électrique traversant un milieu ayant une certaine conductivité :

$$\sigma \frac{\partial U_{surf}}{\partial n}$$

Comme on l'explique en détail dans les références [2] et [3], le modèle anatomique de la tête comprend les interfaces, les conductivités et la distribution de sources peut se représenter par les équations suivantes :

$$LhsV = Rhs \text{ ou encore } V = Lhs^{-1}Rhs \quad (1.9)$$

Voici la composition de V pour N surfaces :

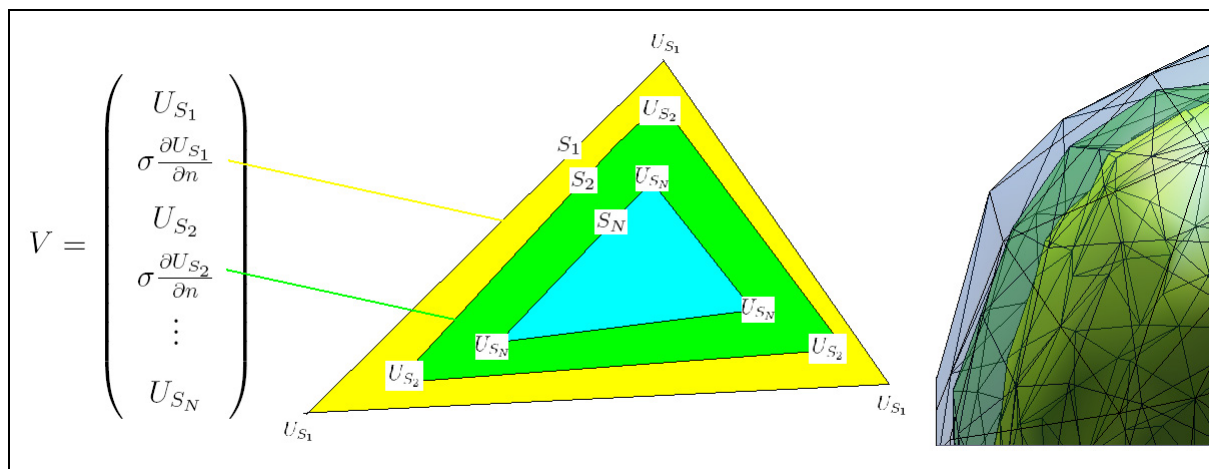


Figure 1.4 Correspondance entre les éléments d'une colonne de la matrice V et les valeurs aux nœuds du maillage des surfaces qui constituent le modèle de la tête.

La matrice Rhs définit la contribution de chaque dipôle vers les potentiels aux nœuds de toutes les interfaces qui créent le modèle de la tête. Rhs contient aussi les densités de courant relatives aux sources qui traversent toutes les faces des maillages de ces interfaces.

De la même manière, la matrice Lhs utilise la propagation électrique et les densités de courant mais qui, cette fois, sont en relation seulement entre les interfaces sans se soucier des sources.

Dans les deux cas, EEG et MEG, on a fait le choix d'avoir les mêmes interfaces et conductivités pour représenter la tête. Les relations électromagnétiques entre ces interfaces sont numérisées dans la matrice Lhs qui contient les propagateurs des potentiels entre les interfaces, et la capacité des courants secondaires à passer d'une région tissulaire à une autre. La matrice est symétrique car on peut voir les phénomènes de l'intérieur vers l'extérieur ou dans l'autre sens. La taille de cette matrice carrée est, la somme de tous les nœuds des interfaces, plus la somme des faces de toutes les interfaces exceptée la surface la plus à l'extérieure. En effet, on comprend bien qu'aucun courant ne sort de la tête pour se propager dans l'air ayant une conductivité nulle ! La matrice de gain EEG est calculée avec la convention que la somme des valeurs du potentiel sur les nœuds du scalp doit être nulle. La

matrice de gain EEG est donc référencée sur la moyenne des potentiels de toutes les électrodes posées sur la tête.

1.6.1 Matrice de Gain EEG

Par définition, la matrice de gain établit la relation entre les mesures sur les capteurs et l'activité des sources. Avec le modèle de la tête défini ci-dessus, la matrice U_{S_N} présente dans V représente les potentiels aux nœuds de la surface du scalp produit par les sources. C'est la matrice de gain, ou presque. Pour obtenir le potentiel mesuré sur les électrodes, on doit considérer une matrice P , appliquée à V , qui définit par des méthodes d'interpolation et de recherche des barycentres électriques, les potentiels aux nœuds de l'interface extérieure vers les capteurs EEG (voir [3]). Nous pouvons écrire l'expression de la matrice de gain EEG tel que :

$$G_{EEG} = PV = PLhs^{-1}Rhs \quad (1.10)$$

OpenMEEG permet de calculer numériquement toutes ces matrices.

1.6.2 Matrice de Gain MEG

Les prochaines informations au sujet du champs magnétique sont inspirées de [1]. En considérant que le champs magnétique varie à l'infinie, (donc sans harmonique), la loi de Biot et Savart nous donne :

$$B = \frac{\mu_0}{4\pi} \int J(p) \times \frac{r-p}{\|r-p\|^3} dp \quad (1.11)$$

Il est utile de décomposer la densité de courant J dans les deux composantes suivantes.

La première correspond au courant primaire J_p se propageant dans un milieu homogène et la seconde est la contribution du champs électrique E traversant un milieu de conductivité σ :

$$J = J_p - \sigma \nabla U = J_p + \sigma E \quad (1.12)$$

On peut alors décomposer l'équation du champs magnétique en deux :

$$B = \frac{\mu_0}{4\pi} \int J_p \times \frac{r-p}{\|r-p\|^3} dp - \frac{\mu_0}{4\pi} \int \sigma \nabla U \times \frac{r-p}{\|r-p\|^3} dp \quad (1.13)$$

OpenMEEG fait aussi la distinction des champs magnétiques issus des courants neuronaux et secondaires. C'est à dire qu'il existe deux options pour calculer les deux types de champs mesurés sur les capteurs : sToMEG et vToMEG. D'une part, les courants primaires viennent des sources, c'est pourquoi le modèle distribué avec la position des capteurs est pris en compte dans l'option sToMEG. D'autre part, on se sert des potentiels aux nœuds des interfaces avec leur conductivité dans l'option vToMEG. La matrice de Gain MEG est alors donnée par l'équation suivante.

$$G_{MEG} = \underbrace{sToMEG}_{\text{Courants primaires (dominants)}} + vToMEG * Lhs^{-1} * Rhs \quad (1.14)$$

1.7 Le logiciel OpenMEEG

OpenMEEG est une application développée par le groupe Odyssee membre de l'INRIA à Sophia-antipolis en France¹³. OpenMEEG est développé en C++, et fonctionne sous les environnements Linux et Windows. OpenMEEG utilise les bibliothèques mathématiques CBLAS et LAPACK écrites en Fortran (comme Matlab d'ailleurs). Il existe plusieurs modules contenant ces bibliothèques, notamment ATLAS et MKL que nous utiliserons respectivement pour Linux et Windows. On suggère l'utilisation d'OpenMEEG sous Linux. Pour procéder à

¹³ On peut citer comme acteurs et développeurs principaux du projet : Maureen Clerc, Théodore Papadopoulo, Alexandre Gramfort et Perine Landreau.

l'installation sous ces deux OS et obtenir des fichiers exécutables, on se référera à l'installation d'OpenMEEG décrite en Annexe A.

1.8 Description des répertoires de données et des types de fichiers de données

Dans cette section nous présenterons l'initialisation d'OpenMEEG en détaillant les fichiers configurables avant de démarrer le processus OpenMEEG. Cette initialisation repose sur les informations qui proviennent de l'IRM anatomique du sujet.

1.8.1 Contenu du répertoire HeadModels

L'intérieur de chaque répertoire patient de 'HeadModels' est structuré de la même façon. Il existe trois types de fichiers différents :

– Fichiers d'extension .tri

OpenMEEG utilise plusieurs surfaces dans l'application de la méthode BEM. Chaque fichier .tri est un maillage d'une surface écrit en format texte. La surface la plus profonde est le maillage des sources qui est souvent segmenté comme l'interface MB/MG. Les autres surfaces délimitent les régions homogènes et peuvent être en théorie aussi nombreuses qu'on le souhaite. Dans notre approche, on utilise 3 maillages de tissus : la peau, la surface du crâne externe, et le cortex entier. Notons que l'on pourrait considérer la surface du crâne interne produite par BrainSuite2.

Un fichier .tri est structuré comme tel :

Première ligne : - nombre de vertices % - NB_VERTICES

Ligne 2 à ligne NB_VERTICES+1 : positions et orientations %X(i) Y(i) Z(i) Ox(i) Oy(i)
Oz(i)

Ligne NB_VERTICES+2 : - NB_FACES NB_FACES NB_FACES

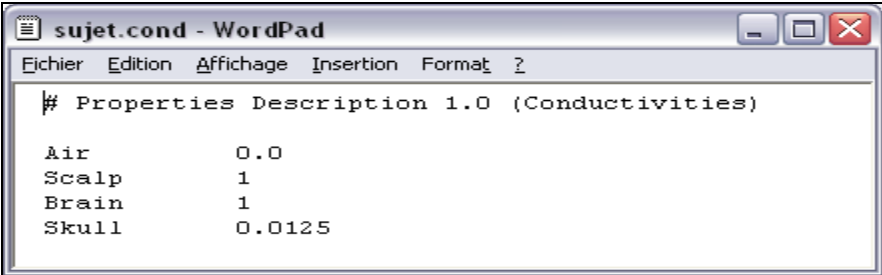
Les dernières lignes : faces % m.faces

Comme OpenMEEG n'accepte que les maillages dans le format .tri, il a fallu développer une fonction "m2tri.m" qui permet d'écrire un fichier texte tri à partir d'une structure de maillage

Matlab. Cette fonction calcule, entre autre, l'orientation de tous les nœuds en considérant la norme en chaque point du maillage. Un nœud permet de réunir et de former plusieurs faces ou triangles. Pour calculer l'orientation d'un nœud, on calcule simplement la moyenne des normales des triangles formés par ce nœud.

– Fichier d'extension .cond

Ce fichier définit les conductivités électriques des domaines situés entre les interfaces telle que le montre la figure 1.5. Leur valeur empirique proviennent de mesures in vitro sur des tissus humains morts. Cette modélisation ne reflète pas complètement le comportement électrique des tissus vivants, ou la variabilité inter-sujets (en particulier pour l'os du crâne), ou encore une éventuelle anisotropie. Cela peut produire des erreurs non négligeables dans l'application des matrices de gain dans le problème inverse. L'unité des conductivités du fichier .cond n'est pas en Siemens mais qui est normalisée par rapport à la conductivité des tissus mous (cerveau, peau).



```

# Properties Description 1.0 (Conductivities)

Air          0.0
Scalp       1
Brain       1
Skull       0.0125
  
```

Figure 1.5 Visualisation du fichier `sujet.cond` qui définit, dans OpenMEEG, la conductivité normalisée de l'air, de la peau, du crane et de la matière grise.

Le choix de la valeur de la conductivité du rapport os/tissus-mous est discutable. On utilise maintenant plutôt la valeur $1/20$, alors qu'avant les personnes faisant de la BEM utilisaient $1/80$. Si ce rapport est élevé, cela signifie que les courants secondaires traversent plus facilement le crâne qui a une forte conductivité. Autrement dit, pour deux rapports différents, et pour les mêmes mesures, l'emplacement des sources ne sera pas le même. Pour un rapport crâne/tissus-mous élevé (bonne conductivité du crâne) on localisera des sources plus profondes que celles qui seraient calculées avec un rapport plus faible. Pour une optimisation

des conductivités des tissus de la tête, je vous réfère aux travaux réalisés par Sylvain Vallaghé du centre Odysée dans [11].

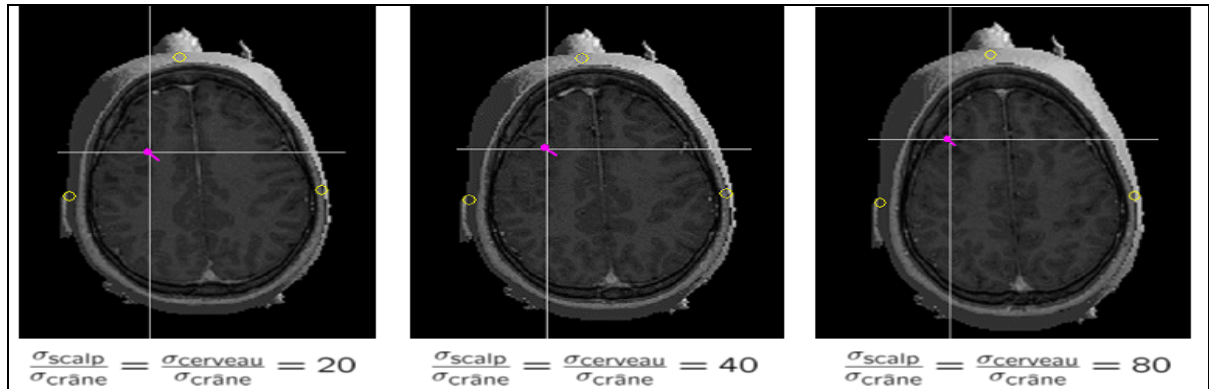
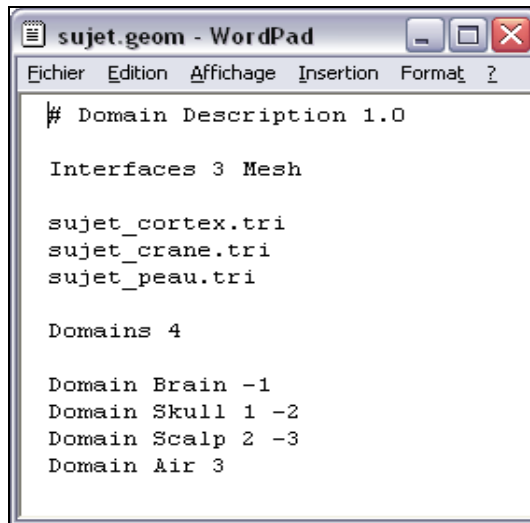


Figure 1.6 Position des sources en fonction du rapport de conductivité os/tissus-mous. La source dipolaire est montrée en violet. On observe que plus la conductivité du crâne est élevée plus la position d'une source obtenue est profonde. Tiré de Sylvain Vallaghé (2006).

– Fichier d'extension .geom

Ce fichier permet de définir comment les maillages sont organisés dans la tête. Ainsi, dans la première partie du fichier .geom, on trouve les chemins des interfaces (maillages) dans un ordre précis. La partie inférieure définit comment sont placées ces surfaces en définissant des domaines (aussi utilisés par le fichier .cond). Les domaines sont délimités par des numéros qui correspondent à la position du chemin d'une interface dans la première partie du fichier .geom. Le signe de ces numéros indique si le domaine est en dessous (négatif) d'une interface, ou au dessus (positif).



```
# Domain Description 1.0

Interfaces 3 Mesh

sujet_cortex.tri
sujet_crane.tri
sujet_peau.tri

Domains 4

Domain Brain -1
Domain Skull 1 -2
Domain Scalp 2 -3
Domain Air 3
```

Figure 1.7 Édition d'un fichier d'extension .geom définissant les chemins des surfaces et leur ordre dans le modèle BEM.

Il est alors possible de reconstituer le système comme l'illustre la figure 1.8.

– Fichier d'extension .posrths

C'est un fichier texte du maillage des sources qui compte autant de lignes que de sources et six colonnes dont 3 pour la position et 3 pour l'orientation des sources. Nous verrons plus tard dans quel cas ce fichier est utilisé. C'est une alternative au fichier *.tri du maillage des sources qui ne prend pas en compte les faces.

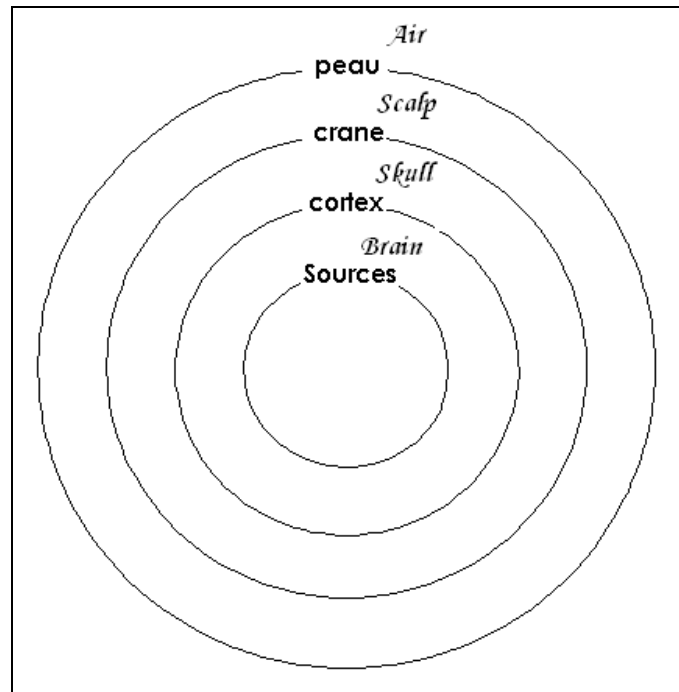


Figure 1.8 Schéma du modèle de la tête en plusieurs couches imbriquées.

1.8.2 Contenu du répertoire 'Computations'

Il existe deux types de fichiers qui sont dans les répertoires sujets de 'Computations'. Ils sont relatifs aux capteurs MEEG.

– Le fichier d'extension .patches

Celui-ci conserve les positions des électrodes EEG, sur \bar{x} \bar{y} \bar{z} , dans l'espace de volume IRM. C'est un fichier texte.

– Le fichier d'extension .squids

Celui-ci conserve les positions et les orientations des squids de la MEG dans l'espace de volume IRM. C'est un fichier texte.

1.8.3 Contenu du répertoire 'IntermediateFiles'

Les fichiers de sorties d'OpenMEEG sont écrits dans les répertoires sujet de 'IntermediateFiles'. À chaque étape, OpenMEEG écrit un fichier sortie dans ce répertoire avec un nom d'extension spécifique à l'étape effectuée.

1.9 Maillages et position des électrodes.

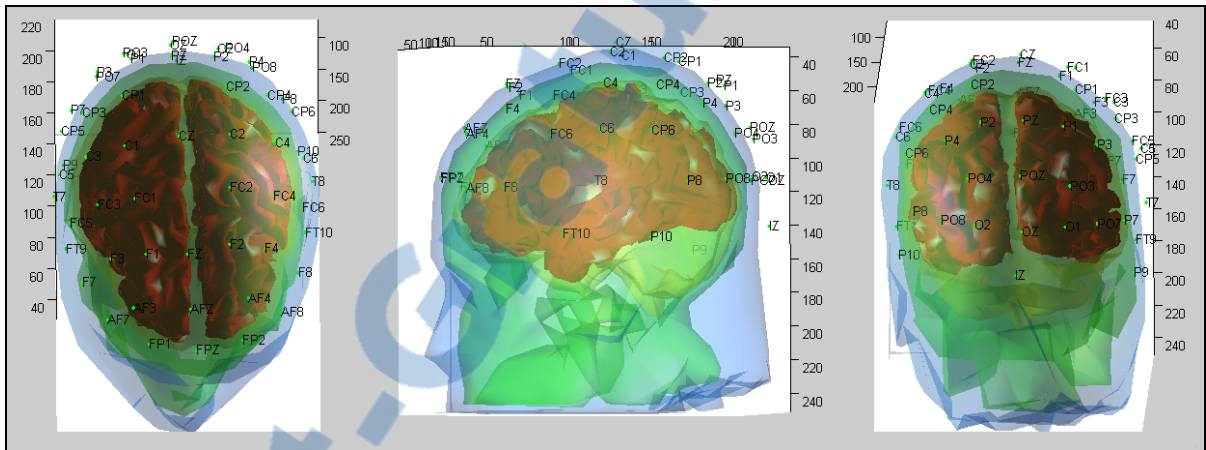


Figure 1.9 De gauche à droite : vues axiale, sagittale et coronale de la surface corticale (rouge) imbriquée dans le cortex (jaune), la surface externe du crâne (vert), et la peau (bleu). Visualisation de la position des électrodes recalées selon la méthode expliquée dans le chapitre 4.

Tel que nous le verrons dans le chapitre 2, BrainVISA et BrainSuite2 permettent d'obtenir des maillages de surfaces à partir d'IRM. Or, ces surfaces peuvent être dans des formats différents (.mesh pour BV, .dfs pour BS, .mat pour Matlab, et .tri pour OpenMEEG) et elles sont trop raffinées dans le cas du format Analyse. Au cours de notre travail, on a constaté que les maillages des interfaces OpenMEEG doivent être simples et ne pas dépasser 1000 noeuds.

Au-delà de ce nombre, la stabilité du logiciel n'est plus assurée. Cette condition respectée, il sera alors possible d'utiliser un maillage de sources très denses en vertices (et faces) comparativement à d'autres logiciels. Il a donc fallu créer un package 'préparation_des_données_OpenMEEG' qui permet notamment de :

- Réduire le nombre de noeuds des interfaces.

- Préparer et écrire les fichiers dans les répertoires patients de HeadModels et Computations.
- Visualiser si les surfaces, la surface corticale, et les électrodes sont toutes bien recalées et orientées.

En résumé le package 'préparation_des_données_OpenMEEG' écrira, dans les fichiers .tri, .patches ou .squids, les positions des vertices et des capteurs dans un repère millimétrique basé sur des images IRM de voxels homogènes.

1.10 Résumé sur OpenMEEG

La performance d'OpenMEEG est de fournir des matrices de gain calculées à partir de modèles multi-couches réalistes. Ce logiciel nous assure ainsi de modéliser le problème direct de façon fiable en étant à la pointe de ce type de modélisation. Ainsi avec OpenMEEG nous obtenons, d'une part, une matrice de gain EEG calculée en fonction des courants volumiques traversant les différents tissus de la tête et d'autre part, une matrice de gain MEG, utilisant la loi de Biot et Savart sur les courants primaires et secondaires. Ce logiciel est toujours en développement constant et au moment de la rédaction de ce texte, l'INRIA poursuit ses travaux quant à l'intégration d'OpenMEEG dans BrainVISA. Peu d'outils dans le domaine des logiciels en neuro ingénierie fournissent d'aussi bons résultats comme le montre les simulations de localisation de sources du chapitre 4. On note aussi que l'utilisation d'OpenMEEG dans l'environnement Windows est risquée. Les mauvais résultats obtenus sont probablement dus, à la librairie Intel MKL qui ne calcule pas et n'inverse pas correctement la matrice LHS. Nous avons néanmoins développé un module qui appelle toutes les fonctions d'OpenMEEG dans l'environnement Linux émulé par Cygwin¹⁴. Ce dernier module écrit automatiquement les commandes OpenMEEG dans un fichier nommé commande.sh. Ce fichier est situé dans le répertoire de Cygwin qui porte le nom de l'utilisateur qui a ouvert la session. Grâce à l'appel de commandes.sh dans .batchrc, Cygwin exécutera automatiquement les commandes OpenMEEG dès son ouverture dans le but d'obtenir des matrices de gain calculées dans l'environnement Windows.

¹⁴ Cygwin est un émulateur Linux fonctionnant avec Windows.

1.11 Conclusion sur le chapitre 1

Dans ce premier chapitre, nous avons expliqué que la résolution du problème direct consiste à évaluer les mesures (EEG ou MEG) à partir d'un générateur d'activité cérébrale. Nous avons d'abord défini l'espace des sources, composé de dipôles représentant l'activité électrique distribuée sur le cortex. Puis, nous avons introduit l'espace des mesures en expliquant le processus de mesure des technologies EEG en MEG. Par la suite, nous avons établi la relation entre ces deux espaces en définissant mathématiquement la matrice de gain. Enfin, nous avons décrit cette relation en prenant en compte l'anatomie et les conductivités électriques des tissus biologiques de la tête tel que le modélise le logiciel OpenMEEG.

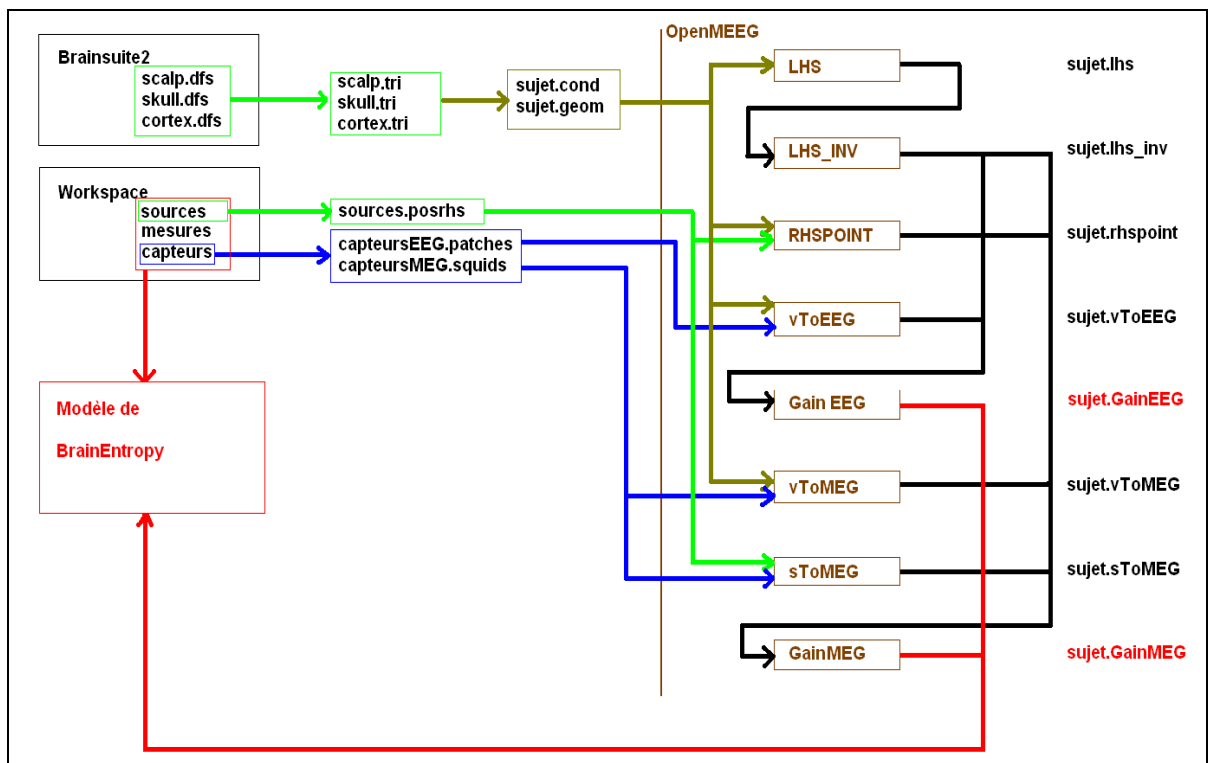


Figure 1.10 Organigramme des fonctions d'OpenMEEG et de leurs entrées (sources, surfaces et position des capteurs).



CHAPITRE 2

LES LOGICIELS BRAINVISA ET BRAINSUITE

2.1 Introduction

Dans le chapitre 1, nous avons défini l'espace de sources (dipôles électriques) et celui des mesures (signaux EEG et MEG). La relation entre sources et mesures est exprimée par une matrice de gain qu'on obtient, dans le cadre de ce mémoire, grâce au logiciel OpenMEEG. Cet outil numérique prend en entrée, un certain nombre d'informations sur l'anatomie cérébrale du sujet et la position des capteurs. Dans le chapitre 2, nous verrons comment, à partir des données d'acquisition, obtenir les informations nécessaires pour le fonctionnement d'OpenMEEG.

Avec le logiciel BrainVISA, nous extrairons la surface des sources à partir d'une image IRM. De plus, étant donné que les fichiers d'acquisition EEG ne contiennent pas seulement les électrodes mesurant l'activité cérébrale, il a fallu développer une interface, dans BrainVISA, permettant d'exclure les électrodes n'ayant pas cette fonction¹⁵. BrainVISA inclut un outil de calcul de matrice de gain basé sur un modèle sphérique de la tête assez simpliste. Nous l'utiliserons dans le but de comparer le comportement de ces matrices faces à celles produites par OpenMEEG. La collecte de ces informations, avec BrainVISA, nous permettra de faire des localisations de sources en utilisant le logiciel BrainEntropy, détaillé dans le chapitre 4 de ce mémoire.

Dans le but d'utiliser la méthode BEM, pour la modélisation de la tête du sujet, nous devons obtenir des surfaces 3D représentant les frontières entre les tissus biologiques. BrainVISA, ne permet pas d'obtenir par exemple un maillage du crâne adéquat, ou celui qui délimite la matière grise du liquide céphalo-rachidien. Ce manque d'informations nécessita d'effectuer une recherche d'un outil permettant d'obtenir des maillages précis des tissus biologiques de

¹⁵ électrodes mesurant par exemple le rythme cardiaque.

la tête. Cette recherche aboutit sur le logiciel BrainSuite 2. Nous verrons comment l'utiliser dans la suite du chapitre 2.

2.2 Présentation du logiciel BrainVISA

Nous présentons brièvement BrainVISA avec les fonctions de base fournies dans la version téléchargeable du logiciel¹⁶. On fait l'hypothèse qu'il y aura toujours une acquisition EEG (et IRM anatomique).

BrainVISA résulte de la collaboration de méthodologistes travaillant à l'Institut Fédératif de Recherche de neuro-imagerie (IFR 49) en France. Le but de cette collaboration était de fournir aux chercheurs cliniciens (ou non) des résultats méthodologiques en neuro-imagerie avec une interface facile à utiliser. Cette interface est également conçue pour simplifier l'échange de codes développés par différents laboratoires. Le premier prototype de BrainVISA a été développé par Y. Cointepas au début des années 2000. En 2003 la première version fut diffusée, et c'est en 2004 que fut mis à disposition un premier outil d'analyse MEG/EEG. Depuis, le noyau de BrainVISA est maintenu et développé par l'équipe de JF. Mangin au centre Neurospin et des ingénieurs de l'IFR 49. Certaines institutions Françaises comme l'INSERM, le CNRS, le CHU de la Pitié-Salpêtrière ou de Marseille et Lyon, participent aussi activement au développement et au maintien de BrainVISA. L'outil est également utilisé au niveau international, en particulier au centre MEG de l'UdM.

BrainVISA est tout d'abord une plateforme d'intégration multimodale qui reste aujourd'hui encore assez unique. Ce logiciel rassemble donc, d'un côté, les utilisateurs tels que les médecins, les chercheurs en neurosciences ou encore les étudiants en milieu hospitalier, et d'un autre côté les méthodologistes, les ingénieurs et les programmeurs. Ce logiciel doit être regardé comme un outil dédié au traitement des images cérébrales. Il utilise un langage simple (python) permettant aux méthodologistes de construire de longues chaînes de traitements élémentaires (des "pipelines" dans le jargon de la communauté, des "moulinettes"

¹⁶ Version de BrainVISA 3.02, voir brainvisa.info/index_f.html

dans celui de BrainVISA). BrainVISA utilise aussi un outil de visualisation 3D (Anatomist) parmi les plus performants pour visualiser, fusionner et analyser des images volumiques. Nous parlerons du fonctionnement d'Anatomist plus loin.

BrainVISA vise à être un logiciel ouvert, autrement dit,

- a) Les fichiers sources sont accessibles (en langage python).
- b) Aucune approche d'analyse d'image n'est imposée : n'importe quel laboratoire peut ajouter ses propres outils, et créer finalement ses propres "moulinettes".
- c) Aucun format d'image n'est imposé pour autant qu'on procure à BrainVISA les fonctions de conversion qu'il gère automatiquement.
- d) La plupart des outils fournis avec BrainVISA sont faciles à comprendre, même par des débutants. En outre, python étant un langage interprété, il n'y a pas besoin de compilateur. Par conséquent, la programmation de "moulinettes" dans BrainVISA est semblable à l'écriture d'un script, et les scripts existants peuvent servir d'exemple.

2.3 Mise en contexte

Le but de l'utilisation de BrainVISA dans la problématique étudiée ici est d'extraire et d'archiver (dans des fichiers) de l'information provenant des acquisitions IRM et EEG. Les images IRM fournies par les hôpitaux sont souvent archivées en format DICOM (.dcm). Les fichiers originaux sont par exemple :

- tranches IRM du sujet (.dcm);
- positions des électrodes et leurs noms (.spf);
- mesures (.avg).

Dans cette étude nous utilisons BrainVISA pour générer des fichiers Matlab contenant les informations nécessaires à la résolution du problème inverse :

- maillage du cortex (.mat);
- matrice de gain (.mat);
- structure contenant les positions des électrodes recalées dans le volume IRM et leur nom (.mat);

- mesures (.mat).

BrainVISA fait le lien entre les données mesurées en milieu hospitalier (EEG, MEG, IRM...) vers la plateforme d'analyse Matlab BrainEntropy (détaillée au chapitre 5). Pour réaliser ce lien il a fallu développer le package "Création d'un modèle EEG".

2.4 Utilisation de la librairie "Création d'un modèle EEG"

Pour procéder à l'installation de BrainVISA et de cette librairie en particulier, on se référera à l'Annexe B. La librairie "création d'un modèle EEG" est composée en plusieurs étapes tel que l'illustre la figure 2.1.

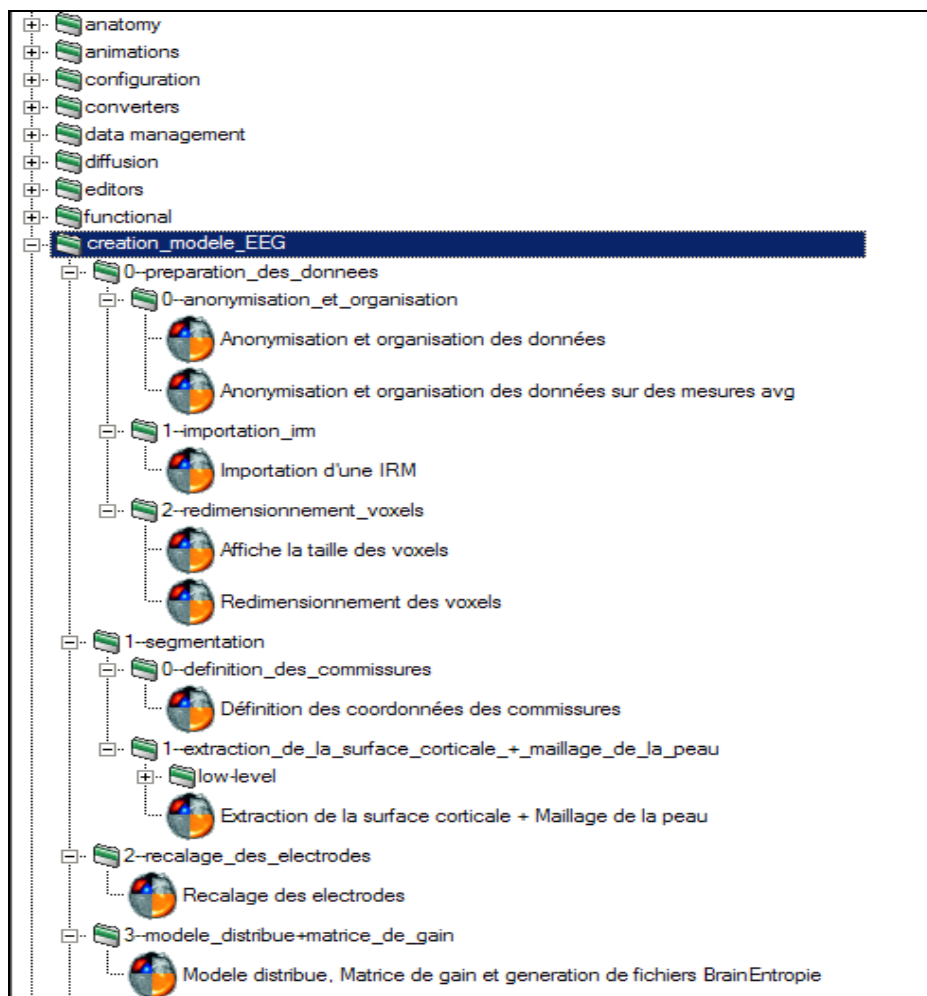


Figure 2.1 Hiérarchie du package « création d'un modèle EEG ».

2.4.1 Étape 0 : Préparation des données

La préparation des données consiste à fournir au logiciel BrainVISA l'image IRM, les informations des électrodes et le nom ou l'identifiant du sujet afin d'archiver ces fichiers dans la base de données du logiciel. La préparation des données s'effectue en trois sous-étapes.

Étape 0-0 : Anonymisation et organisation des données

Ce premier "pipeline" permet de créer le dossier sujet à partir de son IRM (IRMa_DICOM), des signaux acquis en EEG (EEG_signal) et de la position des électrodes (elec_digit).

Étape 0-1 : Importation de l'IRM

Conversion du format de l'image IRM vers le format .ima lisible par le visualisateur Anatomist.

Étape 0-2 : Redimensionnement des voxels

Étant donné que l'algorithme de segmentation travaille sur des voxels cubiques, cette sous-étape permet de vérifier que les voxels du volume IRM sont bien isotropes. C'est à dire que la largeur, la longueur et la profondeur d'un voxel sont bien égales. Si ce n'est pas le cas, on peut ici convertir les voxels pour que toutes les dimensions soient normalisées à la même taille.

2.4.2 Étape 1 : Segmentation de la surface corticale

Étape 1-0 : Définition des commissures

Pour segmenter correctement les deux hémisphères, le logiciel requiert des points de référence. Par convention, ces points correspondent aux deux commissures¹⁷ antérieure et postérieure, plus un point sur l'hémisphère gauche qui identifie la gauche réelle sur la

¹⁷ Les commissures sont les deux points de jonction inter-hémisphérique utilisés aussi pour définir le repère de Talairach.

visualisation d'Anatomist (convention neurologique)¹⁸. Dans la version actuelle de BrainVISA on suggère de laisser le champ "allow_flip_initial_MRI" à false. Les détails de cette opération sont décrits dans le rapport d'utilisation.

Étape 1-1 : Extraction de la surface corticale + Maillage de la peau

C'est dans cette étape que l'on bénéficie le plus de la performance de l'outil de segmentation dans BrainVisa. En effet, il est possible de segmenter l'interface matière blanche/matière grise des hémisphères à partir des tranches IRM. La surface issue de cette segmentation est représentée numériquement comme un maillage de nœuds constitué de faces triangulaires. Généralement, on code ce type de maillage selon la structure suivante :

- m.vertices : Tableau des positions (x,y,z) des nœuds du maillage dans le repère de l'IRM.
- m.faces : Tableau de triplets d'indices de "vertices" représentant les triangles qui composent la surface.

Le maillage de la surface corticale obtenu par BrainVISA contient un grand nombre de nœuds. Une décimation de ce maillage s'impose avant de le considérer comme le modèle distribué de sources afin de simplifier le calcul et la taille de la matrice de gain.

La surface de la peau est également segmentée dans le but de a) procéder au recalage des électrodes b) l'utiliser dans la méthode BEM pour le calcul de la matrice de gain c) visualiser la projection des potentiels ou des champs magnétiques à la surface de la tête (voir chapitre 4).

2.4.3 Étape 2 : Le recalage des électrodes

Le recalage consiste à modifier les positions des électrodes mesurées au moment de leur pose (enregistrées dans le fichier avec l'extension .spf avec l'instrument Polhemus) vers des positions recalées par rapport au maillage de la peau référencé dans l'espace IRM. Il s'agit de mettre en correspondance les positions des électrodes avec l'anatomie IRM du sujet. Une fois

¹⁸ Il existe en effet la convention radiologique qui inverse la gauche et la droite. Sans une identification précise du côté gauche (ou droit), on ne peut pas connaître l'orientation réelle du volume anatomique.

obtenu le maillage de la morphologie de la tête, nous devons placer sur ce maillage trois points de référence. Ces points correspondent à l'emplacement des trois électrodes de référence : une située au nasion et les deux autres aux points periauriculaires (à côté des oreilles). Pour effectuer cette opération dans Anatomist, on doit indiquer l'emplacement des points de référence sur la tête numérisée du sujet (représentée figure 2.4) à l'aide du curseur. Les coordonnées du curseur s'enregistreront automatiquement dans l'un des trois champs Nz, LPA, RPA en utilisant le bouton approprié sur BrainVISA.

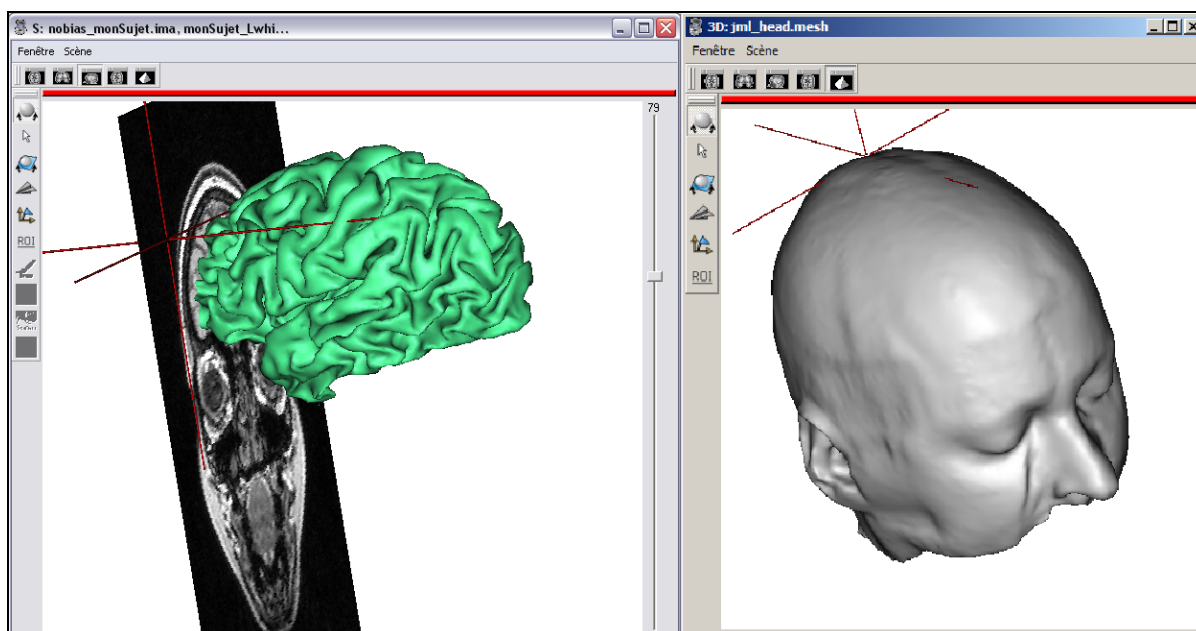


Figure 2.2 Résultat de la segmentation de l'hémisphère gauche pour obtenir le support anatomique des sources gauches. La segmentation de la peau à partir d'une IRM peut servir au recalage des électrodes ou au modèle BEM.

Ces positions de référence sont enregistrées dans le fichier indiqué par le champs 'electrodesCoordMat' de l'interface illustrée figure 2.3. Ainsi, en utilisant un recalage de type Procruste¹⁹, les coordonnées des électrodes seront enregistrées dans l'espace de voxels IRM.

¹⁹ Recalage solide appliquant des rotations et des translations en $\bar{x} \bar{y} \bar{z}$. Le recalage procruste minimise l'écart entre un trio de points de référence d'un objet et le trio homologue d'un autre objet.

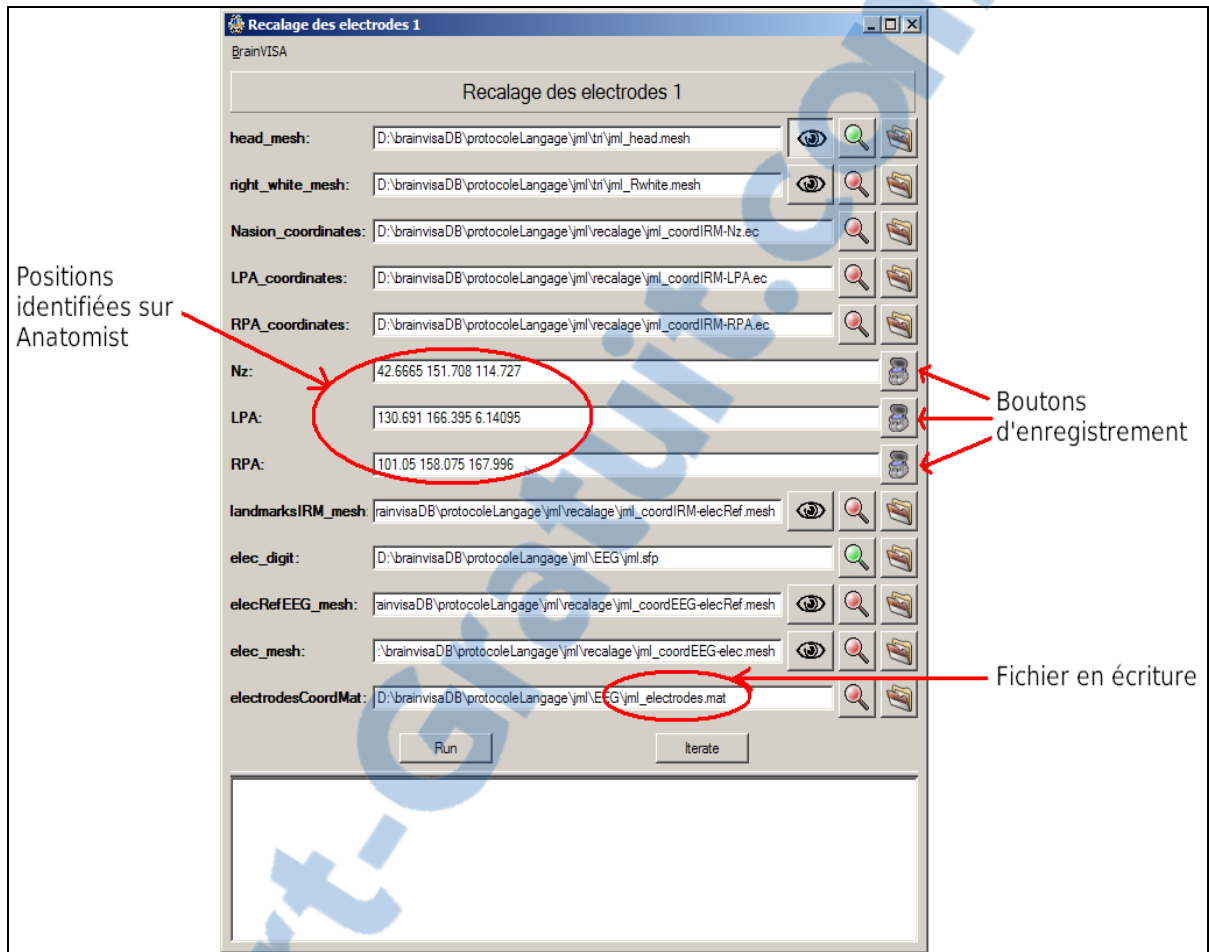


Figure 2.3 Interface de commande du recalage des électrodes permettant de placer la position des électrodes d'un repère relatif vers le repère volumique de l'IRM anatomique.

2.4.4 Étape 3 : Modèle distribué, suppression d'électrodes, matrice de Gain et génération de fichiers pour BrainEntropy.

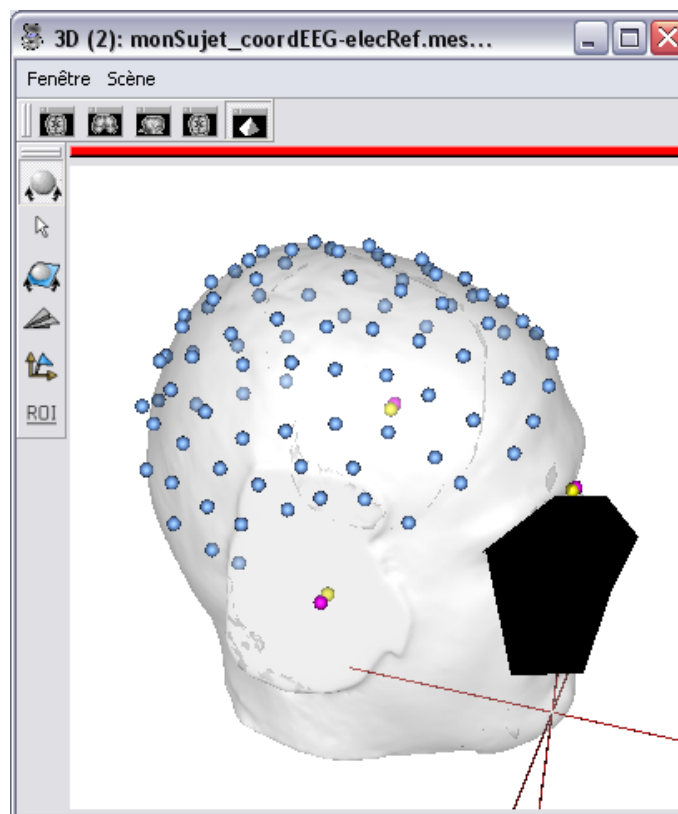


Figure 2.4 Étape 2 : Électrodes recalées sur la surface de la peau. La méthode utilisée minimise l'écart entre trois électrodes de référence (boules jaunes) appartenant au système d'électrodes et trois points de référence (boules roses) positionnées sur le nasion et aux points péri auriculaires.

Nous allons détailler dans l'étape 3 les paramètres modifiables de l'interface illustrée figure 2.5. Les fichiers de sortie de ces étapes permettront de résoudre le problème inverse.

a) La valeur du champs "decimation_factor_faces"

La valeur du champs "decimation_factor_faces" influence directement la quantité de faces du maillage du cerveau qui définira l'espace des sources. En effet, l'algorithme procède à un sous échantillonnage des maillages issues de la segmentation précédente ("left_white_mesh", "right_white_mesh"). Plus la valeur de ce champs est grande, plus le nombre de sources sera

élevé. Pour avoir un ordre d'idée, quand "decimation_factor_faces" est à 20000, on obtient un maillage de 20000 faces et environ 10000 nœuds. L'espace des sources contiendra donc environ 10000 dipôles, ce qui demandera une puissance de calcul conséquente dans la résolution du problème inverse. Un fichier Matlab de sortie contenant le maillage des sources est archivé dans le fichier indiqué par le champs "white_mat_decimed".

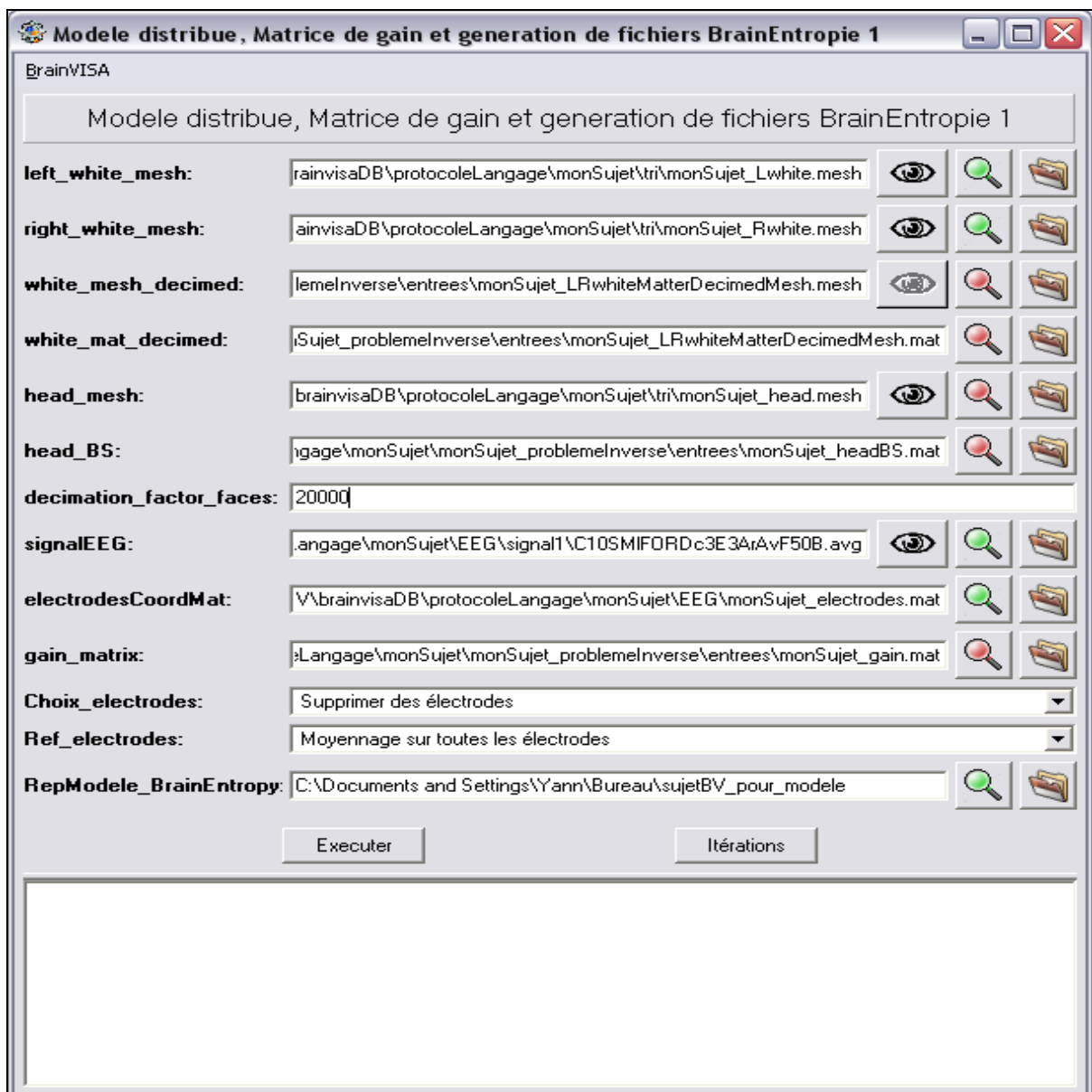


Figure 2.5 Étape 3 : Module de BrainVISA permettant d'obtenir le maillage des sources, les informations des capteurs et la matrice de Gain EEG.

b) Choix des électrodes

Certaines électrodes sont parfois dysfonctionnelles ou n'ont pas la fonction de mesurer l'activité cérébrale. Dans ce cas, elles ne doivent pas être prises en compte dans le calcul de la matrice de gain. Le champs "Choix_electrodes" permet de choisir de conserver toutes les électrodes ou d'en supprimer de la modélisation. L'application demande à l'utilisateur quelles sont les électrodes à supprimer du problème. Après la sélection des capteurs qui ne mesurent pas l'activité cérébrale et la fermeture de la fenêtre représentée figure 2.6, le calcul de la matrice de gain se fera sans les capteurs indiqués.

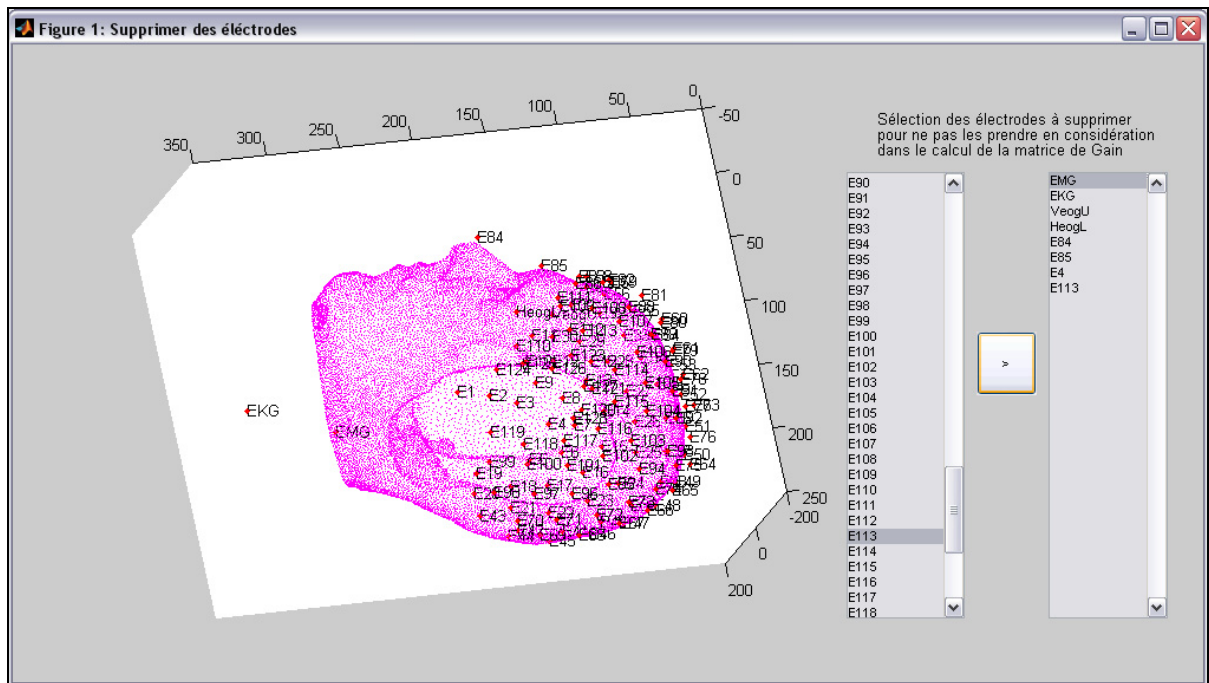


Figure 2.6 Fenêtre de suppression des électrodes non désirables du modèle. Leurs signaux ne seront pas pris en compte lors de la localisation.

La procédure de cette étape compare les capteurs restants (supposés tous mesurer l'activité cérébrale) à la structure 'Channel' contenant la localisation, l'orientation, le type et le nom de toutes les électrodes. Cette structure 'Channel' provient du fichier de sortie de l'étape 2 qui est indiqué par le champs "electrodesCoordMat" dans l'interface utilisateur (figure 2.5).

La comparaison permet d'enregistrer une nouvelle structure 'Channel' épurée des capteurs éliminés par l'utilisateur dans un fichier nommé 'monSujet_Capteur.mat'. De même, on crée un fichier 'monSujet_Data.mat' contenant les mesures des capteurs sélectionnés à partir du fichier de mesures original (d'extension .avg ou .mat dépendamment du centre qui a acquis les données). 'monSujet_Capteur.mat' et 'monSujet_Data.mat' sont archivés dans le répertoire du sujet en cours, mais leurs chemins ne sont pas indiqués par la figure 2.5.

c) Choix de la référence des électrodes et moyenne de la matrice de gain

L'EEG mesure des différences de potentiels (tension en Volt) à la surface de la tête. Une différence de potentiel nécessite forcément deux points de mesures. Une électrode constitue l'un de ces points et il existe plusieurs façons de déterminer le deuxième qui est la référence. Dans la figure 2.5, le champs "Ref_electrodes" propose à l'utilisateur deux types de référence des électrodes :

- 1) une électrode particulière est la référence;
- 2) la référence est la moyenne de toutes les électrodes.

Le choix est important pour moyenner correctement la matrice de gain calculée par BrainVISA. BrainVISA permet de calculer des matrices de gain EEG et MEG que nous utiliserons pour les comparer avec celles d'OpenMEEG. Ici BrainVISA n'utilise pas la méthode BEM, mais il considère la tête comme un milieu sphérique et homogène. C'est donc la matrice de gain EEG issue du calcul de BrainVISA (Gtmp) qu'il va falloir moyenner en fonction de la référence des électrodes. Le fichier contenant la nouvelle matrice de gain moyennée est indiqué par le champs "gain_matrix" de la figure 2.5.

c.1) Une électrode particulière est la référence

Cette approche utilise une électrode particulière, dite de référence. On utilise en général une électrode située au sommet de la tête (Cz) ou placée sur le nez²⁰. Le potentiel des électrodes est référencé par rapport à une électrode d'indice *ref* :

$$G = \begin{bmatrix} Gtmp_{1,1} - Gtmp_{ref,1} & \cdots & Gtmp_{1,N_s} - Gtmp_{ref,N_s} \\ \vdots & \ddots & \vdots \\ Gtmp_{N_c,1} - Gtmp_{ref,1} & \cdots & Gtmp_{N_c,N_s} - Gtmp_{ref,N_s} \end{bmatrix} \quad (2.1)$$

c.2) La référence par rapport à la moyenne

Cette référence permet d'harmoniser le résultat entre les électrodes selon l'activité globale enregistrée. On soustrait à chaque forwardfield la moyenne de ce même forwardfield :

$$G = \begin{bmatrix} Gtmp_{1,1} - \overline{G}_1 & \cdots & Gtmp_{1,N_s} - \overline{G}_{N_s} \\ \vdots & \ddots & \vdots \\ Gtmp_{N_c,1} - \overline{G}_1 & \cdots & Gtmp_{N_c,N_s} - \overline{G}_{N_s} \end{bmatrix} \quad (2.2)$$

avec N_C le nombre de capteurs, N_S le nombre de sources et

$$\overline{G}_j = \sum_{k=1}^{N_C} Gtmp(k, j) / N_C \quad (2.3)$$

On force ainsi la somme des facteurs de gain d'une source à être nulle.

$$\sum_{i=1}^m G(i, j) = 0 \quad (2.4)$$

²⁰ Il s'agit d'un choix qui dépend du type d'activité mesurée.

Comme la matrice de gain d'OpenMEEG est référencée par rapport à la moyenne des électrodes, nous utiliserons aussi cette référence sur la matrice de gain calculée par BrainVISA afin de les comparer en utilisant la même modélisation.

2.5 Résumé de l'utilisation de BrainVISA

La librairie 'création d'un modèle EEG' développé dans l'environnement BrainVISA, permet de transformer l'information originale des mesures et de l'IRM dans le but d'effectuer une localisation de sources. L'étape 3 de cet outil réalise des fichiers contenant toute l'information exigée pour calculer le problème inverse. Nous pouvons maintenant considérer un modèle contenant : a) le maillage du cortex, b) les informations des capteurs utiles, c) les mesures avec la fréquence d'échantillonnage, d) la matrice de gain associée aux capteurs et au maillage. Le récapitulatif des fichiers Matlab de sortie est illustré figure 2.7 et le procédé pour les obtenir est illustré figure 2.8.

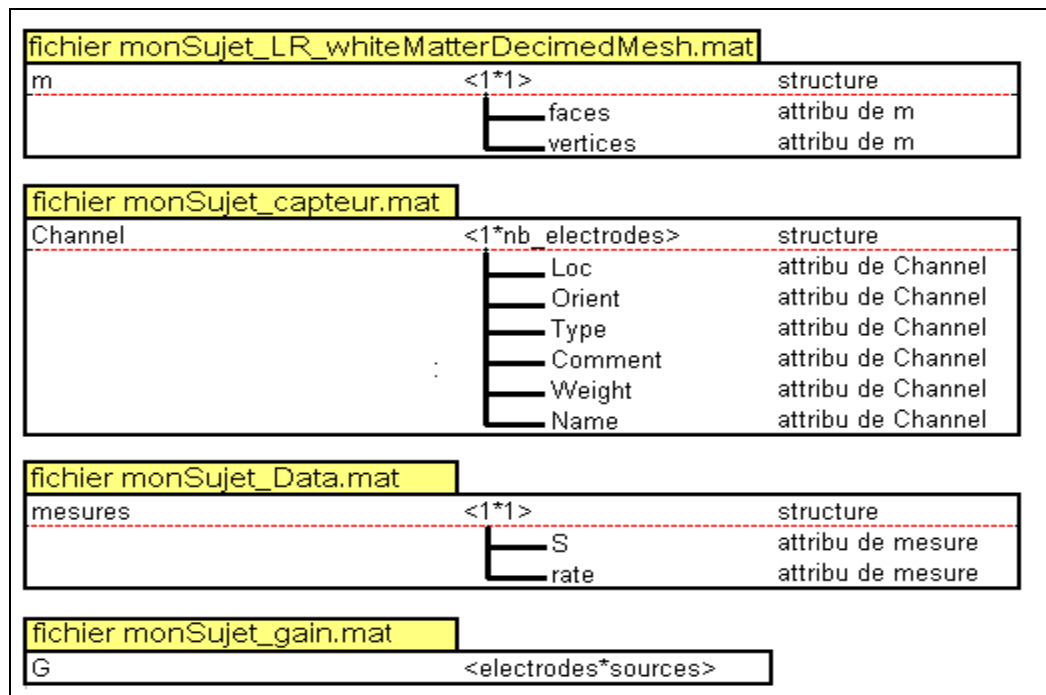


Figure 2.7 Structures des fichiers de sortie de l'outil qui construit un modèle EEG: le maillage de surface corticale, les capteurs, les mesures, la matrice de gain EEG.

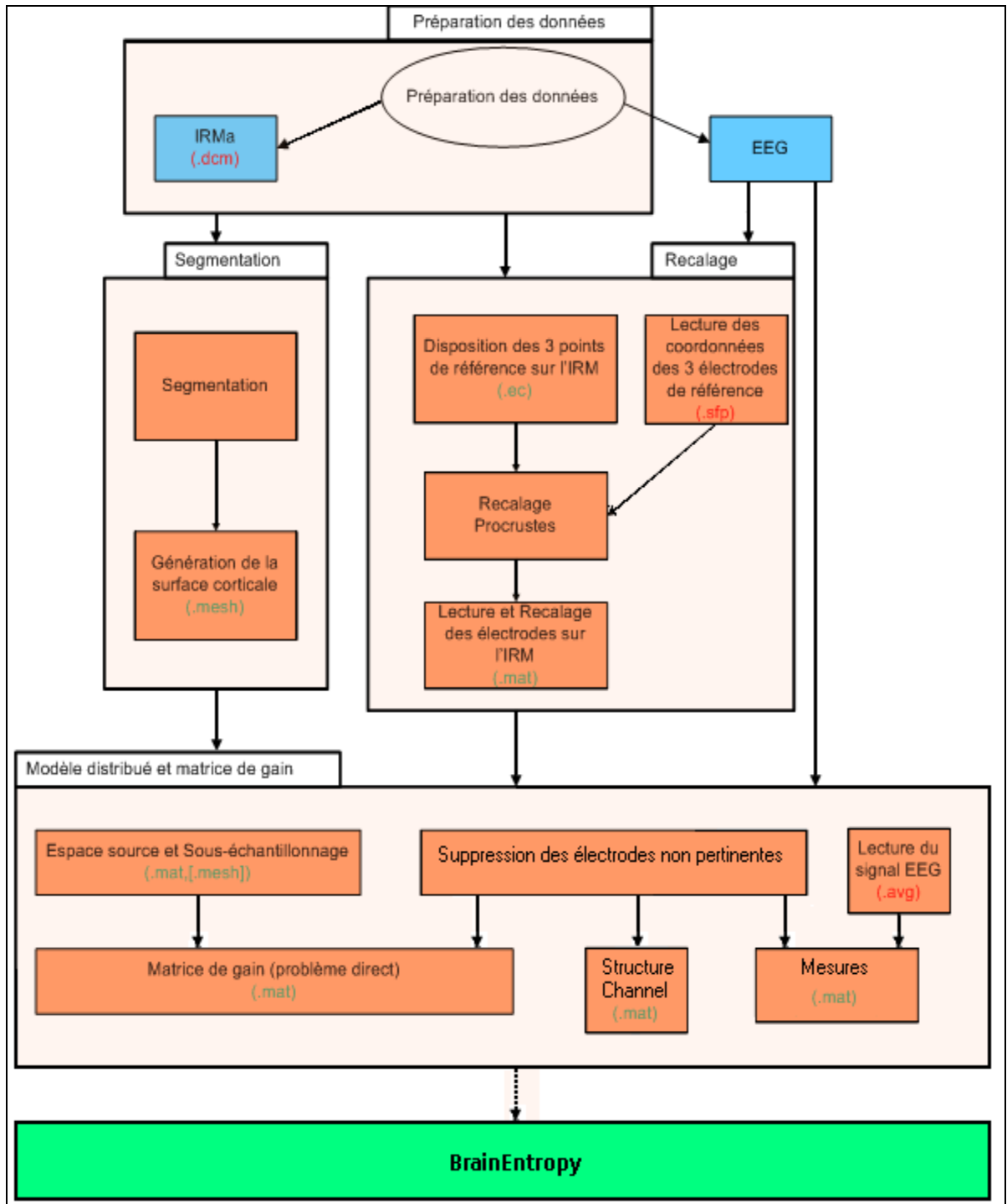


Figure 2.8 Organigramme des différents modules de l'outil « création d'un modèle EEG ». À partir d'une IRM anatomique le processus consiste à segmenter et décimer les maillages du cortex et de la peau. À partir des électrodes qui mesurent l'activité cérébrale, l'outil recale ces électrodes vers le repère de l'IRM permettant ainsi le calcul d'une matrice de gain.

2.6 BrainSuite2

2.6.1 Présentation du logiciel BrainSuite2

BrainSuite2 est présenté dans ce chapitre car ce logiciel permet d'obtenir les maillages délimitant les tissus biologiques de la tête utile à la création du modèle direct. Ce logiciel permet de modéliser et visualiser la tête comme un empilement de plusieurs tissus. Toutes les surfaces générées par BrainSuite2 seront utilisées dans la méthode BEM d'OpenMEEG (voir chapitre 1). Quoiqu'il en soit, il existe plusieurs outils en neuro-imagerie pour faire de la segmentation de tissus comme SPM, ITK, BrainVISA, FSL/BET, Freesurfer, Animals, Civet. Cependant, on a choisi BrainSuite2²¹ car ce logiciel segmente le crâne facilement et rapidement, là où les autres ne le font pas ou difficilement. Développé en 2005 par David W. Shattuck à Los Angeles, ce logiciel fonctionne sous Windows XP et 2000. Une version en ligne de commande existe pour les autres OS. Brainsuite 2 utilise les outils OpenGL des cartes graphiques, ce qui offre une visualisation 3D très agréable. Nous allons donc dédier la fin de ce chapitre 2 à ce logiciel capable de créer les maillages suivants :

- Surface MB/MG
- Surface MG
- Crâne intérieur
- Crâne extérieur
- Peau

2.6.2 Lecture et visualisation du volume IRM

Deux types de formats de fichiers de volume IRM sont lisibles par BrainSuite2. Pour lire le format analyze 7.5, il faut ouvrir un fichier avec l'extension *.img. (File > Open Volume...). Les fichiers *.img sont accompagnés d'un fichier header *.hdr comportant certaines caractéristiques du volume comme le nombre de voxels sur les trois axes, ou leur taille en mm. Il est aussi possible d'importer des fichiers de volume en format DICOM. Dans la

²¹ Les fichiers d'installation de BrainSuite 2 sont disponibles à l'adresse : <http://brainsuite.usc.edu>

sélection du fichier (File > Import DICOM*...), il est important de sélectionner tous les fichiers relatifs aux tranches. Notons que parfois le lecteur de fichier DICOM ne fonctionne pas correctement, à cause de l'orientation des axes. Dans ce cas, on doit convertir les tranches en "format analyse" avec le "convertisseur Aims" de BrainVISA. Une fois le volume IRM chargé dans Brainsuite2, la visualisation se fait automatiquement dans les vues axiale, coronale et sagittale. Pour une représentation 3D on peut utiliser un outil 3d : Tool > Surface Viewer.

2.6.3 Orientation du volume

Une des difficultés avec les images IRM est l'absence d'orientation standard et universelle des volumes dans l'enregistrement des données. En effet, l'axe \bar{x} dans le repère IRM peut présenter l'axe axial, coronal ou sagittal dépendamment de l'imageur et de la convention choisie par le technicien au cours de l'acquisition. BrainSuite2 possède un outil capable de réorienter les volumes avant de procéder à un quelconque traitement. Ceci est utile si l'on désire, par exemple, segmenter la peau, le crâne, le cortex, dans le même référentiel qu'un maillage de sources segmenté par un autre logiciel. Dans le cas d'une segmentation de la surface corticale, Brainsuite2 utilise un masque du cervelet préalablement orienté. Afin d'obtenir une bonne segmentation il est nécessaire d'orienter le volume comme ce masque avant de commencer tout processus de traitement²². Cet outil se trouve dans Tools > Reorient Volume.

2.6.4 Segmentation de surfaces

La segmentation de surfaces offerte par BrainSuite2 est le principal atout de ce logiciel. Cette procédure, outre la simplicité de son utilisation, est relativement rapide. La première étape de la segmentation des tissus consiste à enlever le crâne et la peau de manière à ne conserver

²² Pour segmenter la surface corticale, il faut orienter le volume comme l'illustre la figure 2.8, ou réorienter le masque.

que le cerveau. Ceci s'exécute en cliquant sur BrainSuite > Skull stripping (BSE). La fenêtre illustrée sur la figure 2.9 apparaît.

Plusieurs facteurs sont modifiables dans la première étape d'extraction du cortex. Sans entrer dans les détails de la signification de chaque paramètre, on conseille, si le "skull stripping" pose problème, de baisser le coefficient 'Diffusion Constant' quitte à le mettre à zéro. Il est possible de lancer le processus sous-étapes par sous-étapes en cliquant sur 'Next >', ou bien sans interruption en utilisant 'Do All >>'. Le résultat devrait ressembler à la figure 2.10 avant de passer à la prochaine étape. Deux options sont maintenant possibles. Soit effectuer une génération de surfaces des différentes frontières entre les tissus de la tête en appuyant sur 'Scalp & Skull'. Soit extraire les surfaces corticales de la MG ou l'interface MG/MB en appuyant sur 'BFC'. Puisque nous avons déjà vu comment segmenter la surface corticale avec BrainVISA, nous n'entrerons pas dans les détails ici, mais on peut se référer au tutorial (Help > BrainSuite2 tutorial).

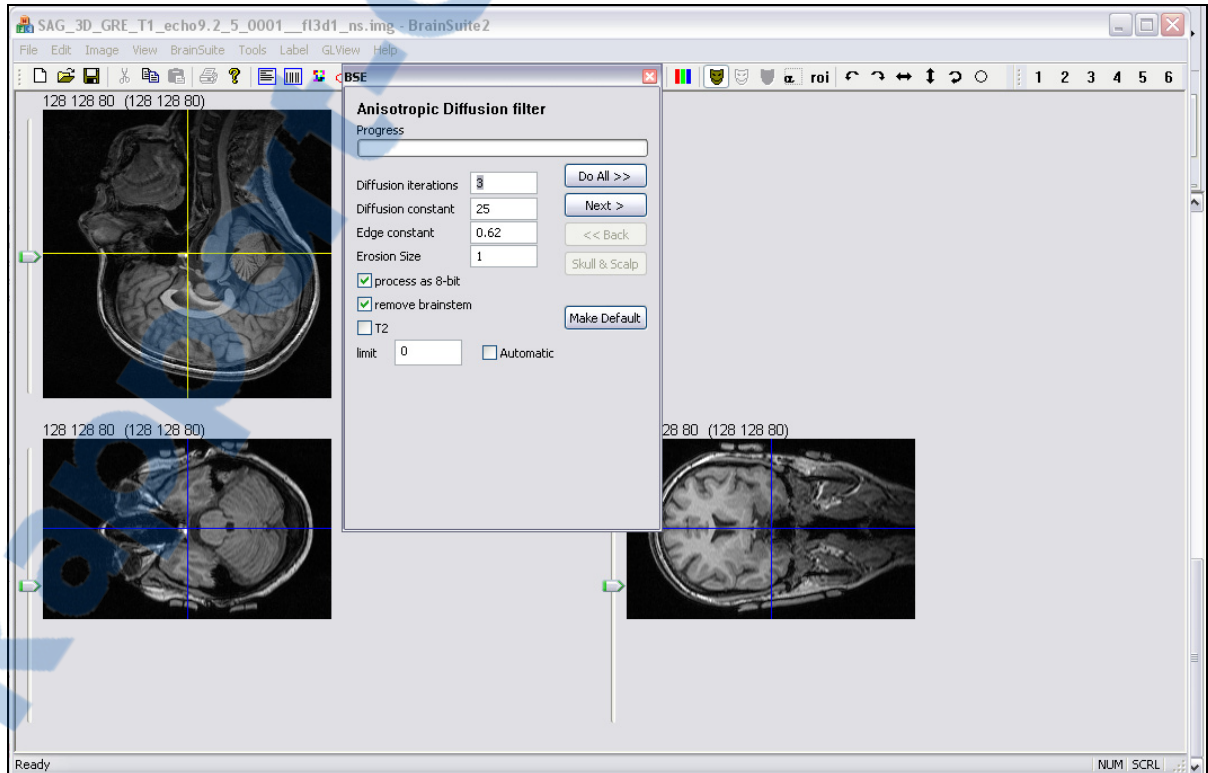


Figure 2.9 Illustration du « skull stripping » qui est la première étape du logiciel pour obtenir les surfaces de tissus anatomiques.

Puis dans la fenêtre 'Skull and Scalp' (2.11) appuyer sur 'Do All' sans toucher aux paramètres. BrainSuite2 va alors extraire les différentes surfaces à partir du volume de l'IRM. Le résultat s'affiche alors comme dans la figure 2.12 et on archivera par la suite ces données pour qu'elles soient prises en entrée du logiciel OpenMEEG grâce à la fenêtre 'view surface properties' illustrées figure 2.13.

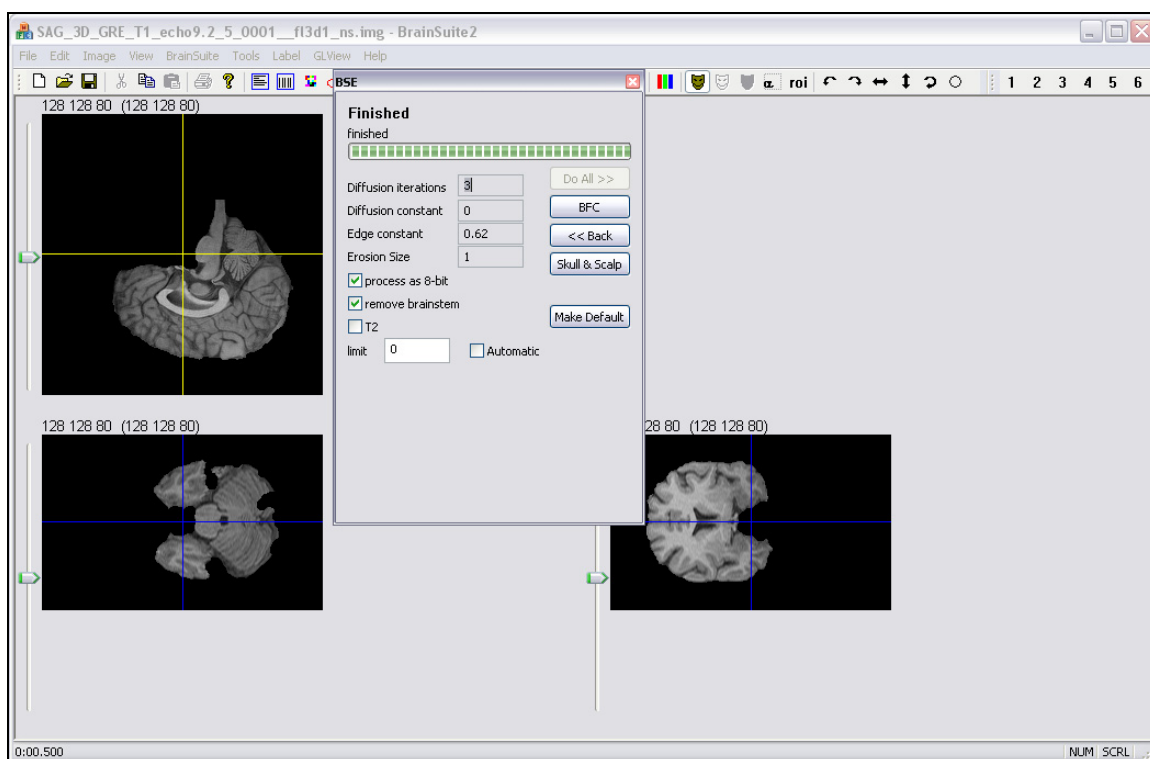


Figure 2.10 « Skull stripping » effectué : le crane et la peau ont été enlevés.

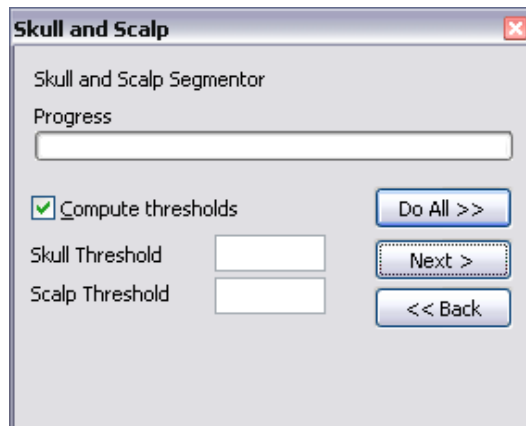


Figure 2.11 Fenêtre «Skull and Scalp».

Quand l'opération sera terminée, le logiciel affichera dans une fenêtre OpenGL la surface extérieure de l'os segmenté. Pour voir et enregistrer les surfaces, il faut appeler le 'Surface Display Properties' en faisant un clic droit dans la nouvelle fenêtre 3D.

La fenêtre du 'Surface Display Properties' fournit plusieurs options pour manipuler les surfaces. Cela inclut des paramètres comme la visualisation et manipulation des surfaces, la couleur de l'arrière plan et autres ... Pour contrôler un aspect spécifique d'une surface, sélectionnez la dans la liste de surfaces et changez les boîtes de sélection sur la droite.

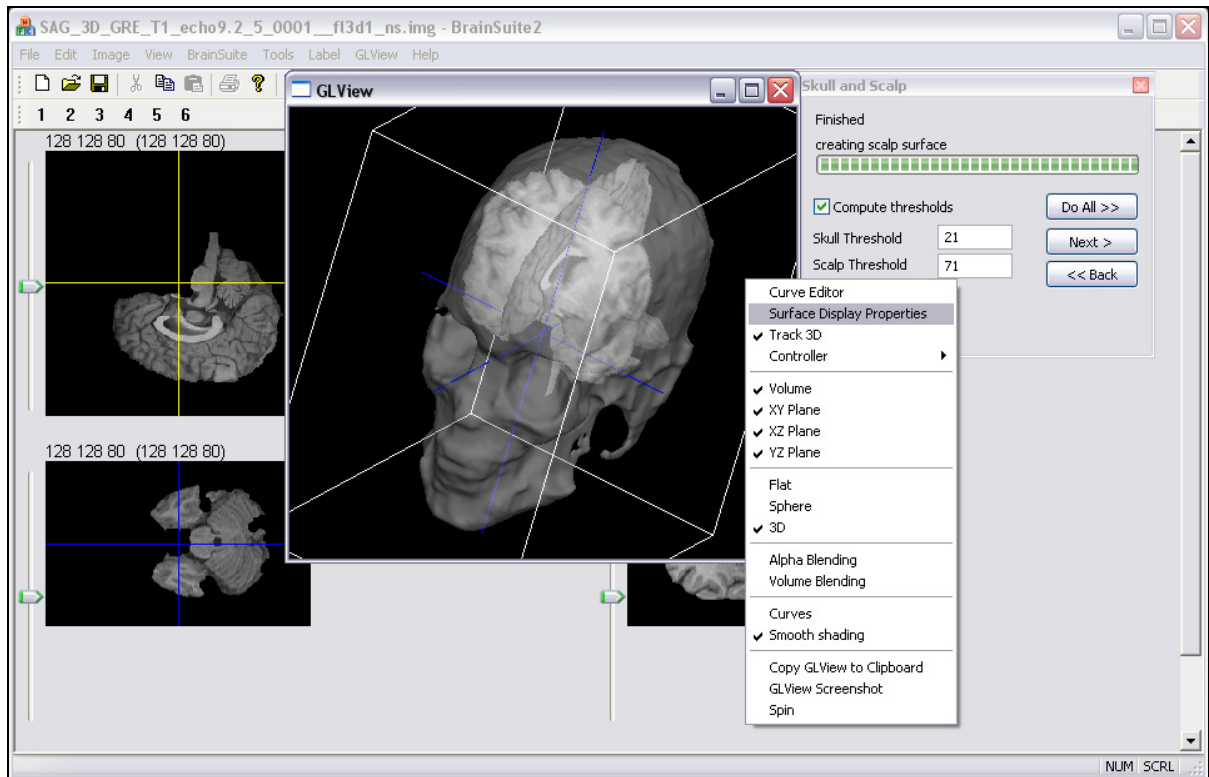


Figure 2.12 « Skull stripping » accompli.

Vous pouvez sauvegarder une surface dans le format dfs en la sélectionnant dans la liste, et en appuyant sur le bouton 'save'. Attention : Il est nécessaire d'écrire l'extension *.dfs à la fin du nom de fichier sinon BrainSuite2 ne sauvegarde rien !

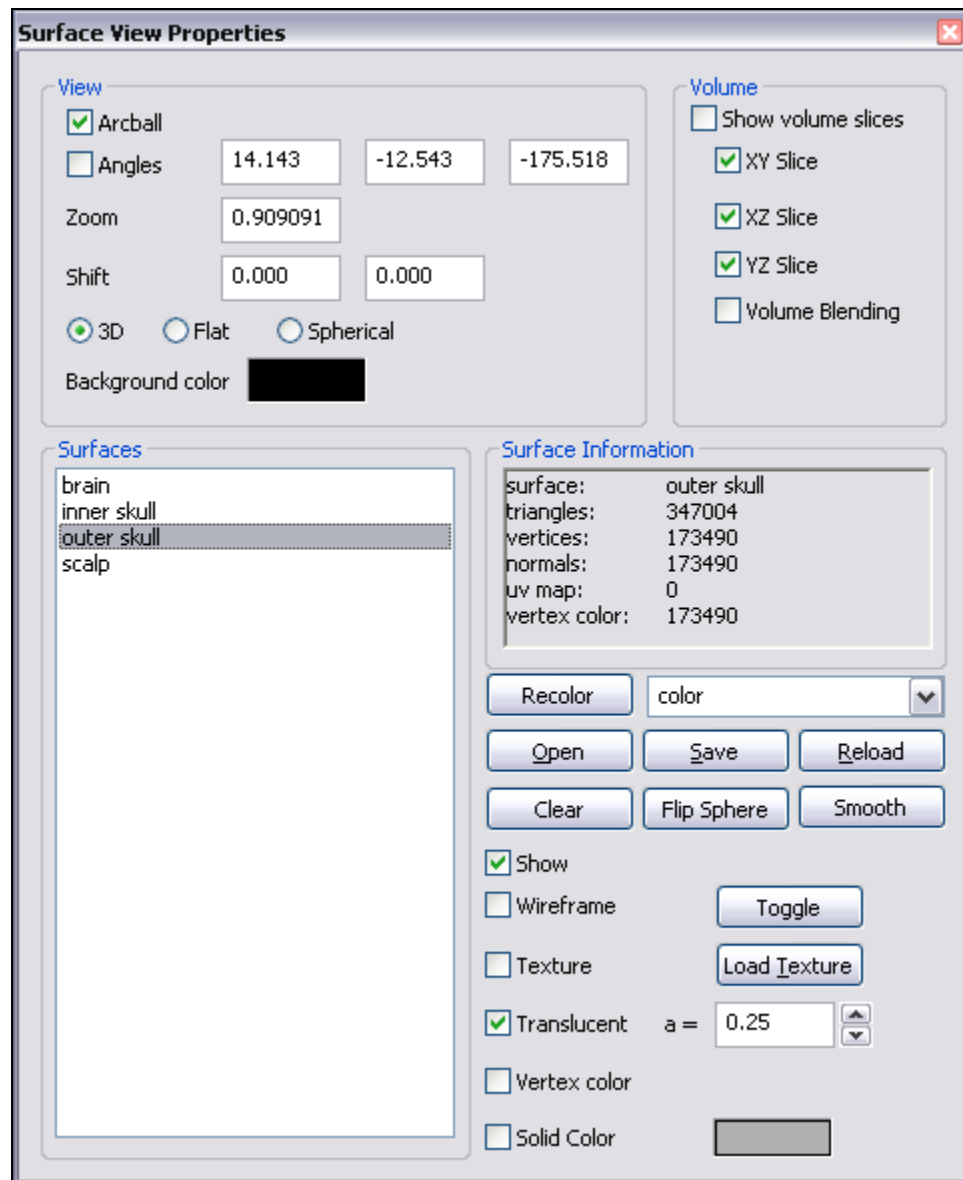


Figure 2.13 La fenêtre « Surface View Properties » permet de contrôler la visualisation des différentes surfaces et de les enregistrer avec l'extension .dfs pour s'ouvrir avec BrainSuite2.

La figure 2.14 illustre un exemple des quatre surfaces générées par BrainSuite2.

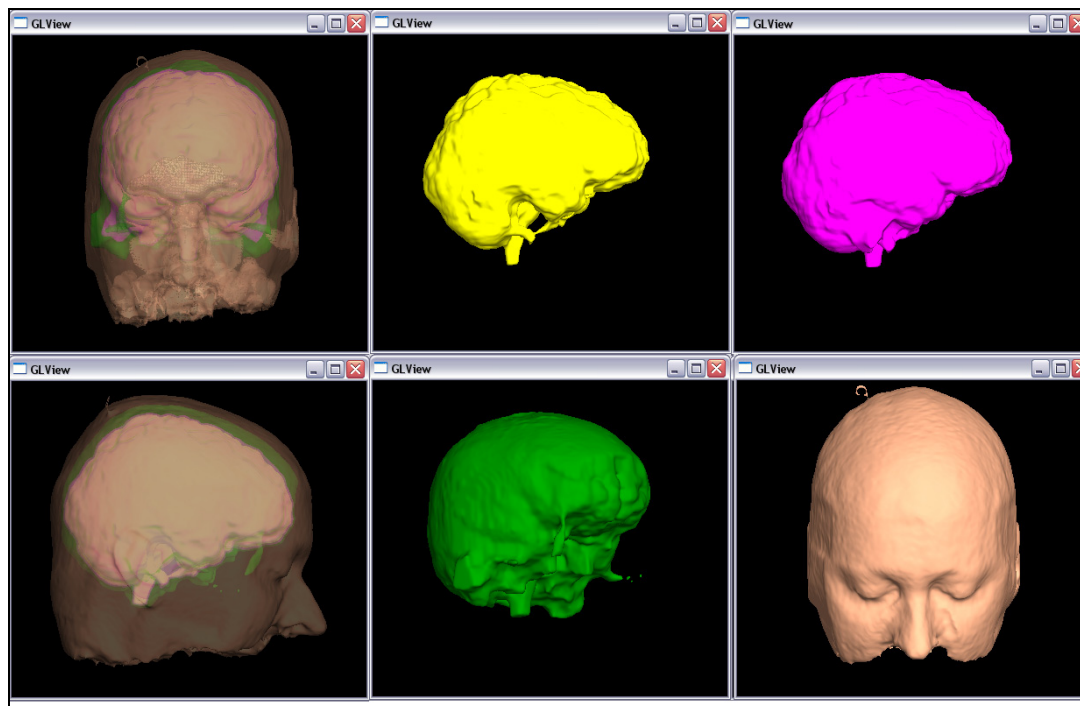


Figure 2.14 Affichage des quatre surfaces générées : Peau, crâne extérieur, crâne intérieur, cortex.

2.7 Conclusion sur le chapitre 2

Ce chapitre 2 illustre deux outils capables d'enregistrer des fichiers contenant l'information essentielle pour la modélisation du système électromagnétique de la tête.

Même si les fichiers originaux (IRM, mesures) ne sont pas toujours dans le même format selon les centres d'acquisition, le module de conversion de BrainVISA, permet de travailler dans le format de données exigé par un logiciel de prétraitement. Dans les prochains chapitres nous verrons comment exploiter le modèle résultant des manipulations étudiées ici pour effectuer une localisation de sources.

CHAPITRE 3

PROBLÈME INVERSE ET LES MÉTHODES FOCCUS

Dans le contexte de la neuro-imagerie, on s'intéresse ici à utiliser des méthodes de résolution du problème inverse afin d'estimer l'emplacement de l'activité cérébrale dans la matière grise. La résolution du problème inverse consiste à estimer l'intensité des sources à partir de mesures connues. Cependant, ce problème est mal posé car il présente de nombreuses difficultés liées à sa taille²³ et à sa non-linéarité²⁴ due au fait qu'il existe plusieurs solutions expliquant les mesures. La résolution du problème inverse nécessite alors d'ajouter des contraintes de régularisation, a priori, pour que le problème se formule comme un problème d'optimisation et d'estimation de paramètres.

Ainsi, une fois la distribution spatiale des sources définie et la matrice de gain calculée (après recalage des électrodes) nous pourrions estimer l'amplitude des sources qui produira une image définie sur la surface du cortex. Cette approche est de type "imaging" à l'opposé de techniques statistiques. Le problème étant mal posé, il est nécessaire de mettre des contraintes additionnelles pour réduire le nombre de solutions expliquant les données et stabiliser les solutions. Ces contraintes additionnelles sont introduites ici selon trois approches : l'approche Bayésienne, avec l'approche régularisée L_p et l'approche par minimum d'entropie de Shannon. Ces deux dernières approches sont expliquées et extraites de la publication [6].

3.1 Complémentarité EEG et MEG

La combinaison des signaux MEG et EEG dans un même problème inverse permet de mieux exploiter la complémentarité entre les deux jeux de données et de restreindre les ambiguïtés sur les solutions du problème inverse. La fusion des deux modalités permet à la fois d'estimer des zones profondes, mesurables en EEG et difficilement détectables en MEG, et

²³ Estimation de l'intensité d'environ 10000 sources pour seulement 300 capteurs MEEG.

²⁴ Problème dont la solution n'existe pas, ou si elle existe, n'est pas unique ou ne dépend pas de façon continue des données.

d'obtenir des régions d'activations plus focales accentuées par la participation des mesures en MEG.

Selon la loi de Biot et Savart, on sait que l'intensité du champ magnétique décroît avec le carré de la distance entre le capteur et le dipôle. Ainsi le champ magnétique des sources profondes décroît plus vite que le potentiel, et leurs contributions sont donc plus importantes en EEG que en MEG. De plus, les sources radiales (sur les gyri) créent un champ magnétique à peine perceptible à cause de l'orientation des capteurs MEG, mais elles produisent un potentiel électrique élevé représenté par un pôle électrique négatif au dessus de la source apparaissant sur le scalp.

Le champ magnétique n'est guère affecté par la matière biologique offrant une meilleure résolution spatiale avec des réponses plus focales qu'en EEG. Le nombre élevé de capteurs MEG participe aussi à une meilleure résolution spatiale. En général, le potentiel électrique sur le scalp est très diffus. Cette diffusion est principalement causée par les courants extracellulaires qui sont très sensibles aux conductivités des différents tissus cérébraux, notamment au niveau de l'os du crâne qui engendre une grande diffusion des lignes de potentiel qui le traverse. Dans la figure 4.10 nous constatons que le réseau activé reconstruit en MEG est plus restreint que celui reconstruit en EEG. Ainsi il est plus facile de localiser avec précision des sources en MEG qu'en EEG et la fusion des deux modalités permet donc de mieux discerner l'activité trop diffuse estimée en EEG.

La MEG et l'EEG donnent des réponses complémentaires avec des informations distinctes provenant des sources selon leur position et leur orientation. Toutes ces caractéristiques justifient la nécessité d'avoir des enregistrements MEG et EEG simultanés lors d'un même protocole. Il sera alors possible de mieux détecter les contributions de l'ensemble du cerveau. Ainsi la fusion des modalités MEG et EEG fait l'objet d'une nouvelle contrainte dans la résolution du problème inverse.

3.2 La Formulation Bayésienne du problème inverse

Les mesures MEEG sont modélisées comme la somme du signal obtenu par le problème direct additionné d'un bruit physiologique et d'une erreur de mesure. Ces "bruits" sont contenus dans une matrice de bruit N de dimension nombre de capteurs par temps.

$$M = GS + N \quad (3.1)$$

L'association des matrices de gain en EEG et en MEG est faite dans ce projet de maîtrise selon la méthode "row" qui consiste à concaténer en ligne les deux matrices regroupant ainsi les caractéristiques électromagnétiques du système. Les notations sont les suivantes :

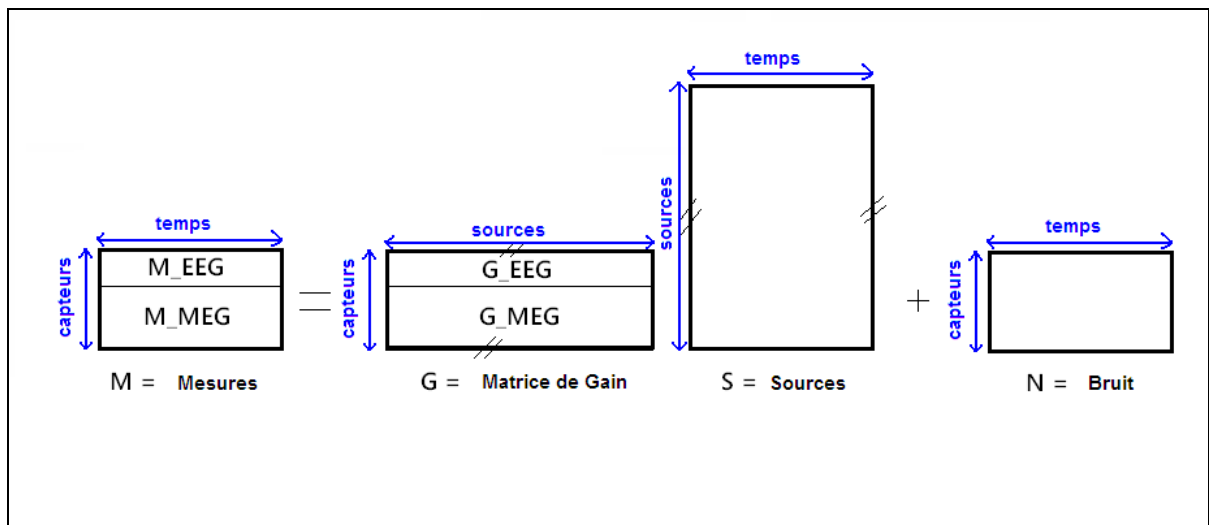


Figure 3.1 Équation matricielle du problème direct. Les mesures (M) en EEG et en MEG sont expliquées, via la matrice de gain (G), par l'activité des dipôles représentée par la matrice des sources (S) et par un bruit physiologique (N).

Soit N_C le nombre de capteurs;

N_S le nombre de sources;

T le temps;

M est la matrice de mesure de taille $N_C * T$.

G la matrice de Gain de taille $N_C * N_S$.

S la matrice des amplitudes des sources $N_S * T$.

N le bruit caractérisant les fluctuations apportées sur la matrice des mesures ne provenant pas des sources $N_C * T$.

Dans une approche probabiliste, les intensités des sources et celles des capteurs sont décrites par une loi de probabilité jointe $P(S, M)$. Si les mesures (M) étaient indépendantes des sources, nous aurions $P(S, M) = P(S)P(M)$. Toutefois, les mesures sont en partie expliquées par les sources et nous avons plutôt $P(S, M) = P(M/S)P(S)$ où $P(M/S)$ s'appelle la loi conditionnelle de M connaissant S . $P(S)$ est la loi de probabilité de connaître les sources. Nous pourrions écrire aussi $P(S, M) = P(S/M)P(M)$ et par conséquent

$$P(S/M) = \frac{P(M/S)P(S)}{P(M)} \quad (3.2)$$

Ce résultat, connu sous le nom de formule de Bayes, permet de calculer $P(S/M)$, qui est en fait la loi de probabilité d'une configuration de sources connaissant les mesures, autrement dit cela correspond au problème inverse.

a) $P(M/S)$ désigne la loi probabilité des mesures connaissant les sources, ce que modélise le problème direct. On appelle encore $P(M/S)$ la vraisemblance, car comme les sources sont fixées, avec (3.1) on voit que les mesures fluctuent avec le bruit. En supposant un bruit gaussien blanc dans le temps, dans l'espace et de moyenne nulle on a l'expression de la vraisemblance suivante :

$$P(M/S) = e^{-\frac{N^2}{2\sigma^2}} = e^{-\frac{(M-GS)^2}{2\sigma^2}} \quad (3.3)$$

La log-vraisemblance est donnée par :

$$\ln(P(M / S)) = -\frac{1}{2\sigma^2} \|M - GS\|^2 \quad (3.4)$$

b) $P(S)$ est une probabilité a priori sur les sources que l'on spécifie comme ne pouvant pas être nulle. En effectuant un changement de variable sur ce a priori, on obtient une fonction de régularisation $f(S)$ que l'on peut noter pour l'instant sous la forme :

$$P(S) = e^{-\lambda f(S)} \quad (3.5)$$

La configuration de sources qui maximise cette loi est celle qui satisfait $f(S) = 0$.

c) $P(M)$ s'appelle l'évidence qui est en fait une loi de probabilité connue qui sera négligée dans la suite du raisonnement.

Dans ce cadre Bayésien, la solution MAP (maximum a posteriori) consiste à trouver S^* qui maximise la probabilité a posteriori $P(S / M)$ ou ce qui revient au même, son logarithme :

$$S^* = \arg \max \ln(P(M / S)P(S)) = \arg \max \ln(P(M / S)) + \ln(P(S)) \quad (3.6)$$

En tenant compte de (3.4) et de (3.5), la maximisation de (3.6) peut alors être vue comme la minimisation de l'énergie $U(S)$:

$$U(S) = \|M - GS\|^2 + \lambda f(S) \quad (3.7)$$

λ est un coefficient qui pondère le rôle de la contrainte " $f(S) = 0$ ". Sous cette forme, (3.7) peut être vue comme la forme régularisée du problème inverse, $f(S)$ étant la régularisation de Tikhonov et λ le paramètre de régularisation. Un λ petit donnera plus d'importance à l'espace des mesures, ainsi le biais sur les sources sera augmenté mais la variance (reconstruction des mesures) sera réduite. Inversement si λ est grand alors on privilégiera

l'espace des sources, ceci apportera une variance forte mais un biais sur les sources plus faible. λ fait donc le compromis entre $f(S) = 0$ et $\|M - GS\|^2 = 0$.

Le problème inverse sous sa forme bayésienne se ramène donc au problème de la minimisation de l'énergie $U(S)$. Pour ce faire il existe des méthodes de résolution linéaire et non linéaire.

3.3 Résolution linéaire

Dans le cadre linéaire, on suppose que la distribution a priori de chaque image des sources est gaussienne, centrée, de matrice de covariance C_s . On suppose, de plus, que les intensités des sources sont temporellement indépendantes. Dans ce cas on a :

$$f(S) = \text{trace}(S^t C_s^{-1} S) \quad (3.8)$$

avec la fonction *trace* qui est la somme des éléments de la diagonale. L'énergie $U(S)$ est alors quadratique par rapport à S car elle est la somme de deux fonctions quadratiques. La minimisation de l'énergie est donnée par la formule exacte de l'estimation MAP :

$$S^* = C_s^{-1} G^t (GG^t + \lambda I)^{-1} M \quad (3.9)$$

C'est selon cette interprétation qu'un certain nombre de propositions ont été faites pour la métrique C_s^{-1} . L'approche la plus ancienne qui utilise $C_s^{-1} = I$ calcule la solution de norme minimale (diagonale). Le problème se simplifie par la minimisation de

$$S^* = G^t (GG^t + \lambda I)^{-1} M \quad (3.10)$$

Dans (3.7) nous trouvons deux termes : $\|M - GS\|_F^2$ qui représente l'erreur des sources par rapport aux mesures (bruit); et $f(S)$ qui représente l'erreur des sources par rapport au modèle construit avec la métrique C_S .

Pour avoir un système équilibré, il est intéressant de connaître la valeur de λ où les deux erreurs sont minimales. Le λ optimal est celui pour lequel on trouve une distance euclidienne minimale entre le point d'origine et l'un des points qui forment la courbe en L de la figure 3.2.

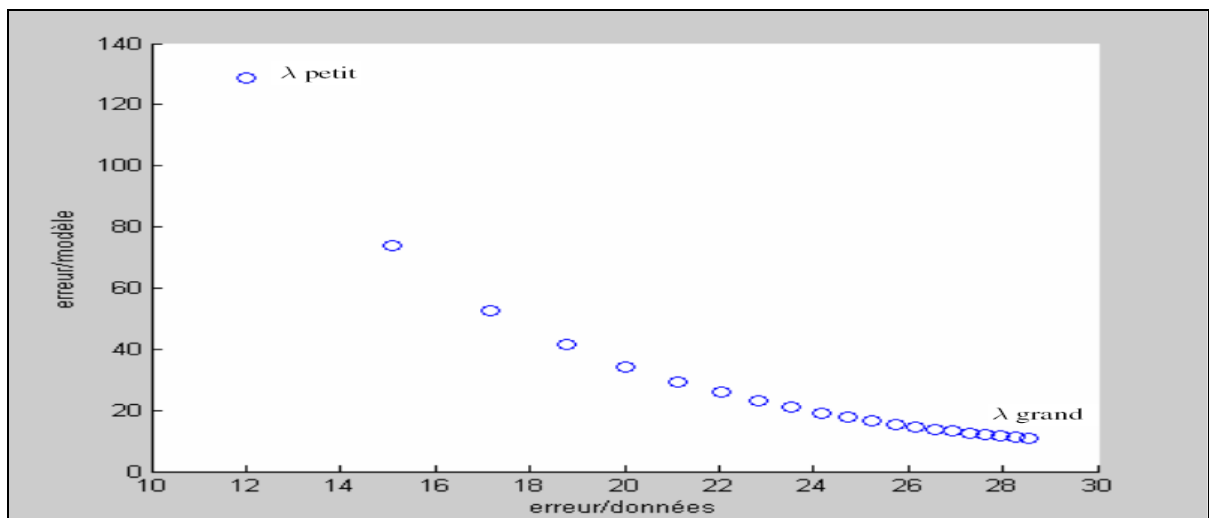


Figure 3.2 Courbe en L de Pareto représentant les erreurs des sources/modèle et les erreurs des sources/mesures pour différent λ compris entre 0 et 2. La valeur de λ optimale est celle qui est la plus proche de l'origine.

3.4 Résolution non linéaire

Le caractère sous-contraint du problème inverse donne une faible résolution spatiale aux méthodes linéaires. Ces techniques tendent à produire un effet de flou qui éparpille l'activité des sources. Elles sont donc inadaptées à une évaluation précise de l'étendue des sources. Pour s'affranchir de cette limitation, il faut avoir recours à des méthodes non linéaires généralement beaucoup plus coûteuses en temps de calcul. Ces techniques s'affranchissent de

l'hypothèse gaussienne formulée pour les méthodes linéaires; la fonction $f(S)$ devient une extension de la solution.

3.4.1 Minimisation de la norme L_p

Une grande partie des travaux menés sur les méthodes non linéaires portait sur des métriques à base d'entropie et sur les normes L_p pour $p < 2$. Les deux méthodes expliquées par la suite sont développées sur la base des travaux de D. Rao (1999). Les méthodes non linéaires sont des méthodes itératives, où on modifie la métrique des sources C_s^{-1} tant que les solutions ne sont pas stables.

La fonction de coût $L_p(S)$ à minimiser est

$$L_p(S) = \|M - GS\|_F^2 + \lambda E^{(p)}(S) \quad (3.11)$$

Une mesure de "diversité" populaire est la fonction de coût suivante :

$$E^{(p)}(S) = \text{sign}(p) \sum_{i=1}^n |S[i]|^p, \quad p \leq 2 \quad (3.12)$$

Cette norme fait intervenir un paramètre p qu'on choisie entre 0 et 2. Plus p est grand, plus la solution est spatialement entendue et énergétique. Pour p proche de 0 les solutions deviennent concentrées (ou "sparse") et focales.

Pour $p = 0$ les sources "allumées" prennent la valeur 1, ainsi on peut connaître le nombre de sources actives en faisant la somme des intensités des sources estimées.

D'après la publication de D. Rao pour que les sources soient stables nous devons avoir le couple S^* et λ^* qui sont des points stationnaires des fonctions de Lagrange suivantes :

$$\begin{aligned}\nabla_S L(S) &= \nabla_S E^{(p)}(S^*) + G^t \lambda^* = 0 \\ \nabla_S L(S) &= GS^* - M = 0\end{aligned}\tag{3.13}$$

Le gradient de la mesure de diversité, prenant en compte chaque élément i , noté $\nabla_{S[i]}$, peut aussi s'écrire sous la forme

$$\nabla_{S[i]} E^{(p)}(S) = |p| |S[i]|^{p-2} S[i]\tag{3.14}$$

En substituant dans (3.13) il n'apparaît pas de solution simple et évidente, on nous propose alors d'utiliser une représentation factorisée

$$\begin{aligned}\nabla_{S[i]} E^{(p)}(S) &= \alpha(S) C_S(S) S \\ \text{avec } \alpha(S) &= |p| \text{ et } C_S(S) = \text{diag}(|S[i]|^{p-2})\end{aligned}\tag{3.15}$$

A partir de (3.13) et de (3.15), on obtient les points stationnaires qui satisfont

$$\begin{aligned}\alpha(S^*) C_S(S^*) S^* + G^t \lambda^* &= 0 \\ \text{et } GS^* - M &= 0\end{aligned}\tag{3.16}$$

Notons que l'inverse de la métrique des sources $C_S^{-1}(S^*) = \text{diag}(|S[i]|^{2-p})$ existe pour toutes les sources, donc à partir de (3.16) nous avons

$$S^* = -\frac{1}{\alpha(S^*)} C_S^{-1}(S^*) G^t \lambda^*\tag{3.17}$$

En remplaçant S^* dans la seconde équation de (3.16), on trouve λ^* égal à

$$\lambda^* = -\alpha(S^*)(GC_S^{-1}(S^*)G^t)^{-1}M \quad (3.18)$$

En remplaçant (3.18) dans (3.17) on en déduit que

$$S^* = C_S^{-1}(S^*)G^t(GC_S^{-1}(S^*)G^t)^{-1}M \quad (3.19)$$

S^* apparaît donc comme une solution de type "point fixe" qu'on obtient itérativement où S_k et C_{S_k} sont obtenus à la $k^{ième}$ itération :

$$S_{k+1} = C_{S_k}^{-1}G^t(GC_{S_k}^{-1}G^t)^{-1}M \quad (3.20)$$

avec

$$C_{S_k}^{-1} = \text{diag}(|S_k(:,t)|^{2-p}) \quad (3.21)$$

Pour chaque instant t , on effectue les itérations de cet algorithme pour obtenir $S^*(t)$. Pour initialiser cet algorithme il faut choisir la matrice S_0 . On choisit de l'initialiser avec la méthode L2 (Tikhonov), c'est-à-dire

$$S_0 = G^t(GG^t)^{-1}M \quad (3.22)$$

L'erreur quadratique sur les sources entre chaque itération k est de plus en plus petite, $\sum_i (S_k(i) - S_{k-1}(i))^2$. De ce fait, la condition d'arrêt de l'algorithme est satisfaite lorsque cette erreur est inférieure à une tolérance définie au préalable.

3.4.2 Entropie de Shannon

Ici la fonction de régularisation est définie par :

$$H(S) = -\sum_{i=1}^n \tilde{S}[i] \ln(\tilde{S}[i]) \quad \text{avec} \quad \tilde{S}[i] = \frac{|S[i]|^2}{\|S\|_2^2} \quad (3.23)$$

L'approche est similaire à la précédente. En remplaçant la factorisation de (3.15) on obtient la métrique suivante

$$\begin{aligned} \nabla_S H(S) &= \alpha(S) C_S(S) S \\ \text{avec :} \\ \alpha(S) &= \frac{2}{\|S\|_2^2} \\ C_S(S) &= -\text{diag}(H(S) + \ln(\tilde{S}[i])) \end{aligned} \quad (3.24)$$

permettant de calculer les sources selon l'équation :

$$S^* = C_s^{-1}(S^*) G' (G C_s^{-1}(S^*) G')^{-1} M \quad (3.25)$$

Ainsi la configuration de source S^* est fonction d'elle même $S^* = g(S^*)$ et on résout cette équation par une méthode dite point fixe.

3.4.3 Convergence de l'entropie de Shannon

Pour arriver à une stabilité nous devons calculer la nouvelle métrique et la nouvelle configuration de source à chaque itération. Pendant la phase de test, il s'est avéré que cette méthode, ne converge pas forcément avec des données réelles et qu'elle n'est pas robuste. Il faut procéder à une remise en question de l'étude par minimisation de l'entropie de Shannon en ajoutant le critère de régularisation tel que vu dans la partie linéaire de cet exposé. On

choisit un λ de valeur faible pour régulariser le plus possible en respectant la valeur des sources calculées. On peut aussi voir que le fait d'ajouter un λ petit multiplié par une matrice diagonale revient en fait à ajouter du bruit pour permettre l'inversion du terme $GC_S G'$ plus facilement dans l'équation suivante.

$$S^* = C_S^{-1} G' (GC_S G' + \lambda I)^{-1} M \quad (3.26)$$

Rao propose une solution pour forcer la convergence à chaque itération. Elle consiste à minimiser l'entropie de Shannon en utilisant l'algorithme suivant :

$$S_{k+1} = \begin{cases} S_k & \text{si } S_k^t C_S(S_{k-1}) S_k < 0 \text{ ie } H_S(S_k) < H_S(S_{k-1}) \\ S_k + (1 + \mu)(S_{k-1} - S_k) & \text{si } S_k^t C_S(S_{k-1}) S_k > 0 \text{ ie } H_S(S_k) > H_S(S_{k-1}) \end{cases}$$

Selon lui, si S_k existe alors S_{k+1} existe aussi. L'incrément $S_k - S_{k+1}$ est alors décroissant avec une proportion plus ou moins grande en fonction du paramètre μ . Ceci assure que l'entropie est réduite à chaque itération. Nous allons donc nous intéresser à la manière dont on peut réduire l'entropie pour converger plus aisément. Regardons de plus près l'expression ci dessous.

$$S_{k+1}^t C_S(S_k) S_{k+1} \quad (3.27)$$

Si (3.27) est négatif cela veut dire que les sources à l'instant $k+1$, mesurées avec la métrique $C_S(S_k)$, donnent une distance inférieure à celle des sources mesurées aussi avec la même métrique $C_S(S_k)$ à l'instant k . Nous pouvons alors en déduire que $H_S(S_{k+1}) < H_S(S_k)$ et dans ce cas on ne corrigera pas la valeur de S_k car l'algorithme convergera de lui même : $S_k = S_{k+1}$, la minimisation de l'entropie est assurée.

Par contre, en suivant le même raisonnement, si (3.27) est positif, alors les sources à l'instant $k+1$, mesurées avec la métrique $C_S(S_k)$, donnent une distance supérieure à celle des

sources mesurées aussi avec la même métrique $C_S(S_k)$ à l'instant k . $H_S(x_{k+1}) > H_S(x_k)$ entraîne la divergence de l'algorithme général. Dans ce cas, pour y remédier, nous allons ajouter aux sources S_k une valeur dépendant des événements passés $(S_k - S_{k+1})$ modulée par le paramètre μ . Dès lors $S_k = S_{k+1} + (1 + \mu)(S_k - S_{k+1})$ permettra de minimiser l'entropie. L'algorithme s'arrête quand l'entropie devient inférieure à une valeur définie au préalable.

3.5 Conclusion sur le chapitre 3

L'approche de la minimisation de la norme L_p et celle de la minimisation de l'entropie de Shannon ont été implémentées dans le logiciel BrainEntropy de manière à visualiser les sources estimées sur la surface du cortex. Dans le chapitre suivant, on comparera ces méthodes avec d'autres afin de quantifier leur efficacité. Pour ce faire, on utilisera un simulateur d'activité cérébrale pour générer des mesures qui seront injectées dans les méthodes de résolution du problème inverse. On pourra alors utiliser un outil (l'analyse ROC), permettant de comparer les sources estimées avec celles simulées dans le but de quantifier la robustesse des méthodes de résolution du problème inverse.

CHAPITRE 4

BRAINENTROPY : UN OUTIL DE SIMULATION ET DE RÉOLUTION DU PROBLÈME INVERSE EN IMAGERIE EEG - MEG

Développé dans le cadre de cette maîtrise, BrainEntropy est un logiciel qui permet de localiser l'activité cérébrale mesurée en EEG, en MEG ou les deux simultanément en utilisant les méthodes de résolution du problème inverse expliquées dans le chapitre 3 (l'approche régularisée L_p et l'approche par maximisation de l'entropie de Shannon). Développé avec Matlab, BrainEntropy présente une 'interface Homme-machine' faite pour être utilisée par des personnes étudiant l'activité cérébrale (neurologues, psychologues, médecins, ...) et qui ne se soucient pas ou peu du traitement des données modélisant le processus bioélectromagnétique.

On rappelle que les éléments communs à ce processus dans la résolution du problème inverse et dans la modélisation du problème direct sont : l'espace des sources, l'espace des mesures et la relation entre ces deux espaces intégrée dans la matrice de gain. De plus, on rappelle que nous avons fait l'hypothèse que les sources sont représentées par un modèle distribué de milliers de dipôles électriques orientés perpendiculairement à la surface corticale (matière grise). BrainEntropy regroupe ainsi ces éléments dans un 'modèle' de données structuré selon une approche orientée objet. Ce 'modèle' de données, créé par BrainEntropy, peut être archivé et lu pour faciliter la manipulation des éléments modélisant le processus bioélectromagnétique et les mesures.

Dans le cas où l'on ne dispose pas de matrice de gain, BrainEntropy intègre OpenMEEG pour en calculer une selon un modèle réaliste de la tête. La condition d'utilisation sera alors de fournir : le modèle distribué des sources, les maillages délimitant les tissus biologiques et la position des capteurs. Une originalité dans la préparation des données, pour le calcul de la matrice de gain EEG, est la possibilité de recalibrer toutes les électrodes à une certaine distance de la tête. Cet outil sert à corriger le recalage initial des électrodes dans le repère IRM qui parfois n'est pas réaliste.

BrainEntropy intègre aussi un simulateur permettant d'activer une région de sources actives et de générer des signaux sur les capteurs. Nous verrons que cet outil est utile pour valider une matrice de gain et pour quantifier la robustesse des méthodes de résolution du problème inverse.

En résumé ce chapitre explique :

- comment préparer les données en utilisant la position des électrodes EEG la plus réaliste pour créer une matrice de gain optimale;
- comment créer un 'modèle' pour BrainEntropy;
- comment sont organisées les données dans un modèle de BrainEntropy;
- comment le simulateur permet de valider une matrice de gain;
- comment quantifier la robustesse des méthodes de résolution du problème inverses avec une analyse ROC (Simulation);
- la résolution du problème inverse à partir de vraies données.

Cet environnement de calcul structuré constitue la principale contribution de ce travail.

4.1 Préparation des données : Maillages et recalage des électrodes

Dans la plupart des cas, quand l'on souhaite localiser l'activité cérébrale, on ne dispose que de la position des capteurs recalés dans le volume IRM, de l'image IRM, et des signaux acquis par les capteurs EEG/MEG. Il manque la matrice de gain. BrainEntropy contient un outil de préparation des données qui intègre les applications d'OpenMEEG pour calculer les matrices de gain EEG et MEG à partir de l'emplacement des capteurs (EEG ou MEG) et des surfaces délimitant les tissus biologiques obtenus par BrainSuite2 (voir chapitre2).

Dans un premier temps, cet outil permet de visualiser et de vérifier si le maillage des sources et ceux délimitant les tissus biologiques sont bien encastrés dans le même repère volumique. En effet, il se peut que la segmentation des sources soit déjà faite par le centre d'acquisition EEG/MEG et que le maillage de la surface corticale ne soit pas orienté dans le même repère volumique que l'image IRM fournie. Dans un deuxième temps, cet outil de préparation des

données de BrainEntropy pour le calcul de la matrice de gain, permet de visualiser et vérifier si les électrodes EEG sont bien recalées dans le repère volumique de l'IRM. En effet, on visualise parfois que le recalage des électrodes numérisées avant l'acquisition dans le volume IRM soit biaisé ou non réaliste²⁵. Il faut revoir plus précisément le recalage. Telle est la question qui fait l'objet de la suite de la description de cet outil de préparation des données pour le calcul de la matrice de gain EEG. (Le recalage de la position des capteurs MEG ne fait pas l'objet de ce travail. Cependant, on peut faire confiance aux positions des capteurs MEG recalées dans le volume IRM fournies par le centre d'acquisition.)

Même si OpenMEEG permet de calculer une matrice de gain EEG quelque soit la position des électrodes, il est plus rigoureux de les recalculer sur la surface de la peau ou de les écarter sensiblement de la surface de la tête (2mm). Pour ce faire nous utilisons une méthode, inspirée de [12], dont le principe consiste à calculer les vecteurs reliant le centre de la tête à chaque électrode. On déplacera les électrodes le long de ces vecteurs selon une distance qui les écarte de la surface de la tête vers l'extérieur. Le centre de la tête (P_0) est défini comme le barycentre des nœuds de la surface numérique de la peau.

Pour mesurer la distance entre la peau et les électrodes, nous devons trouver les points d'intersection entre la surface de la peau et les droites reliant le centre de la tête et les électrodes. Considérons une électrode placée en P_1 . Le problème revient à résoudre le calcul du point d'intersection P_I d'une droite R (passant par P_0 et P_1) traversant un triangle T formé par trois nœuds du maillage de la peau.

Pour connaître le triangle T dont il est question, l'algorithme fait l'hypothèse que les 3 points du maillage de la peau les plus proches de l'électrode P_1 forment le triangle traversé par la droite R . Notons ces 3 points V_0 , V_1 et V_2 .

²⁵ Parfois, on obtient des électrodes placées à l'intérieur de la tête du patient !

Le triangle T se trouve dans le plan π avec le vecteur normal $\vec{n} = \overline{V_0V_1} \times \overline{V_0V_2}$. D'abord on vérifie l'existence du point d'intersection. Si $\vec{n} \cdot \overline{P_0P_1} \neq 0$ il existe une intersection. Puis nous devons déterminer si ce point est dans le triangle T . Dans le cas contraire on définira un autre triangle en prenant le quatrième point du maillage le plus proche de l'électrode à la place du troisième.

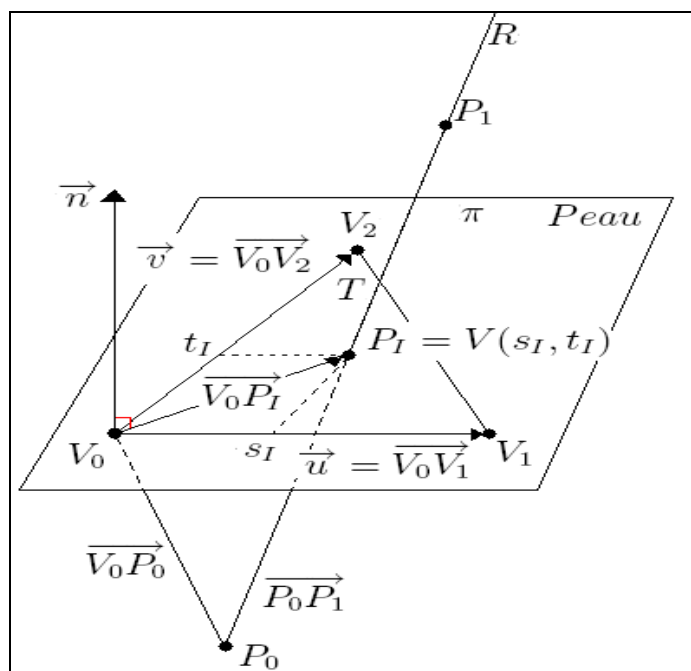


Figure 4.1 Illustration du triangle du maillage de la peau (T) traversé par une droite (R). P_0 est au centre de la tête. Le point d'intersection (P_I) permet d'éloigner une électrode de la peau vers l'extérieur le long de cette droite.

Pour connaître le point d'intersection P_I entre une droite R et un plan π , prenons l'expression paramétrique de la droite R :

$$P(r) = P_0 + r\overline{P_0P_1} \quad (4.1)$$

où r est un nombre réel représentant une distance. Si $\vec{n} \cdot \overline{P_0P_1} \neq 0$ alors R et π se croisent en un unique point P_I . Au point d'intersection, le vecteur $\overline{V_0P_I} = \overline{V_0P_0} + r\overline{P_0P_1}$, est perpendiculaire à \vec{n} . Ceci est équivalent à utiliser l'une des conditions du produit scalaire $\vec{n} \cdot (\overline{V_0P_0} + r\overline{P_0P_1}) = 0$. La solution donne²⁶ :

$$r_I = \frac{-\vec{n} \cdot \overline{V_0P_0}}{\vec{n} \cdot \overline{P_0P_1}} \quad (4.2)$$

Nous présentons maintenant une méthode qui utilise un calcul 3D pour déterminer l'inclusion du point d'intersection P_I dans le triangle T . Comme [Moller-Trumbore, 1997], nous utilisons l'équation paramétrique de π relative à T , mais avec une solution dérivée qui calcule les coordonnées paramétriques du point d'intersection dans le plan. L'équation paramétrique du plan est donnée par :

$$V(s, t) = V_0 + s\vec{u} + t\vec{v} \quad (4.3)$$

où s et t sont des nombres réels, et $\vec{u} = \overline{V_0V_1}$ et $\vec{v} = \overline{V_0V_2}$. Un point $P = V(s, t)$ est dans le triangle T si $s \geq 0$, $t \geq 0$, et $s + t \leq 1$. Donc, pour un point P_I donné nous devons d'abord calculer ses coordonnées (s_I, t_I) , et ensuite vérifier les inégalités pour vérifier son inclusion dans T . De plus, un point $P = V(s, t)$ est placé sur une arête de T s'il vérifie une de ces conditions : $s = 0$, $t = 0$, $s + t = 1$ (Chaque condition correspond à une arête).

Les coordonnées des points du triangle T correspondent aussi à $V_0 = V(0, 0)$, $V_1 = V(1, 0)$, $V_2 = V(0, 1)$.

²⁶ Un segment sur la droite R coupe π seulement si $0 \leq r_I \leq 1$, permettant de savoir si l'électrode est en dehors ou à l'intérieur de la tête.

Dans le but de connaître s_I et t_I pour déterminer l'inclusion, on considère un point P_I obtenu avec (4.2) et (4.1) formant un vecteur $\overline{V_0P_I}$ avec V_0 et on résout l'équation paramétrique du plan (4.3) de $\overline{V_0P_I}$ pour s et t : $\overline{V_0P_I} = s\vec{u} + t\vec{v}$. Il est possible d'annuler les composantes du plan \vec{u} et \vec{v} en leur appliquant un produit scalaire par un vecteur orthogonal à ces composantes et à la normale du plan. On a les relations suivantes :

$$\begin{aligned}\overline{V_0P_I} \cdot (\vec{n} \times \vec{v}) &= s\vec{u} \cdot (\vec{n} \times \vec{v}) + t\vec{v} \cdot (\vec{n} \times \vec{v}) = s\vec{u} \cdot (\vec{n} \times \vec{v}) \\ \overline{V_0P_I} \cdot (\vec{n} \times \vec{u}) &= s\vec{u} \cdot (\vec{n} \times \vec{u}) + t\vec{v} \cdot (\vec{n} \times \vec{u}) = s\vec{v} \cdot (\vec{n} \times \vec{u})\end{aligned}$$

Ainsi s_I et t_I sont données par :

$$s_I = \frac{\overline{V_0P_I} \cdot (\vec{n} \times \vec{v})}{\vec{u} \cdot (\vec{n} \times \vec{v})} \text{ et } t_I = \frac{\overline{V_0P_I} \cdot (\vec{n} \times \vec{u})}{\vec{v} \cdot (\vec{n} \times \vec{u})} \quad (4.4)$$

Le point d'intersection permet ainsi de placer l'électrode à la surface du scalp. Dans le cas où l'on souhaite "décoller" l'électrode, on effectuera un déplacement le long de la droite R :

$$P_{elec} = P_I + \alpha \overline{P_0P_I} \quad (4.5)$$

où α représente la distance en millimètres entre P_I et P_{elec} puisque nous travaillons dans le repère IRM millimétrique.

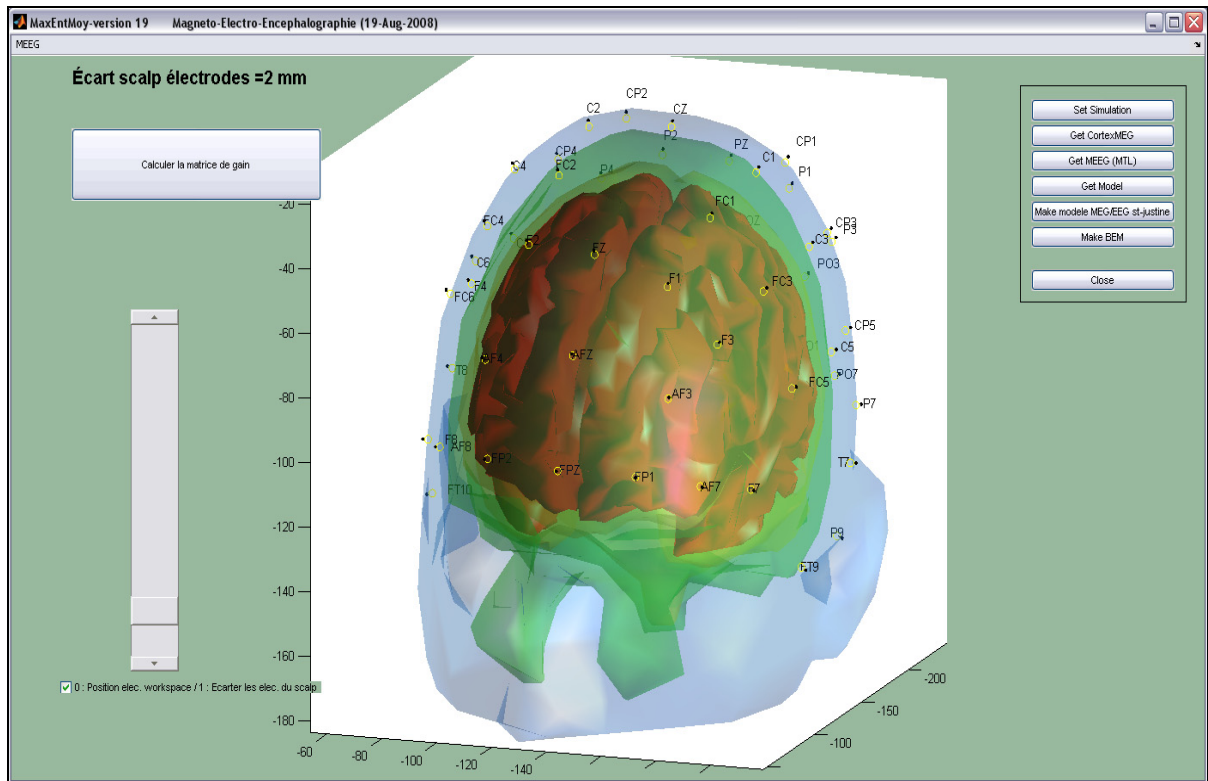


Figure 4.2 Préparation des données pour créer une matrice de gain BEM avec BrainEntropy. Ici on visualise toutes les électrodes distantes de 2mm de la surface de la tête. Le calcul de la matrice de gain s'exécute en cliquant sur le bouton "calculer la matrice de gain".

4.2 Création d'un Modèle

Un modèle de BrainEntropy est une structure de données particulière (cette structure sera détaillée dans la prochaine section) qui est créée à partir des éléments montrés par la figure 4.3 tels que : le répertoire de travail d'une acquisition EEG/MEG d'un sujet, l'espace des sources du sujet²⁷, la ou les matrice(s) de gain EEG ou MEG selon le type d'acquisition, les informations des capteurs, les signaux mesurés pendant l'acquisition et le maillage de la peau du sujet. Le contenu des fichiers pris en entrée par BrainEntropy pour la création d'un modèle est détaillé figure 2.7.

²⁷ qui est représenté soit par un maillage du cortex, soit par les deux maillages des deux hémisphères

La création et la sauvegarde d'un modèle est importante afin d'éviter de charger tous les éléments à chaque fois que l'on souhaite examiner l'activité des sources en fonction des mesures. De plus, l'organisation des données dans le modèle permet à BrainEntropy de connaître la nature des mesures et d'afficher les options adéquates.

Tous les éléments constituant un modèle ne sont pas obligatoires. Par exemple, dans le cadre d'une simulation, il n'est pas nécessaire de créer un modèle contenant des mesures puisqu'elles seront générées par le simulateur de BrainEntropy. Dans ce cas, on laissera vide le champs "fichier Data" de la fenêtre illustrée figure 4.3.

Le champs "HeadMesh", qui correspond au maillage de la peau, n'est pas obligatoire. Dans le cas où il serait présent, BrainEntropy affichera sur la peau numérisée du sujet la projection des champs électrique ou magnétique mesurés (ou simulés) sur les capteurs. Dans le cas contraire, BrainEntropy affichera les champs électromagnétiques sur une carte de couleur 2D représentant l'interpolation de l'intensité de la mesure de chaque capteur en vue de dessus.



Figure 4.3 Interface IHM pour la création d'un modèle BrainEntropy. Le modèle pourra ensuite être chargé directement et il contient au moins les informations nécessaires à une simulation.

4.3 Description des classes et des variables principales

Cette section présente l'architecture du programme. Il est construit selon une approche orientée objet dans l'environnement de MATLAB (voir [10] pour plus de détails). On appelle

un objet une variable qui appartient à une classe et qui utilise ses propriétés. Ainsi plusieurs objets peuvent être l'instance d'une même classe. Une classe représente un concept, comme par exemple dans notre cas : un modèle, une modalité, des données, un capteur, une surface corticale ou un dipôle. Avec MATLAB, une classe est un sous-répertoire du répertoire courant portant le nom de la classe et précédé du symbole @. Exemple : @maclasse. Ce répertoire contient au moins les cinq fichiers suivants qui définissent la classe : char.m, display.m, get.m, set.m, constructeur.m.

- display.m et char.m sont liés car en fait le premier appelle ce que contient le second. C'est donc dans ces deux fichiers qu'il est défini ce qui apparaît à l'écran lorsque le programmeur demande à imprimer ce que contient l'objet qu'il manipule.
- get.m et set.m sont des fichiers d'interfaces avec les instances d'une classe permettant d'obtenir ou de modifier les attributs d'un objet.
- constructeur.m est le constructeur qui permet, comme son nom l'indique, de construire les objets et d'initialiser tous les attributs de ces objets. Les constructeurs de BrainEntropy sont rapportés en annexe C.

La structure générale de BrainEntropy est particulière. En effet, la plupart des fonctions utilisent la variable 'console' attachée à l'interface du logiciel qui contient dans son 'UserData' les variables et les objets d'un modèle. La figure 4.4 nous donne l'aspect général du squelette de données de BrainEntropy. Les flèches de la figure 4.4 représentent un lien parent-enfant entre deux classes. Par exemple, un dipôle est un enfant de la surface corticale. Dans le schéma 4.4 nous montrons la classe modèle. Cette classe est en fait le modèle de données regroupant les informations de toute la modélisation. Elle est mise à jour au chargement d'un fichier 'modèle' de BrainEntropy.

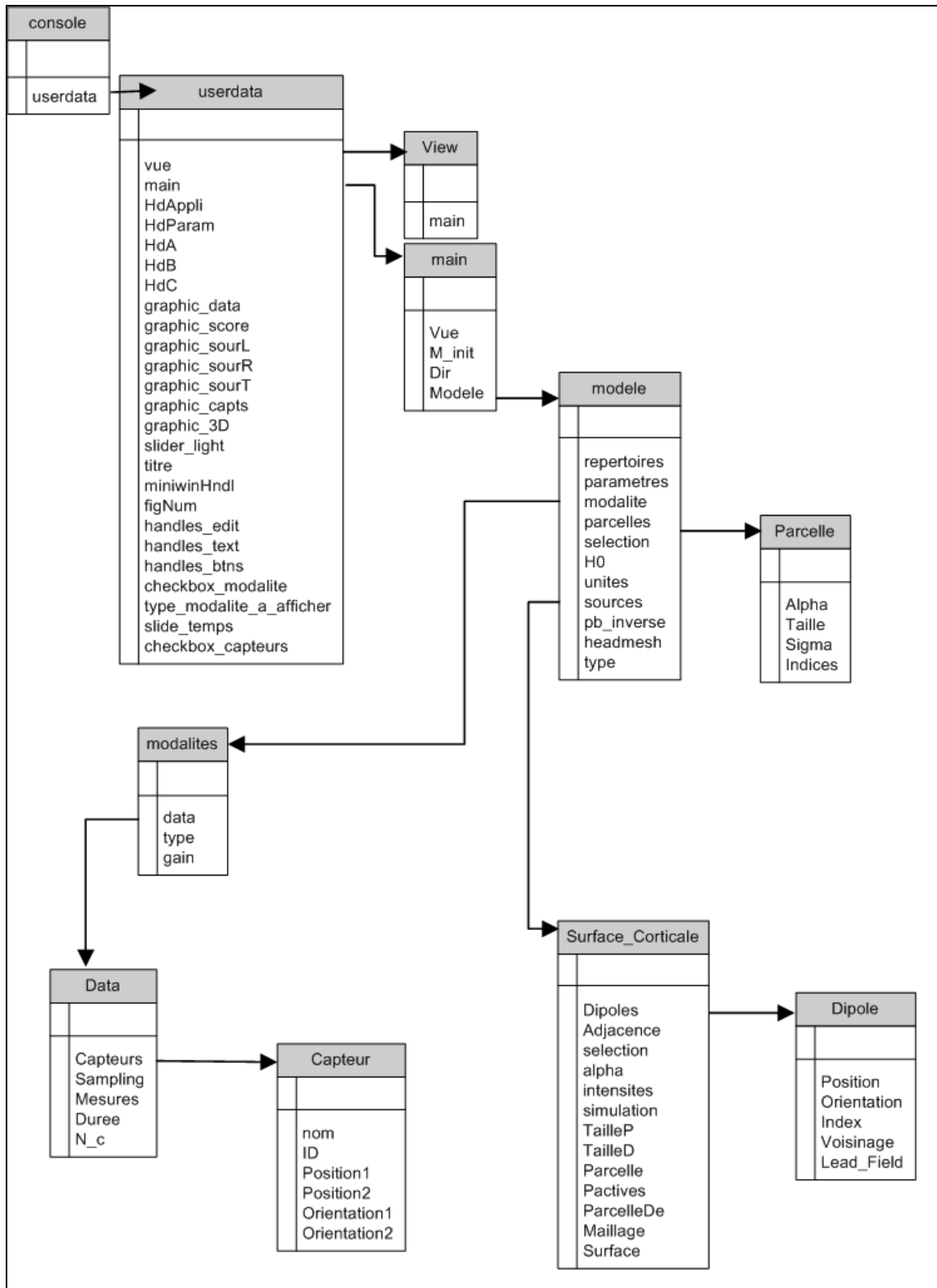


Figure 4.4 Organigramme de la structure des classes de BrainEntropy.

4.4 Simulation d'activité cérébrale et validation des matrices de gain

Le simulateur d'activité permet de valider empiriquement le problème direct. On définit aléatoirement une ou plusieurs régions actives dans le modèle distribué et on génère des mesures sur les capteurs MEG/EEG grâce aux matrices de gain. Trois paramètres sont définis par l'utilisateur :

– Le nombre et la taille de régions actives :

Pour contrôler l'étendue d'une région active, on utilise une particularité de la matrice d'adjacence du maillage des sources. La taille de cette matrice d'adjacence est le nombre de sources (ligne) par nombre de sources (colonne) et elle indique les nœuds du maillage des sources qui sont géométriquement directement voisins. Ainsi, cette matrice est symétrique avec des 1 sur sa diagonale, et binaire avec des 1 là où les dipôles sont voisins adjacents et des 0 dans d'autres cas. La particularité de cette matrice, est que lorsqu'on la met à la puissance n elle nous indique le nombre de nœuds, jusqu'à n , du chemin le plus court dans le maillage pour rejoindre deux nœuds. Ainsi on peut étendre à sa guise une région autour d'un germe en spécifiant la puissance de la matrice d'adjacence et en inspectant les coefficients de la ligne (ou de la colonne) de ce germe qui ne sont pas à 0.

– Le rapport signal sur bruit :

Du bruit gaussien est ajouté sur les intensités des sources en fonction de ce rapport. L'utilisateur saisit un chiffre en décibel pour définir le RSB puis le programme fait une conversion $rsb = 10^{rsb - Db}$ pour passer en linéaire dans la fonction addElecNoise.

Le simulateur va être utile dans cette étude car il permet de valider des matrices de gain MEG et EEG. La façon habituelle pour valider une matrice de gain est de visualiser la projection des signaux sur la peau du sujet. Cette projection représente la configuration des champs (MEG) ou des potentiels (EEG) produits par les sources actives en utilisant la matrice de gain $M = GS$.

Lorsque des dipôles de courant d'une même région sont actifs, ils créent un champs magnétique orienté perpendiculairement aux courants qui traversent ses dipôles. Ce champs magnétique est identifié par un pôle négatif sortant de la tête du patient (représenté en bleu dans la figure 4.6) et par un pôle positif entrant (rouge).

Les dipôles de courant génèrent de plus un potentiel électrique orienté parallèlement au sens du courant neuronal. À la surface de la peau on détecte alors un pôle électrique négatif à l'arrière de la source et un pôle positif en avant de la source comme le montre la figure 4.7. On trouve dans la littérature, l'explication de l'origine des champs projetés sur le scalp comme par exemple dans [9].

La simulation de régions actives à plusieurs endroits du cerveau (voir figure 4.6), permet de vérifier les dernières propriétés électromagnétiques modélisées dans les matrices de gain EEG et MEG pour les valider.

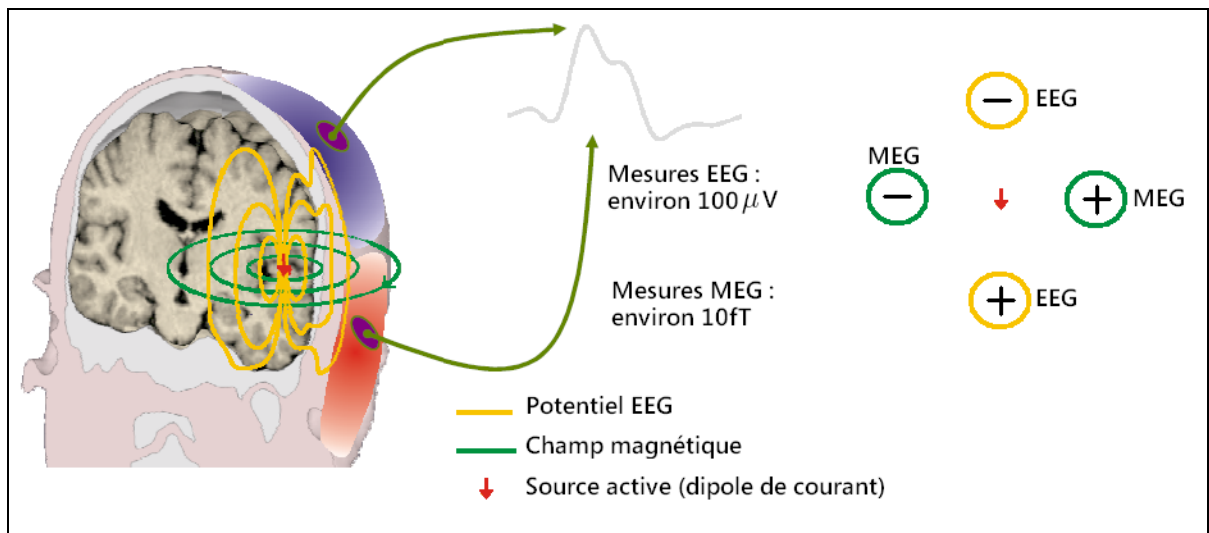


Figure 4.5 Illustration des potentiels électriques et des champs magnétiques gravitant autour d'un dipôle de courant. Tiré de : http://lsiit-miv.u-strasbg.fr/contenu/fichiers_page/MEG-EEG-online.pdf

La meilleure façon de valider les matrices de gain serait de résoudre le problème inverse sur des données réelles connaissant les stimuli. Par exemple, en utilisant des données acquises au cours d'une expérience somatotopique²⁸ ou rétinotopique²⁹.

4.5 Exemples de visualisation

En fonction de l'orientation des courants qui traversent les cellules pyramidales dans la matière grise, les topographies des champs électromagnétiques à la surface de la peau sont différentes. Spécialement en MEG, où l'amplitude du champs magnétique mesuré est très sensible à l'orientation des courants neuronaux dans les cellules pyramidales.

- Si les sources sont actives dans un sillon (sources tangentielles), le champs magnétique sera bien mesuré par les capteurs MEG. Toujours dans le cas d'une source tangentielle, on observera les pôles électriques juxtaposés, sur la surface de la peau, entourant la source.
- Si les sources sont actives sur la surface du cortex (sources radiales), le champs magnétique n'aura pas le bon angle pour être détecté par les squids. En ce qui concerne le potentiel électrique, on le détectera à la surface de la peau par un pôle négatif à l'endroit de la source et un pôle positif de l'autre côté de la tête.

Ce potentiel négatif, pour une source radiale, est en général assez diffus et mesuré par plusieurs capteurs autour de la région active. La figure 4.7 montre comment la matrice de gain EEG d'OpenMEEG modélise bien la réalité comparativement à celle de BrainVISA qui est mal polarisée et trop focale.

²⁸ Stimulation sensorielle sur les doigts

²⁹ Stimulation visuelle

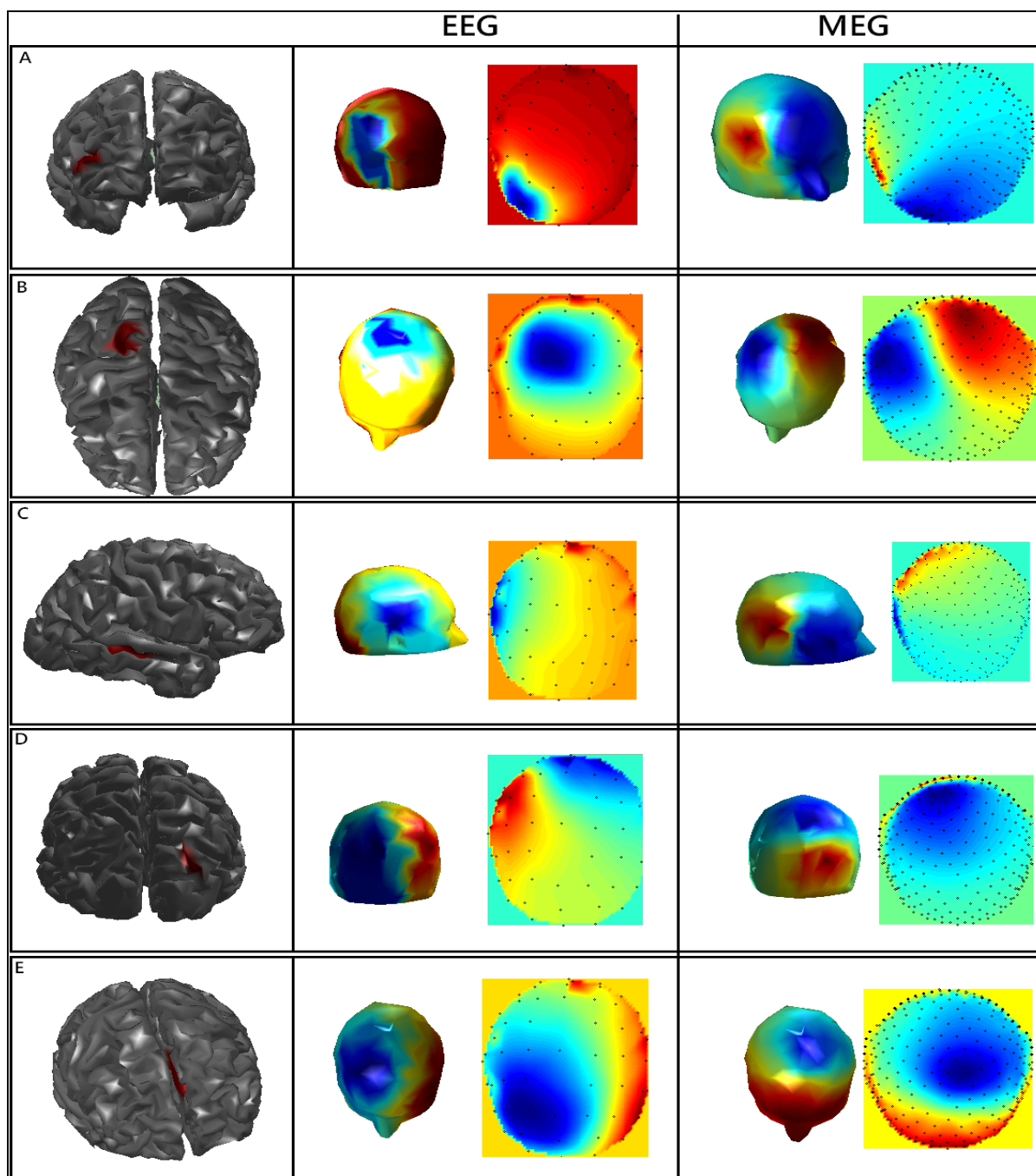


Figure 4.6 Simulations du problème direct. Activation des sources représentées en rouge dans les régions frontale (A), pariétale (B), temporale (C), occipitale (D) et méiale interhémisphérique (E). Les mesures sur les capteurs en EEG et en MEG sont dans la deuxième et dans la troisième colonne respectivement. On montre les mesures projetées sur le scalp ainsi que leurs interpolations sur les électrodes.

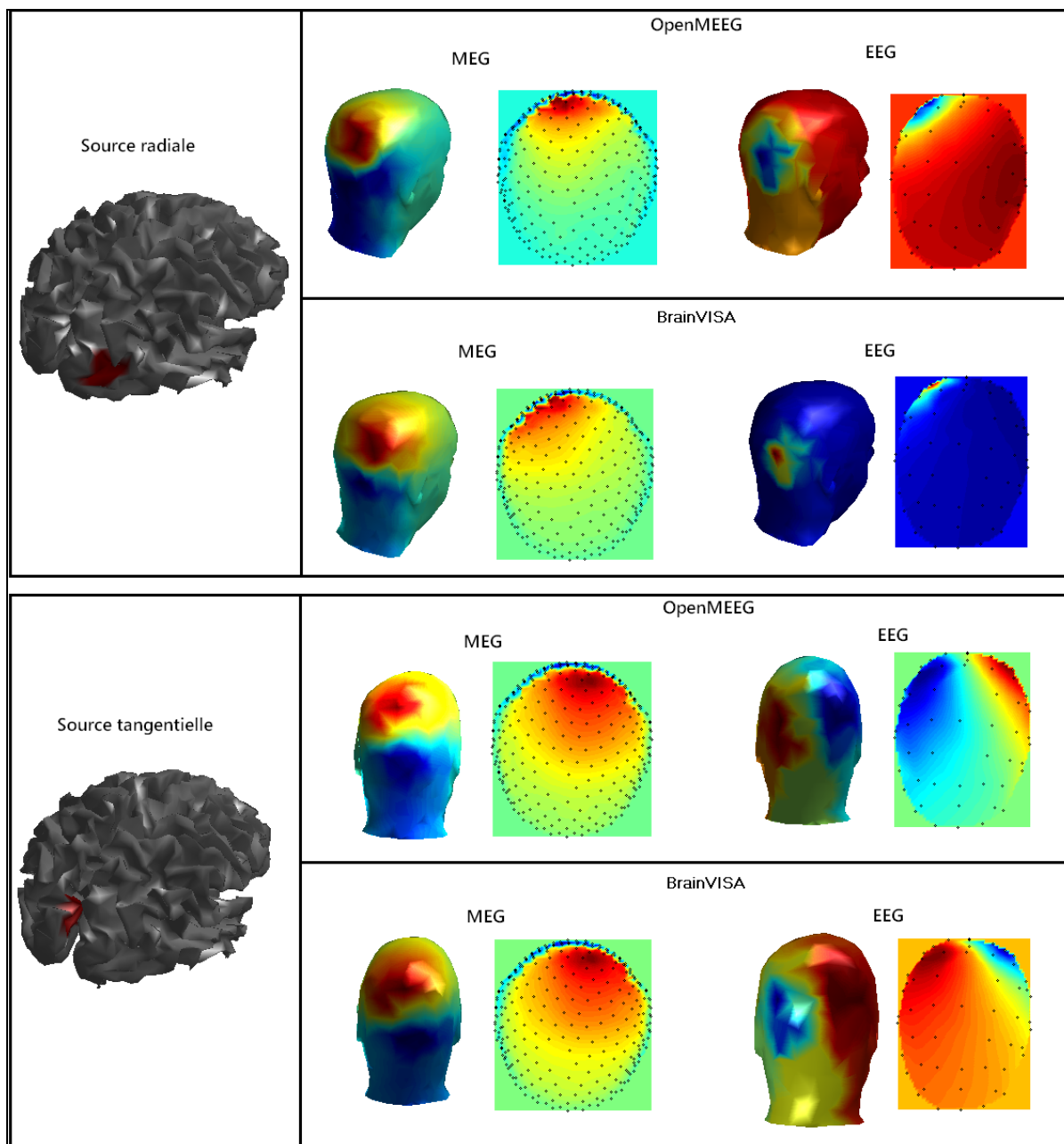


Figure 4.7 Projection des champs magnétique et électrique sur scalp afin de vérifier la validité des matrices de gain générées par OpenMEEG et BrainVISA dans le cas de sources radiales et tangentielles actives.

Il existe une légère différence entre les matrices de gain MEG calculées avec un modèle réaliste ou sphérique. Étant donné que le champ magnétique n'est que peu ou pas atténué par les tissus de la tête, les différences notables entre les matrices de gain MEG calculées par OpenMEEG et BrainStorm proviennent des faits que :

- OpenMEEG prend en compte l'anatomie de la tête contrairement à BrainStorm.
- OpenMEEG prend en compte les champs magnétiques provenant des courants secondaires.

4.6 Est-il judicieux d'utiliser le problème inverse à partir de données simulées ?

On pourrait penser qu'il est possible de valider les matrices de gain en suivant les étapes suivantes :

- activation d'une région de sources actives avec le simulateur produisant des données sur les capteurs;
- résolution du problème inverse, peu importe la méthode, pour avoir une nouvelle cartographie des sources estimées;
- comparaison des sources estimées avec les sources simulées au départ.

Lors de la comparaison, il est légitime de penser que, si les sources estimées sont dans la région générée, alors la matrice de gain est viable. Or c'est une erreur puisque cette approche utilise une matrice de gain identique pour faire un aller-retour. En utilisant la même matrice de gain, il est normal de retrouver les sources du départ en fonction de la robustesse de la méthode du problème inverse utilisée.

Ainsi en comparant une cartographie de cerveau avec des régions simulées et une autre provenant du problème inverse, nous ne pouvons pas valider une matrice de gain. Néanmoins, en utilisant une technique (analyse ROC) permettant de quantifier les différences entre les deux cartographies, nous pouvons valider l'efficacité d'une méthode de localisation de sources.

4.7 Analyse ROC

L'analyse ROC est très largement employée pour valider des algorithmes sur des images médicales. Elle bénéficie d'un consensus général, et elle est considérée comme la méthode de référence dans la communauté radiologique. Dans le cadre du simulateur, celle-ci s'applique très bien comme nous allons le voir dans cette section.

La méthode ROC permet de mesurer la performance d'un imageur à détecter les objets d'une image (maladies, tumeurs, ruban cortical, etc...). Elle confronte la réalité face aux images issues d'une modalité. Il faudra l'expertise d'un médecin (expert) qui nous indiquera quels éléments font partie de l'objet recherché ou non. Le simulateur de BrainEntropy active une région de la surface corticale, ce qui permet de nous affranchir d'un expert. Ainsi en comparant la simulation et les sources estimées, l'analyse ROC détermine la sensibilité et la spécificité des méthodes de localisation de sources selon les définitions du tableau 4.8. La sensibilité correspond au fait que les anomalies présentes dans l'image soient bien détectées lorsqu'elles sont effectivement présentes. La spécificité correspond au fait que des objets ne soient pas détectés s'ils sont effectivement absents. Il est clair qu'en réglant une sensibilité trop forte on risque de confondre du bruit avec des objets réels, et qu'en réglant une spécificité trop forte, on risque de ne pas détecter des objets effectivement présents.

		CONDITION RÉELLE	
		Objet Présent (Positif)	Objet Absent (Négatif)
CRITÈRE D'OBSERVATION	Objet Observé (Positif)	Vrai positif (N_p)	Faux positif (N_{fp})
	Objet Non Observé (Négatif)	Faux négatif (N_{fn})	vrai négatif (N_n)

Figure 4.8 Tableau définissant les paramètres d'une analyse ROC qui permet d'évaluer la performance d'un algorithme de détection.

Les vrais positifs N_p sont les objets que l'on a obtenus positifs et qui sont réellement positifs.

Les faux négatifs N_{fn} sont les objets que l'on a obtenus négatifs mais qui sont réellement positifs.

Les faux positifs N_{fp} sont les objets que l'on a obtenus positifs mais qui sont réellement négatifs.

Les vrais négatifs N_n sont les objets que l'on a obtenus négatifs et qui sont en réellement négatifs.

La sensibilité et la spécificité sont définies de la manière suivante :

$$\text{Sensitivité} = \frac{N_p}{N_p + N_{fn}} .$$

$$\text{Spécificité} = \frac{N_n}{N_n + N_{fp}} .$$

Ces deux paramètres fluctuent en fonction d'un seuil variant entre la plus faible intensité des sources et la plus grande. Les dipôles observés positifs sont ceux dont l'intensité est supérieure à ce seuil et les dipôles avec une intensité inférieure au seuil sont observés comme négatifs. Si un dipôle positif appartient aussi à la zone d'activation simulée, il sera un vrai

positif, sinon un faux positif. Si un dipôle négatif appartient à la région active simulée, il sera considéré comme un faux négatif, sinon comme un vrai négatif. Donc, pour chaque valeur du seuil, il existe un couple sensibilité-spécificité que l'on peut afficher selon les axes suivants : L'axe des abscisses représente $1 -$ la spécificité, et l'axe des ordonnées est la sensibilité. Cette représentation permet de tracer la courbe ROC en reliant tous les points. Plus la courbe est proche du coin supérieur gauche, plus le système est robuste et efficace avec une bonne sensibilité et spécificité. Nous allons donc l'appliquer sur plusieurs méthodes de résolution du problème inverse.

4.8 Comparaison et quantification de l'efficacité de différentes méthodes de résolution du problème inverse

Dans le but de comparer les méthodes expliquées dans le chapitre précédent (Lp et Entropie de Shannon) avec d'autres existantes, la suite de ce travail suggère d'utiliser l'analyse ROC sur les différentes méthodes de résolution du problème inverse afin de pouvoir quantifier leur comportement en EEG MEG et MEEG. Cette étude sera menée sachant qu'il existe une région de sources actives générées par le simulateur de BrainEntropy, comme le montre la figure 4.9.

Nous choisissons d'utiliser comme méthodes comparatives celles que OpenMEEG met à notre disposition dans la résolution du problème inverse :

- MN : (Minimum Norm), c'est la 'technique du pauvre', qui utilise la régularisation de Tikhonov, ou encore la méthode Lp avec $p = 2$, ce qui revient à obtenir une métrique dans l'espace des sources C_S égal à l'identité (voir chapitre 3). Cette méthode est donc très rapide mais donne des sources assez diffuses.

Deux autres méthodes originales de régularisation des sources sont proposées par OpenMEEG. "HEAT" et "Total Variation (TV)". Ces deux méthodes utilisent la norme Lp du gradient de l'amplitude des sources (voir [13]) :

$$f(S) = \|\nabla_{source} S\|_p^2$$

- avec $p=2$ pour la régularisation HEAT.
- avec $p=1$ pour la régularisation TV.

La régularisation TV produit un effet plus focal sur les sources que MN ou HEAT.

Dans la figure 4.10 nous constatons de plus, en EEG, que la meilleure méthode de résolution du problème inverse est la méthode L_p implantée dans BrainEntropy. En MEG la plupart des méthodes donnent des résultats convenables. Le mélange des modalités EEG et MEG dans les résolutions du problème inverse en MEEG donne des résultats satisfaisants pour les méthodes de résolution L_p et HEAT. Toutefois, l'approche HEAT demande un temps de calcul réellement important, tandis que la méthode L_p (le minimum norm MN) s'exécute en quelques minutes.

Pour conclure, on remarque que la méthode L_p se démarque des autres grâce à sa très bonne reconstruction en EEG, son comportement très acceptable en MEG et sa rapidité d'exécution. Dans cette simulation, la valeur de p a été fixée à 1,6. L'optimisation de ce paramètre fût établie en utilisant la courbe en L de Pareto (figure 3.2).

Maintenant que nous avons vérifié la validité des méthodes, nous pouvons les appliquer sur des mesures réelles. Dans la section suivante, nous étudierons donc la résolution du problème inverse à partir de vraies données en utilisant la méthode L_p puisque c'est la plus efficace.

4.9 Application avec des données réelles

La figure 4.9 est une copie d'écran d'une partie de l'interface de BrainEntropy. Ce panneau permet à l'utilisateur de contrôler les paramètres du problème inverse. On peut ainsi choisir si on veut effectuer le problème inverse, seulement à partir des données EEG ou MEG, ou les deux modalités combinées (en cochant les deux cases EEG et MEG). On a aussi la possibilité d'utiliser le calcul de localisation de sources pour un temps donné, ou dans une fenêtre de temps définie par des curseurs 'début' et 'fin'. Enfin BrainEntropy exécutera la méthode de résolution du problème inverse souhaitée dès que l'utilisateur appuiera sur l'un des deux

boutons correspondants. Il est possible de spécifier les paramètres μ et p relatifs aux méthodes de l'Entropie de Shannon et Lp.



Figure 4.9 Contrôle des modalités, de la fenêtre de temps, et de la méthode de résolution du problème inverse.

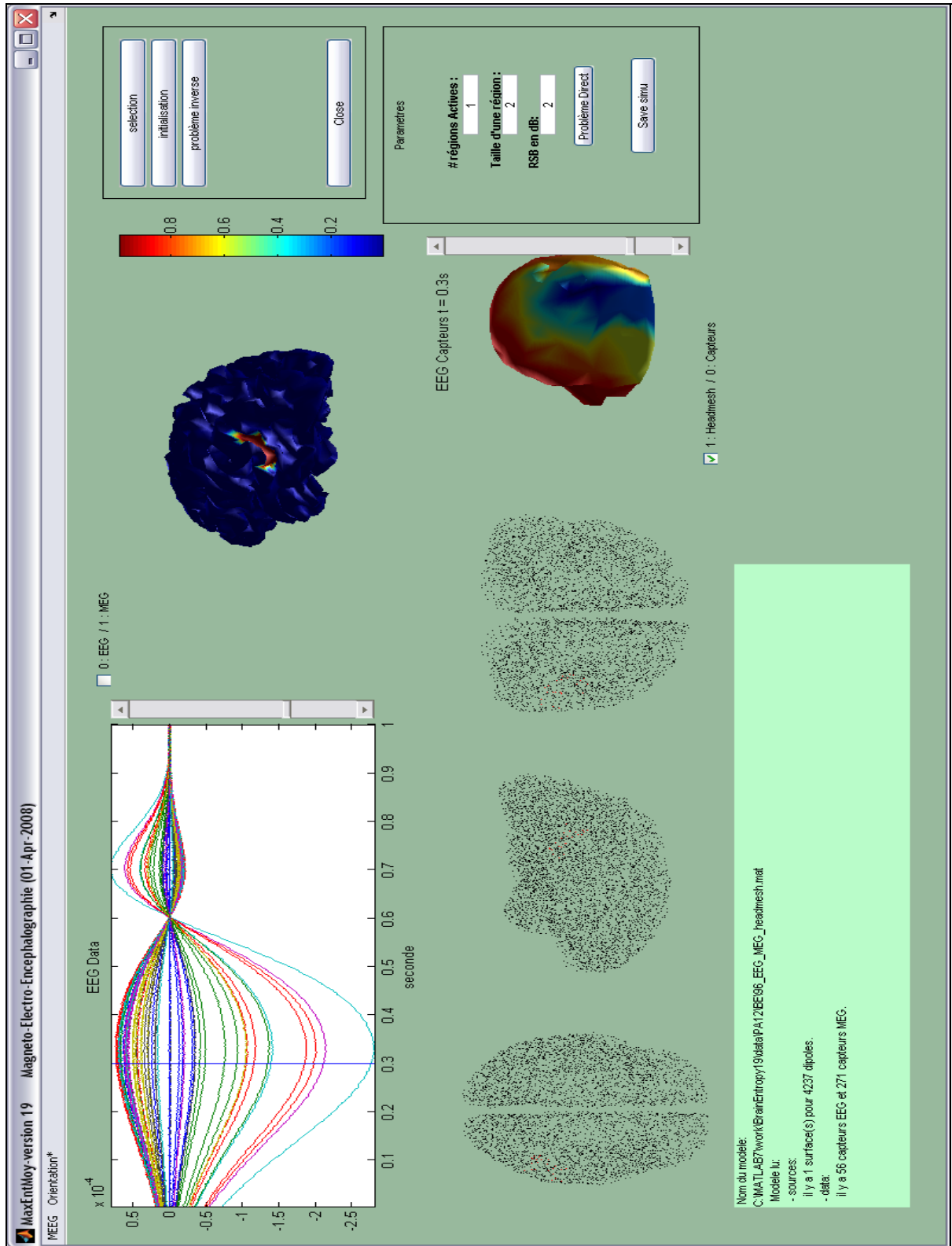


Figure 4.10 Simulateur d'activité bioélectrique dans le logiciel BrainEntropy.

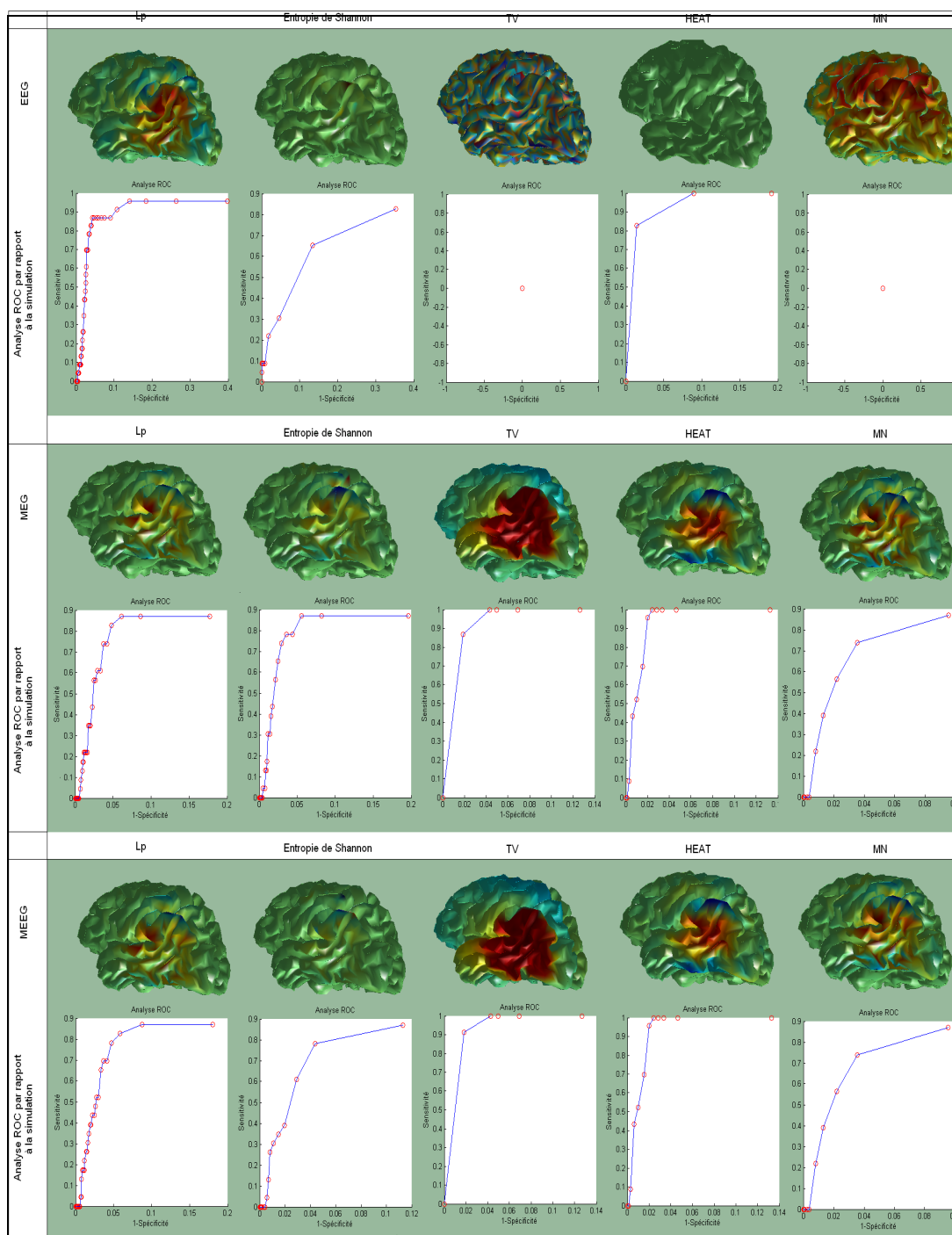


Figure 4.11 Comparaison de méthodes de localisation de sources générées par les mêmes mesures. De gauche à droite, les méthodes sont : Norme L_p , Entropie de Shannon, Total Variation, Heat et « Minimum Norm ». L'aire sous la courbe ROC permet de quantifier la performance d'une méthode à détecter un objet « actif ». Ainsi, la méthode L_p en EEG (première ligne) se démarque des autres méthodes.

Lorsque la méthode sélectionnée a fini de calculer des estimations de sources pour chaque échantillon des mesures, une fenêtre du même type que celle illustrée dans la figure 4.12 apparaît. Avec cette fenêtre on peut localiser et visualiser l'activité cérébrale au cours du temps. Elle montre la carte d'activation des sources sur la surface corticale. Le curseur horizontal permet de sélectionner un échantillon dans les mesures (affichées au bas de la fenêtre) et d'afficher la carte d'activation correspondant à cet instant sur le cortex. Dans le but de filtrer l'intensité des sources on se servira des curseurs verticaux (par exemple pour connaître les sources les plus actives et d'éliminer le bruit). Les sources ayant leur intensité inférieure au seuil du curseur de gauche et celles ayant leur intensité supérieure au seuil de celui de droite sont affichées en gris dans le cortex. Ces curseurs sont des filtres passe haut et passe bas sur l'intensité des sources pour aider l'analyse.

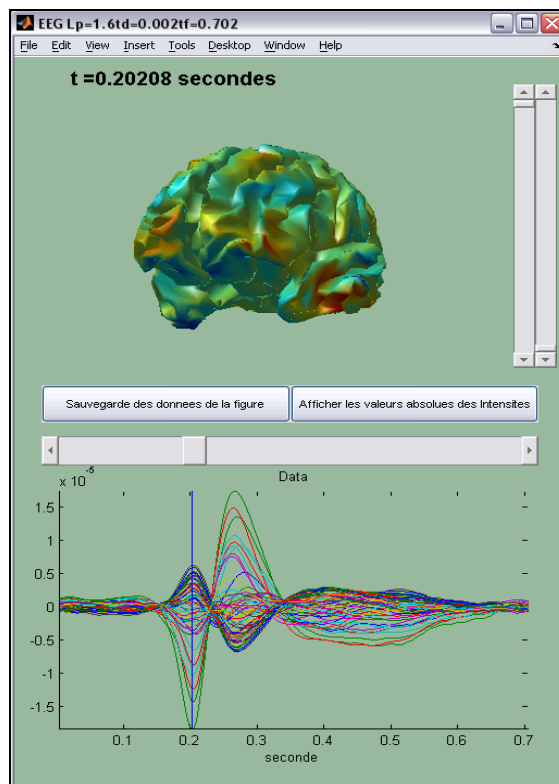


Figure 4.12 Cartographie (en haut) en fonction du temps d'une localisation Lp obtenue à partir d'un signal EEG avec des pointes épileptique moyennées (en bas).

• Exemple : Nous allons utiliser BrainEntropy dans le but de localiser le foyer épileptique d'un patient à partir d'acquisitions faites en EEG et en MEG. On mesure chez les patients atteints de l'épilepsie des pointes d'ondes qui sont syndromatiques de cette maladie. Ces pointes proviennent d'une population de neurones d'une région du cerveau qui décharge spontanément. La notion de décharge épileptique se traduit sur les capteurs EEG et MEG par une oscillation d'environ 300 ms selon les patients. Pour localiser le foyer épileptique, on doit prétraiter les mesures, autant en EEG qu'en MEG. Premièrement, on extrait plusieurs pointes du signal puis on les centre de façon à ce que l'amplitude maximale de la première oscillation de chaque pointe soit située au même instant dans une échelle de temps. Deuxièmement, on effectue une moyenne, sur chaque électrode, de toutes les pointes recentrées dans le temps. Ce prétraitement permet d'augmenter le rapport signal sur bruit dans le but d'obtenir une localisation de sources plus précise. Nous observons en haut des figures 4.13 b), c) et d) le signal épileptique prétraité respectivement en EEG, en MEG et en MEEG. Comme nous le montre ces illustrations, les informations acquises en EEG et en MEG sont distinctes, ce qui justifie de faire une localisation de sources en utilisant ces mesures conjointes.

On choisit de résoudre le problème inverse avec la méthode L_p à partir des mesures prétraitées en EEG, en MEG et en MEEG. Les trois distributions de sources calculées par le problème inverse sont différentes. Toutefois, elles font apparaître une région active commune dans le lobe temporal gauche, qui s'est bien avéré être le foyer épileptique qui fût opéré chez ce patient. De plus, nous avons observé dans la distribution de sources en EEG une action ipsilatérale³⁰ synchrone et inverse du lobe temporal droit quand le foyer épileptique décharge. Pour étudier ce phénomène nous avons isolé en deux groupes les dipôles y participant. Ainsi nous nommerons 'LTD' et 'LTG' les régions dont les intensités s'opposent à l'apparition d'une pointe épileptique. La figure 4.13 a) met en évidence les zones 'LTD' et 'LTG' respectivement dans le lobe temporal droit et gauche. Les deux courbes représentées dans le bas des figures 4.13 b), c) et d) représentent l'activité temporelle de la somme des intensités des dipôles appartenant à une région ('LTD' ou 'LTG'). On montre que cette activité est symétrique et inverse entre les deux zones en EEG, alors qu'elle n'apparaît pas

³⁰ action ayant lieu dans l'hémisphère opposé.

dans les sources estimées en MEG. On montre dans la figure 4.13 d) que la combinaison des deux modalités n'a pas affectée l'activité du foyer épileptique, mais qu'elle a atténuée l'action ipsilatérale comparativement à l'EEG.

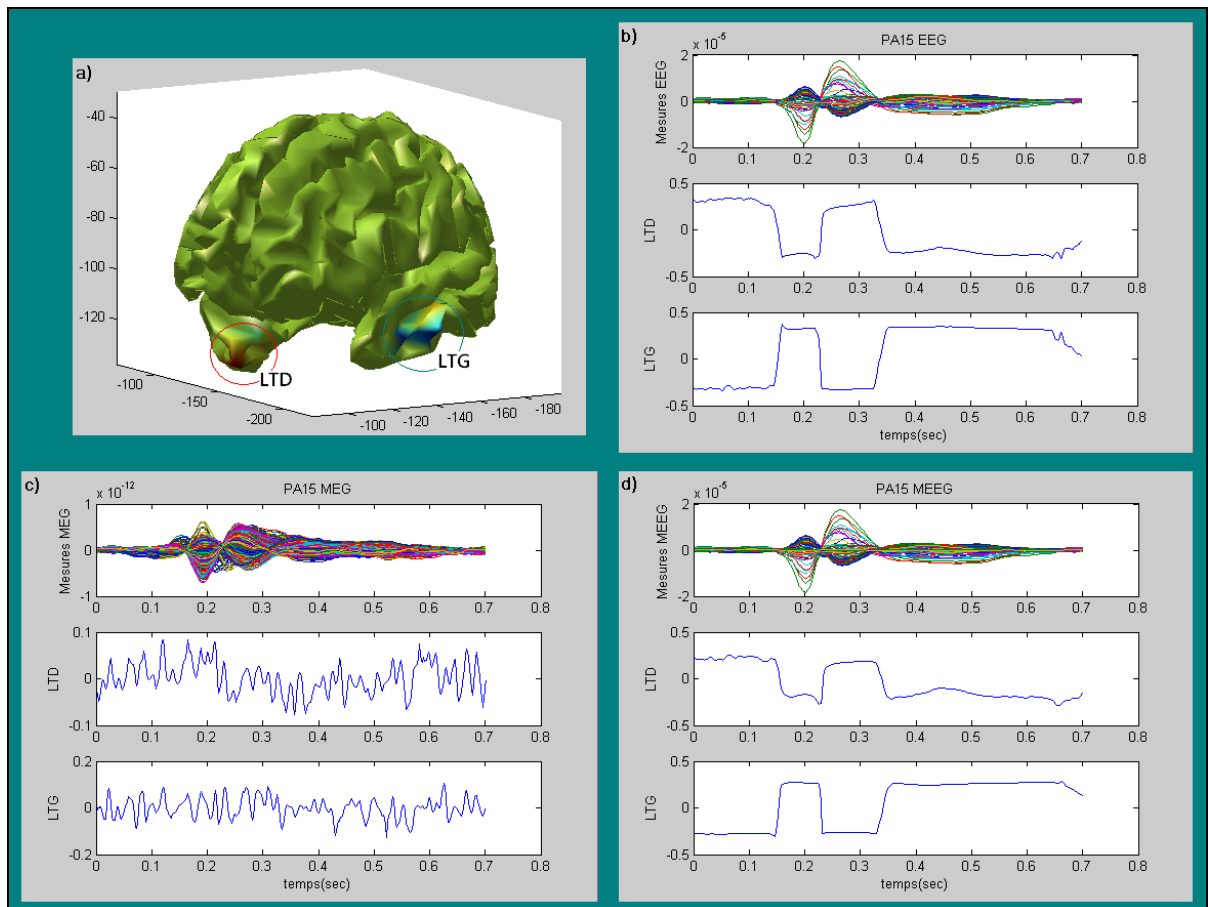


Figure 4.13 Identification d'une réaction bilatérale en EEG mais non présente en MEG lors de l'activation du foyer épileptique situé dans le lobe temporal gauche (LTG). La combinaison EEG et MEG permet d'atténuer le signal de la région non épileptique située dans le lobe temporal droit (LTD).

4.10 Conclusion sur le chapitre 4

BrainEntropy est un environnement informatique développé dans le but de permettre à quiconque travaillant dans les neurosciences d'effectuer des localisations de sources à partir de mesures acquises en MEG et en EEG. L'architecture du programme s'inspire du langage

objet simplifiant l'implantation de nouvelles méthodologies. Ce logiciel intègre plusieurs outils tels que :

- Un outil de préparation des données, utilisé avant d'effectuer le calcul des matrices de gain en EEG et en MEG. La préparation consiste à donner les chemins des fichiers contenant la position des capteurs EEG et/ou MEG, les maillages délimitant les tissus biologiques et le maillage des sources. L'outil de préparation des données affiche alors ces maillages et les capteurs afin de vérifier que tout est bien 'encadré' et donne l'option de recalibrer les électrodes EEG à quelques millimètres de la tête (pour un calcul de gain plus réaliste). À l'appui du bouton "calculer la matrice de gain", en fonction des informations relatives aux capteurs, BrainEntropy exécute OpenMEEG pour obtenir la matrice de gain EEG, MEG ou les deux.
- Un outil de création d'un modèle pour BrainEntropy, illustré dans la figure 4.3, peut alors être utilisé dès lors que l'utilisateur possède le fichier des sources, des capteurs et de la (ou des) matrice(s) de gain.
- D'une part, le modèle créé est alors utilisable par le simulateur de BrainEntropy qui génère des zones actives pour créer des mesures permettant de :
 - Valider les matrices de gain,
 - Quantifier l'efficacité d'une méthode de résolution du problème inverse avec l'analyse ROC.
- D'autre part, le modèle est utilisable par l'outil de résolution du problème inverse de BrainEntropy qui propose deux approches s'appliquant, soit à un instant précis, soit dans une fenêtre de temps des mesures.

CONCLUSION

Dans ce mémoire, nous avons considéré la localisation (spatiale et temporelle) de l'activité cérébrale qui se manifeste dans la matière grise, à partir des informations obtenues en EEG et en MEG. Les logiciels BrainSuite2 et BrainVISA servent à faire un modèle anatomique réaliste de la tête en traitant l'information de l'image IRM. Avec ce modèle réaliste, nous avons ensuite calculé le modèle direct, grâce au logiciel OpenMEEG, qui utilise une méthode BEM pour résoudre la propagation électromagnétique des sources vers les capteurs. Pour manipuler toutes les informations et résoudre le problème inverse, qui consiste à localiser l'activité cérébrale à partir de mesures EEG ou MEG, nous avons présenté le logiciel BrainEntropy qui a été développé dans le cadre de cette maîtrise.

Tout d'abord, ce mémoire a exposé le problème direct en EEG et en MEG. Ce problème consiste à modéliser l'environnement et la propagation de l'activité des sources vers les capteurs. Plutôt que d'utiliser un modèle sphérique et homogène, comme le fait le logiciel BrainVISA, nous avons fait le choix d'utiliser un logiciel récent utilisant l'anatomie réaliste de la tête et qui prend en compte les conductivités électriques des principaux tissus biologiques. Ce choix est important surtout en EEG, car le principal effet mesuré est dû aux courants secondaires qui se propagent à travers la tête, de la matière grise jusqu'au cuir chevelu. On obtient alors, avec le logiciel OpenMEEG, des matrices de gain réalistes. On montre d'ailleurs des résultats comparant les effets électromagnétiques produits par le modèle sphérique et le modèle BEM dans la figure 4.7.

On utilise les logiciels BrainVISA et BrainSuite2 pour extraire des informations anatomiques dans le repère de l'IRM. Disposant de plus de fonctions internes paramétrables, le logiciel BrainVISA segmente mieux l'interface matière grise / matière blanche. On l'utilise alors pour faire le modèle distribué de sources. Par contre, BrainSuite2 est actuellement le meilleur logiciel pour la segmentation des surfaces des tissus biologiques de la tête. Ce logiciel contribue ainsi à la réalisation d'un problème direct de grande qualité en permettant d'archiver des maillages réalistes avec plus de 300 000 faces (150 000 nœuds). L'utilisation

d'autant d'informations anatomiques est un avantage par rapport aux modèles plus simplistes, qui considèrent la tête comme une sphère homogène, mais elle augmente la difficulté dans l'obtention de matrices de gain réalistes.

Le programme BrainEntropy a été développé pour répondre aux besoins MEG/EEG de la communauté des neurosciences concernée par le problème inverse en MEG/EEG. Écrit dans l'environnement MatLab il offre une interface simple proposant à l'utilisateur plusieurs outils. On peut notamment préparer les données du problème direct pour calculer des matrices de gain avec OpenMEEG. Il est aussi possible d'analyser et de visualiser plusieurs réponses du problème inverse en EEG, MEG et MEG/EEG. De plus, le simulateur de BrainEntropy permet de vérifier les matrices de gain et d'analyser l'efficacité des méthodes de localisation de sources grâce à une quantification utilisant l'analyse ROC.

L'utilisation des méthodes de minimisation de la norme L_p et de minimisation de l'entropie de Shannon ont été utilisées pour résoudre le problème inverse. L'avantage de la méthode L_p est dans la possibilité de choisir le paramètre p . La valeur optimale de ce paramètre ($p=1.6$) caractérise une méthode de régularisation qui n'est ni trop focale ($p=1$) ni trop étendue spatialement ($p=2$). Les deux méthodes acceptent la fusion des modalités EEG et MEG ce qui permet de contraindre fortement le problème inverse pour obtenir une distribution de sources plus réaliste. Grâce à la qualité des matrices de gain (EEG et MEG), le problème inverse reçoit peu d'information mutuelle de la part des deux modalités comme le montre la figure 4.13. Ainsi, il n'est peut être pas nécessaire d'utiliser des méthodes qui extraient l'information mutuelle comme par exemple celles publiées dans [14].

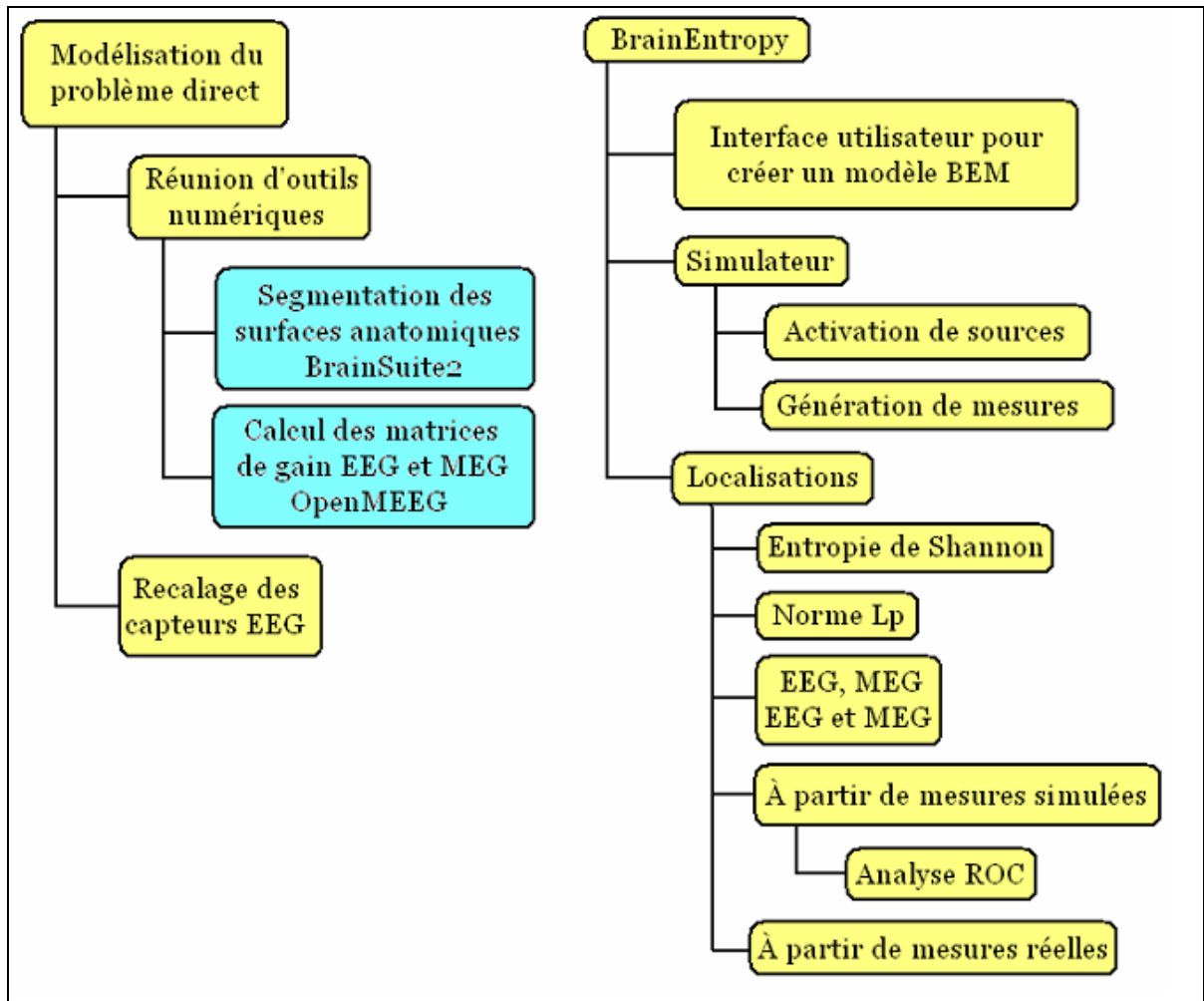


Figure 4.14 Cette figure montre l'organisation des principales contributions du travail effectué pendant cette maîtrise. Les contributions personnelles sont représentées par des boîtes jaunes et l'utilisation d'outils déjà existants par des boîtes bleues. On note un premier travail (à gauche) qui consiste à résoudre le problème direct le plus réaliste possible, et un second travail (à droite) qui contient des outils pour résoudre et vérifier le problème inverse dans l'environnement du logiciel BrainEntropy.

L'acquisition simultanée d'une troisième modalité ajouterait encore plus d'informations a priori pour estimer une distribution de sources encore plus réaliste. On pense notamment à l'imagerie optique qui mesure à la surface du crâne les activités hémodynamiques. Cependant la modélisation de cette technique d'imagerie reste encore très complexe pour pouvoir être fusionnée avec l'EEG et la MEG.

Ce travail a permis de construire un environnement qui regroupe des informations EEG et MEG nécessaires à une localisation de l'activité cérébrale. Une base de données de plusieurs sujets est archivée sous le format du modèle BrainEntropy. Dans le cadre d'autres travaux menés au LATIS, BrainEntropy a été l'outil initial pour manipuler facilement toutes les informations et fonctions du problème. Parmi les développements actuels, on note le passage de ce programme vers Matlab 2008 qui améliore la gestion des données puisque Matlab 2008 intègre beaucoup mieux la notion de classe et d'objet. Cette nouvelle version implémentera aussi une méthode de résolution du problème inverse plus puissante (MEM) basée sur la parcellisation de l'espace des sources et le maximum d'entropie qui a été étudié dans cette maîtrise.

Rapport-Gratuit.com

ANNEXE I

INSTALLATION DE OPENMEEG

4.11 Installation sous Linux

Dans un terminal, placez-vous dans un répertoire de travail pour télécharger les fichiers sources avec la commande suivante :

```
$svn checkout svn ://scm.gforge.inria.fr/svn/openmeeg
```

Téléchargez ensuite les bibliothèques ATLAS et ATLAS-devel. La figure 1.1 est une copie d'écran du site 'RPM Search' qui permet de télécharger presque n'importe quel RPM pour n'importe quelle distribution de Linux. Prenez-y les deux bibliothèques et Cmake à l'occasion. Dans mon cas avec une machine de 32 bits et une distribution Fedora 7, j'ai dû prendre les packages :

- atlas-3.6.0-11-fc6.i386.rpm
- atlas-devel-3.6.0.11.fc6.i386.rpm
- cmake-2.4.7-linux-i386.rpm

Décompresser ces fichiers avec la commande :

```
$rpm -ivh nom_du_package.rpm
```

dans l'emplacement préconisé pour cela : `usr/lib/atlas`

Cmake est un outil capable de lier les sources aux bibliothèques pour créer des fichiers objets et un Makefiles. Appelez 'ccmake' avec comme argument le dossier trunk de openmeeg récemment téléchargé.

```
$/usr/local/cmake-2.4.7-linux-i386/bin/ccmake /home/yann/openmeeg/trunk
```

Fixer les options dans Cmake comme sur la figure 1.2. Enfin tapez `$make install` pour compiler et créer les exécutables dans le répertoire que vous avez indiqué dans le champs `$CMAKE_INSTALL_PREFIX` de Cmake. Si l'option `BUILD_TEST` a été mise à ON dans Cmake, il est possible de taper `$make test` pour voir si l'installation est réussie.


RPM Search - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ?

http://rpm.pbone.net/index.php3/stat/2/simple/2

RPM SEARCH AT RPM.PBONE.NET


all platforms RPM's quick find RPM's last transfer



SEARCH
 NEWS
 DIRECTORIES
 GUEST
 ABOUT
 FAQ
 VARIOUS
 DIARY
 FORUM
 DONATE

Red Hat
Learn about Red Hat innovation in products, services and support.
www.redhat.com

Ads by Google

Donate PBONE via PAYPAL

donate \$10

Highest Result 26K5655
73GB SAS 2.5" 10K RPM
HardDrive Part # 26K5655
www.HardDrivesDirect.com

P1 Overnight 26K5655
73GB SAS 2.5" 10K RPM
HardDrive PN# 26K5655
www.P1Overnight.com

Have to maintain SLED?
Freeze your SLED machines! Try Deep Freeze Linux.
www.faronics.com/linux

Affordable Linux Laptops
Enjoy Stability & Power Of Linux
Excellent Service & Free Shipping!
www.System76.com

Ads by Google

Please enter searched expression
atlas & atlas-devel [Simple RPM Search](#)

IMPORTANT: If You disable cookie then advanced search may not work properly

Fedora 8 Fedora 7 Fedora 6 Fedora 5 Fedora 4 Fedora 3
 Fedora 2 Fedora 1 Fedora Other Rawhide RedHat 9 RedHat 8.X
 RedHat 7.X RedHat 6.X RedHat 5.X RedHat Other RedHat EL 5 RedHat EL 4
 RedHat EL 3 RedHat EL 2.1 TurboLinux PLD Mandriva 2007 Mandriva 2006
 Mandrake 10.X Mandrake 9.X Mandrake 8.X Mandrake 7.X Mandrake Other Conectiva
 OpenSuSE SuSE 10.X SuSE 9.X SuSE 8.X SuSE 7.X SuSE Other
 ALTLinux Engarde Caldera Trustix Arklinux ASPLinux
 Aurox StartCom WhiteBox CentOS PCLinuxOS Other

Results per page 20 40 60 80 100

Sort results Ascending Descending

File Name Provide Summary Description Change Log Requires

Search for rpms which contain given file name (1) e.g. libhistory.so
 New look for FTP search.
 Search for rpms by name.
 Search for rpms by name in FTP world resources (2) e.g. rpm-4*i386.rpm

Terminé

Copie d'écran du site RPM Search pour obtenir atlas et atlas-devel, et Cmake.

4.12 Installation sous windows

```

yann@localhost:~/openmeeg_build
Page 1 of 2
ATLAS_INCLUDE_PATH /usr/include
ATLAS_LIB /usr/lib/atlas/libatlas.so.3
BUILD_RPM OFF
BUILD_SHARED ON
BUILD_TEST ON
BUILD_TOOLS ON
CMAKE_BACKWARDS_COMPATIBILITY 2.4
CMAKE_BUILD_TYPE Debug
CMAKE_INSTALL_PREFIX /home/yann/openmeeg/install
ENABLE_PACKAGING ON
EXECUTABLE_OUTPUT_PATH
LIBRARY_OUTPUT_PATH
MATLAB_WRAP OFF
PYTHON_WRAP OFF
TEST_HEAD3 OFF
USE_ATLAS ON
USE_MKL OFF
ATLAS_INCLUDE_PATH: Path to a file.
Press [enter] to edit option
Press [c] to configure Press [g] to generate and exit
Press [h] for help Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
CMake Version 2.4 - patch 7

```

Configuration de Cmake pour Atlas.

Les prochains sous chapitres décrivent, étape par étape, comment installer OpenMEEG dans l'environnement Microsoft Windows. À travers cette annexe, on commentera les outils qui permettent de compiler les codes sources.

Cygwin

Il est apparu que, même si OpenMEEG fonctionne avec le DOS de Windows, ce logiciel est beaucoup plus stable avec Cygwin et plus bavard dans l'affichage des erreurs. Commencer ainsi par installer Cygwin à partir du site : <http://www.cygwin.com/>. Dans le but d'exécuter OpenMEEG dans un programme Matlab (Comme ce fût le cas dans ce travail) il est possible d'exécuter Cygwin dans Matlab avec la commande `dos('C :cygwin\ Cygwin.bat')` et de rendre l'exécution d'un Shell automatique au lancement de Cygwin. Pour cela on doit indiquer le chemin d'un fichier qui contient des commandes dans le fichier caché `.batchrc` :

```
sh -x commande.sh; logout;
```

Le fichier `.batchrc` est situé dans le répertoire 'user' de Cygwin (exemple : `C:\cygwin\home\Y Potiez`).

On peut, par exemple, écrire les commandes énumérées à la section 'Commande pour Linux' (en remplaçant '/' par \) dans le fichier commande.sh. Pour exécuter OpenMEEG à l'ouverture de Cygwin.

Subversion : Téléchargement d'OpenMEEG

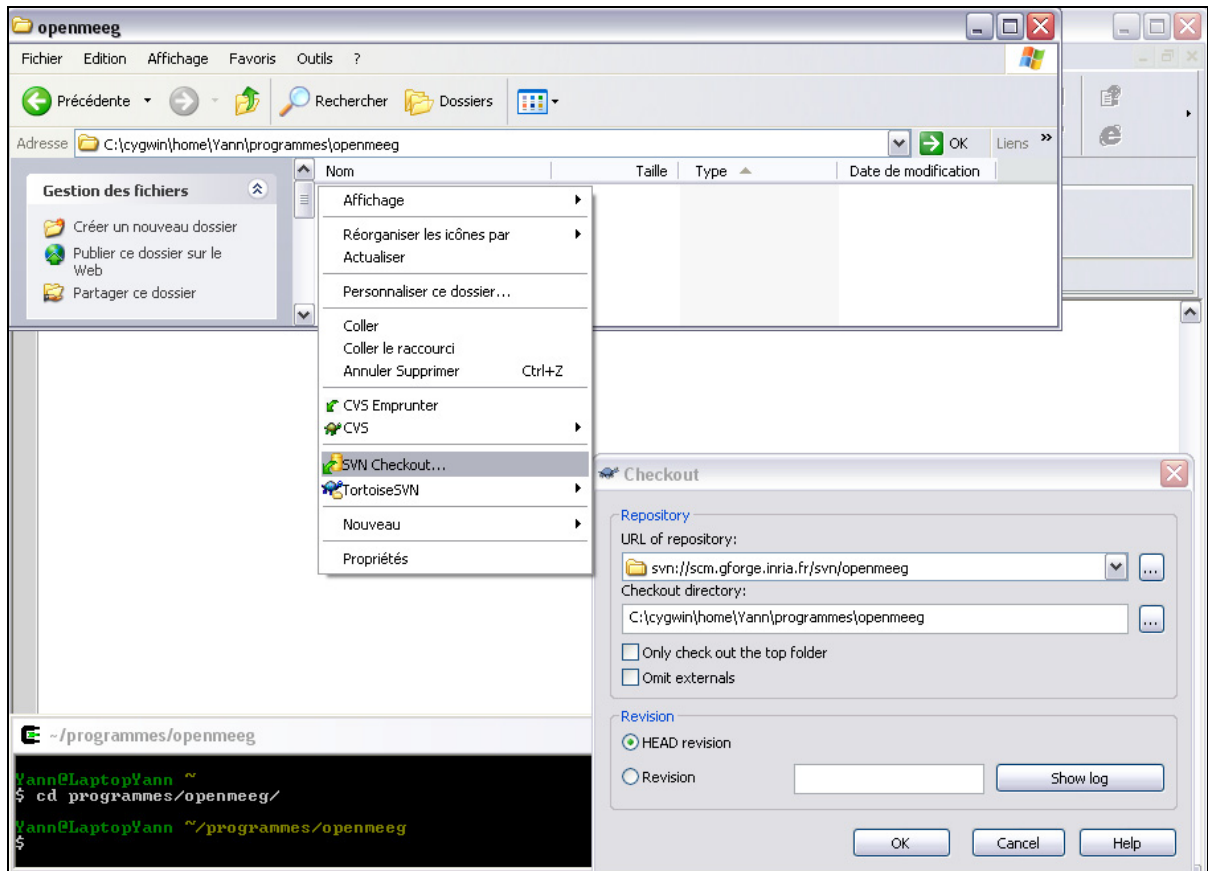
OpenMEEG est téléchargeable par l'intermédiaire de Subversion (SVN). Subversion est un logiciel libre qui permet la mise à jour de codes sources provenant de laboratoires ou d'organismes souhaitant distribuer leurs travaux. Vous pouvez télécharger et installer Subversion à partir du site : http://subversion.tigris.org/project_packages.html

Par la suite, créer deux répertoires imbriqués dans l'espace utilisateur de cygwin :

```
\programmes\openmeeg
```

Téléchargez OpenMEEG dans le répertoire openmeeg qui vient d'être créé en cliquant droit dessus, en sélectionnant SVN checkout et en saisissant l'adresse suivante :

```
https://gforge.inria.fr/scm/?group_id=435
```

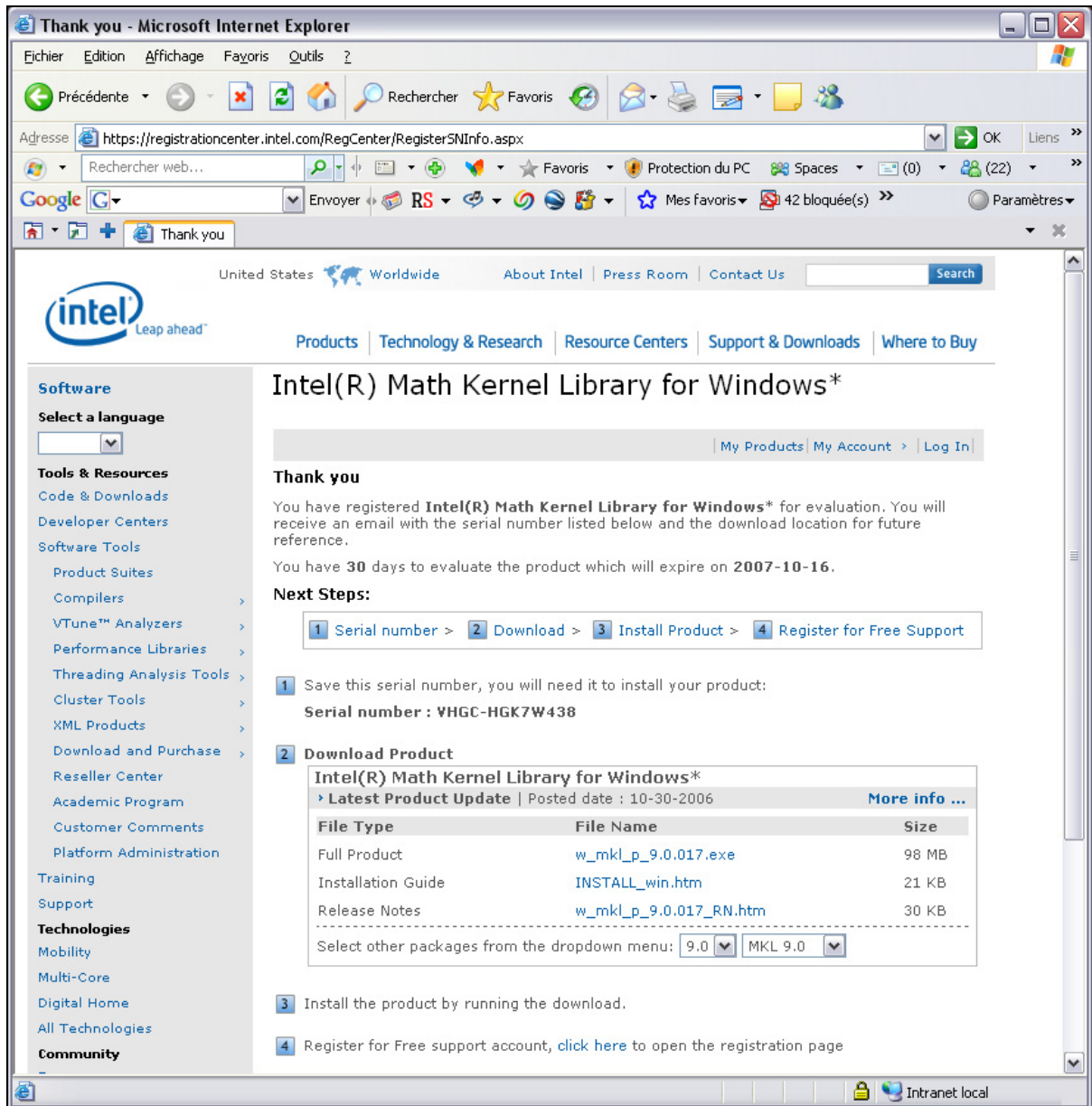


Téléchargement d'OpenMEEG avec SVN.

Librairie mathématique MKL

Certains calculs d'OpenMEEG nécessitent que le code source C++ soit compilé avec la librairie mathématique MKL. Attention l'usage de la librairie MKL ne peut pas servir à des fins commerciales. Vous pouvez l'obtenir en vous enregistrant sur le site suivant pour obtenir une clé avant le téléchargement :

<https://registrationcenter.intel.com/EvalCenter/EvalForm.aspx?productid=795>



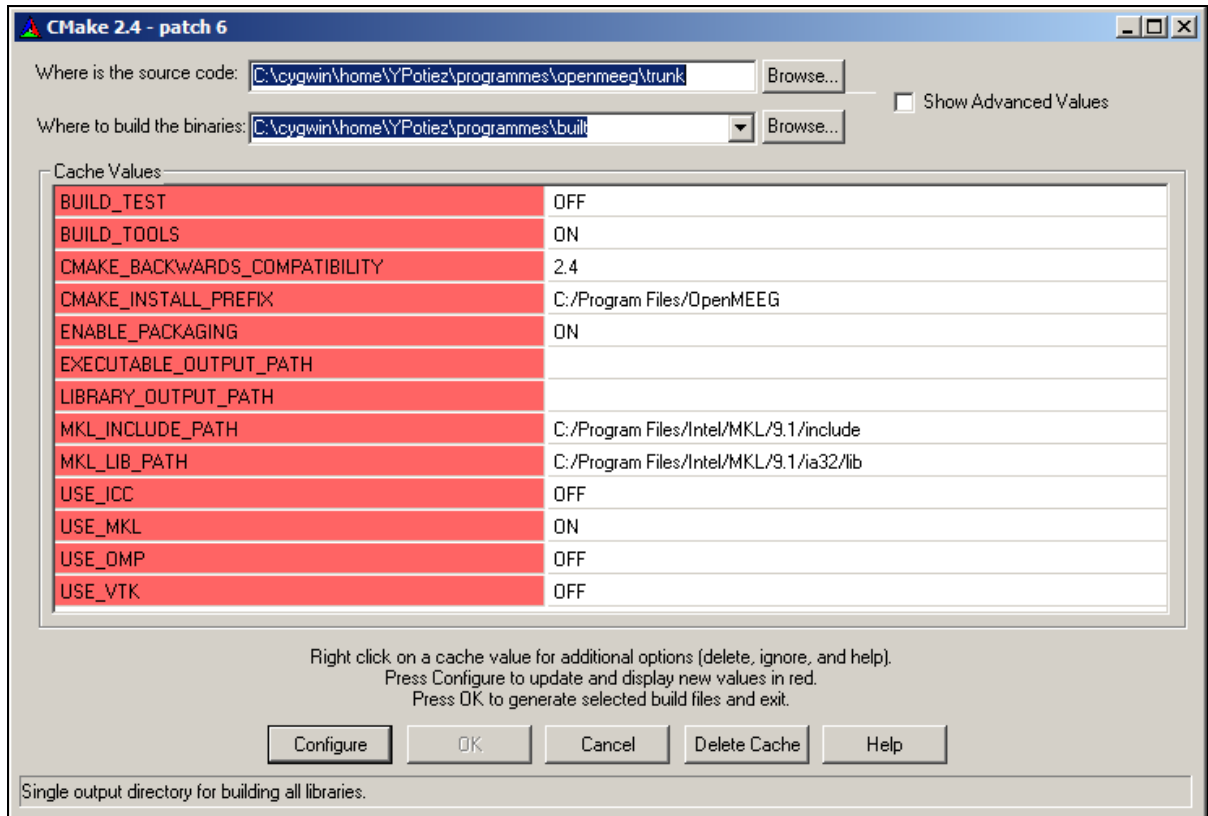
Page web pour télécharger MKL et sa clef.

Cmake est un logiciel qui permet de créer un projet compilable par un logiciel de compilation. Cmake peut se télécharger sur le site officiel :

<http://www.cmake.org/HTML/Download.html>

Premièrement cliquer sur 'Configure'. Cmake vous propose alors de choisir un compilateur. Choisissez Visual Studio 8 2005 (vous l'installerez en suivant l'étape suivante). Cmake a besoin de connaître le chemin des codes sources (trunk), et le chemin dans lequel il écrira

tous les fichiers utiles à la compilation d'OpenMEEG (built). De plus, vous devez indiquer où se trouve le répertoire des headers, et la librairie de MKL avant d'appuyer encore une fois sur le bouton 'Configure', puis 'OK'. Inscrivez les chemins de la même façon que sur la figure 1.5.



Configuration de Cmake pour OpenMEEG.

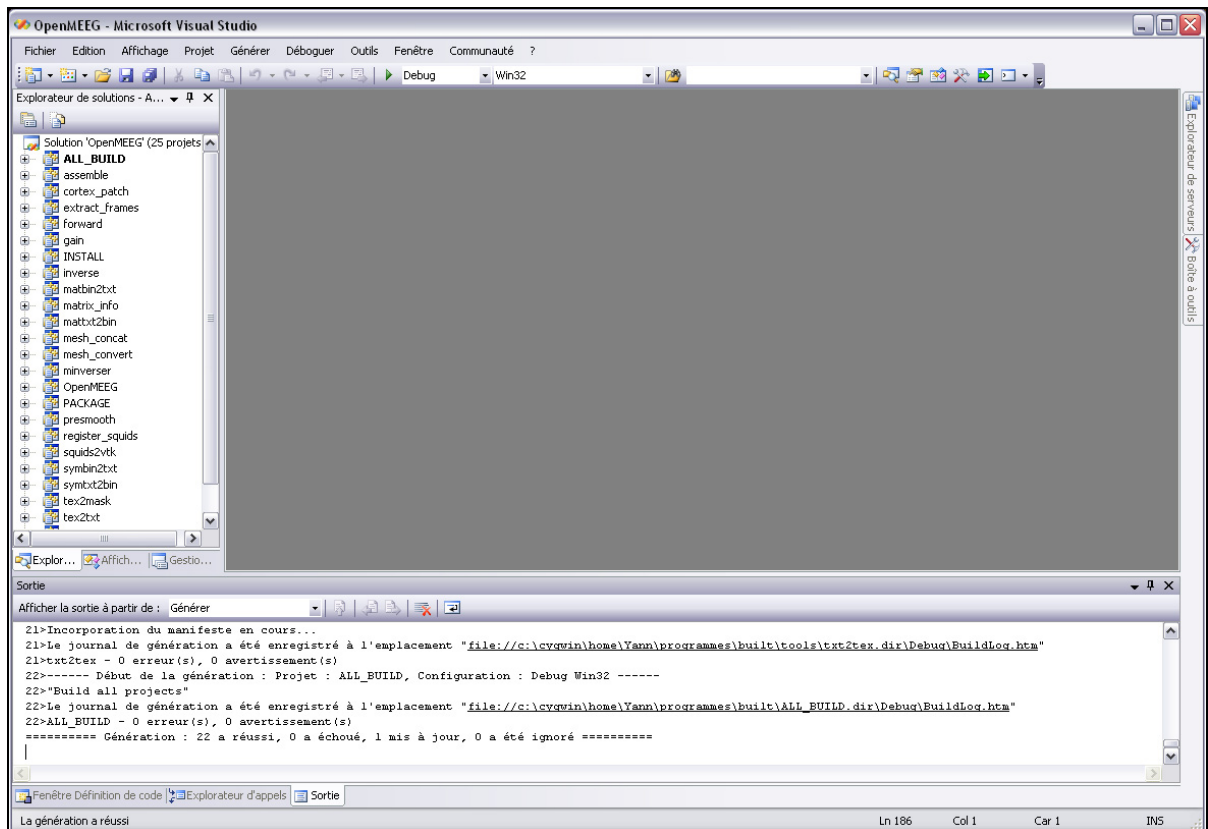
Visual studio

Lorsque Cmake a créé les fichiers du projet dans le répertoire built, vous pouvez compiler le tout avec Visual Studio. Étant étudiant à l'ÉTS (Montréal) il est possible de télécharger Visual Studio 2005 sur MSDN :

<http://www.msdn.com/search/SchoolSearchIntl.aspx>

Quand Visual Studio 2005 sera installé, double cliquer sur le fichier projet OpenMEEG.sln conçu par Cmake. Compiler le tout comme indiqué sur la figure 1.6 (Générer => Générer ALL_BUILD). Vous pouvez aussi dans visual studio regarder comment sont écrits les codes

sources C++. Finalement, vous trouverez les fichiers exécutables dans le nouveau répertoire 'debug' de 'built' à copier puis coller le dossier 'trunk' d'OpenMEEG.



Compilation d'OpenMEEG avec Visual Studio

4.13 3 Commandes pour Linux

On résume ici les lignes de commandes qui permettent de construire des matrices de Gain EEG/MEG.

```
#cd openmeeg/trunk
./om_assemble -rhsPOINT data/HeadModels/sujet/sujet.geom
data/HeadModels/sujet/sujet.cond data/HeadModels/sujet/sujet.posrhs
data/IntermediateFiles/sujet/sujet.rhs
./om_assemble -LHS data/HeadModels/sujet/sujet.geom
data/HeadModels/sujet/sujet.cond
data/IntermediateFiles/sujet/sujet.lhs

./om_minverser data/IntermediateFiles/sujet/sujet.lhs
```

```
data/IntermediateFiles/sujet/sujet.lhsinv
```

```
#EEG
```

```
./om_assemble -vToEEG data/HeadModels/sujet/sujet.geom
data/HeadModels/sujet/sujet.cond
data/Computations/sujet/sujet.patches
data/IntermediateFiles/sujet/sujet.v2eeg
```

```
./om_gain -EEG data/IntermediateFiles/sujet/sujet.lhsinv
data/IntermediateFiles/sujet/sujet.rhs
data/IntermediateFiles/sujet/sujet.v2eeg
data/IntermediateFiles/sujet/sujet.EEG
```

```
#MEG
```

```
./om_assemble -vToMEG data/HeadModels/sujet/sujet.geom
data/HeadModels/sujet/sujet.cond
data/Computations/sujet/sujet.squids
data/IntermediateFiles/sujet/sujet.v2meg
```

```
./om_assemble -sToMEG_point data/HeadModels/sujet/sujet.posrhs
data/Computations/sujet/sujet.squids
data/IntermediateFiles/sujet/sujet.s2meg
```

```
./om_gain -MEG data/IntermediateFiles/sujet/sujet.lhsinv
data/IntermediateFiles/sujet/sujet.rhs
data/IntermediateFiles/sujet/sujet.v2meg
data/IntermediateFiles/sujet/sujet.s2meg
data/IntermediateFiles/sujet/sujet.MEG
```

```
./om_matbin2txt data/IntermediateFiles/sujet/sujet.MEG
data/IntermediateFiles/sujet/sujet.MEGtxt
```

Remarques :

- 1) À la fin du processus, il est possible d'écrire la matrice de Gain en format texte. Pour lire cette matrice dans Matlab il faudra utiliser la fonction load de Matlab.
- 2) NB2 : Pour lire les matrices binaires de sortie dans Matlab il faudra utiliser la fonction load_bin présente dans les outils d'OpenMEEG.

ANNEXE II

INSTALLATION DE BRAINVISA

4.14 1 Installation et configuration de BrainVISA

Cette section est dédiée à l'installation du logiciel BrainVISA, à sa configuration, à l'ajout du package de "Création d'un model EEG" et à l'ajout d'une base de données. Il est important de procéder dans l'ordre de ce document pour ne pas avoir d'erreurs. Bien que BrainVISA soit principalement développé sur une plateforme LINUX, il fonctionne aussi avec Microsoft Windows. La description de l'installation suivante est faite pour Windows mais peu de paramètres changent pour une installation sous LINUX, vous pouvez le constater en allant sur le site de BrainVISA.

Installation

Téléchargez le fichier SHFJ_pack-stable-win32-3.0.2.zip à partir de l'adresse suivante :

<http://brainvisa.info/downloadpage.html>,

puis extraire tout son contenu. Exécuter le fichier 'setup.exe'. L'installation se fait automatiquement, à la fin cliquer sur le bouton "done". Attention Matlab 7 ou une version supérieure doit être préalablement installée sur la machine.

- Exécution :

Pour exécuter BrainVISA double cliquer simplement sur le fichier BrainVISA.bat. Pour Lancer Anatomist (le module de visualisation) utiliser le fichier anatomist.bat.

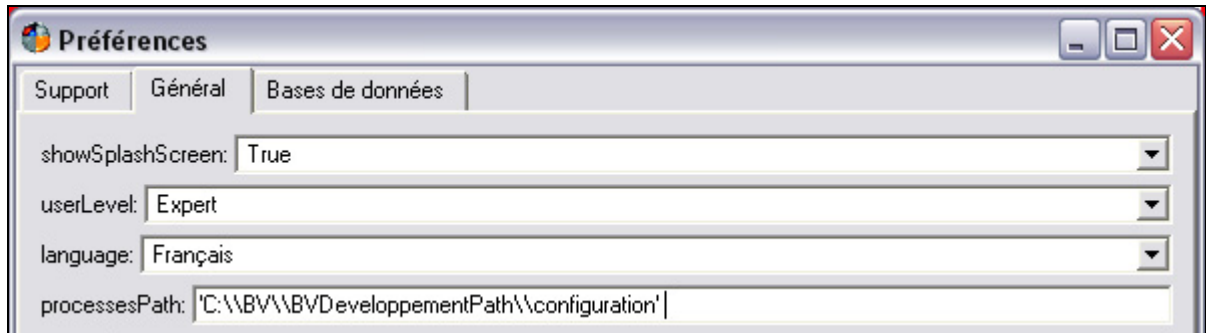
- Configurations :

Dans Préférences > Général > mettre le champs Language à "*French*" et le Champs *userLevel* à "*Expert*". Fermer l'application puis relancez-la pour que la modification de la langue soit prise en compte. Ceci est à effectuer lors de la première utilisation de BrainVISA.

Ajout du package "localisation de sources"

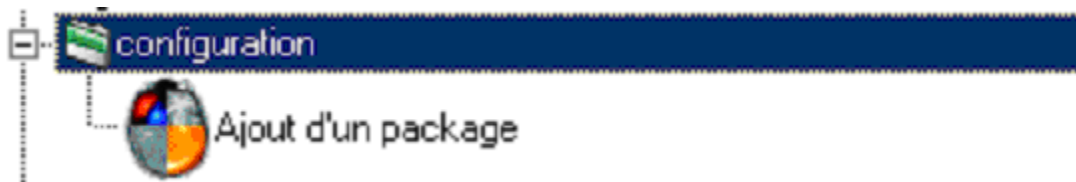
- Définition des chemins BrainVisa :

Le module configuration permet le paramétrage automatique des chemins d'accès BrainVisa. Ouvrir la fenêtre de configuration des chemins, dans Préférences -> General. Dans processesPath mettre l'adresse du dossier configuration :

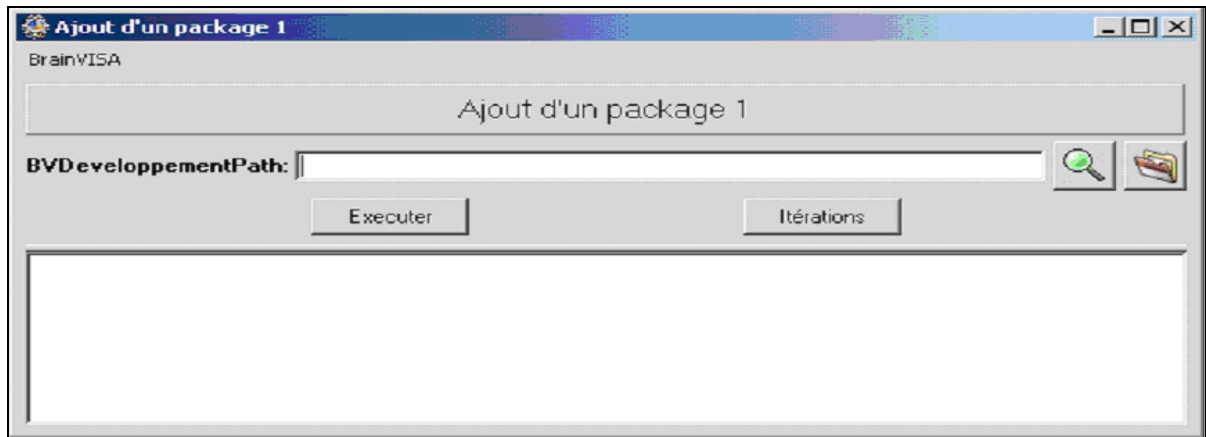


ProcessPath

Les adresses de BrainVISA comportent un slash en plus à ajouter dans le chemin tel qu'illustré dans la figure précédente. Fermer BrainVISA et redémarrer l'application. Dans la partie gauche de la fenêtre principale apparaît un répertoire nommé configuration :



Double-cliquer sur la fonction " Ajout d'un package " et La fenêtre suivante apparaît :



Ajout d'un package

Pour sélectionner le nouveau répertoire package, il suffit de presser le bouton :



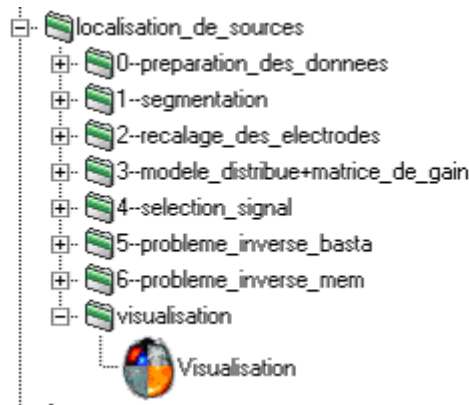
Le package doit contenir au moins les dossiers : matlab, processes et types. Appuyez ensuite sur Exécuter. Fermer et redémarrer BrainVisa pour permettre la prise en compte du package dont nous nous servirons par la suite.

- Matlab :

Pour rendre Matlab exécutable dans BrainVISA, il faut spécifier son chemin d'accès dans les préférences lors de la première utilisation de BrainVISA (ce chemin est mémorisé pour les autres exécutions de BV). Dans Préférences > Général saisir le chemin d'accès du répertoire contenant l'exécutable de Matlab dans le champs matlabExecutable, et 14 dans le champs matlabRelease. Exemple :



Redémarrer BrainVisa pour que les nouveaux paramètres soient pris en compte. Voici l'arborescence qui apparaît dans la partie gauche de la fenêtre principale de BrainVISA lorsque la configuration a été réalisée.



- Ajout d'un chemin dans les préférences :

Dans une étape précédente nous avons ajouté les Paths nécessaires au bon fonctionnement du package de localisation de sources. Or il s'est avéré qu'il en manquait un. Ajouter-le manuellement dans Préférences > Général en saisissant le chemin du répertoire contenant notamment le fichier 'calculG.m' dans le champs matlabPath. Ce répertoire est en fait un sous répertoire contenu dans le kit Brainstorm. Voici la liste des chemins qui doivent être inscrits dans le champs matlabPath (les racines peuvent ne pas être forcément les mêmes que cet exemple)

```
'C:\BV\BVDeveloppementPath\matlab'
'C:\BV\BVDeveloppementPath\matlab\BASTA'
'C:\BV\BVDeveloppementPath\matlab\Brainstorm'
'C:\BV\BVDeveloppementPath\matlab\debug'
'C:\BV\BVDeveloppementPath\matlab\leeglab'
'C:\BV\BVDeveloppementPath\matlab\MEM'
'C:\BV\BVDeveloppementPath\matlab\validation'
'C:\BV\BVDeveloppementPath\matlab\Brainstorm\Custom\pack\_BV'
```

- Configuration d'Anatomist :

- Langue :

Dans Settings > Préférences mettre le champs Language à " fr ". Fermer la fenêtre des Préférences puis presser sur Settings > Save Preferences (fermer l'application puis la relancer pour que la modification de la langue soit prise en compte)

- IMPORTANT : Convention pour l'orientation des vues :

Dans Réglages > Préférences, sur l'onglet Fenêtres sélectionner la "Convention neurologique" pour avoir les vues telles que la gauche réelle se trouve à gauche, et la droite réelle à droite (contrairement à la convention radiologique). Fermer la fenêtre des Préférences puis presser les boutons Réglages > Sauver les préférences.

Ajout d'une base de données

- Base de données :

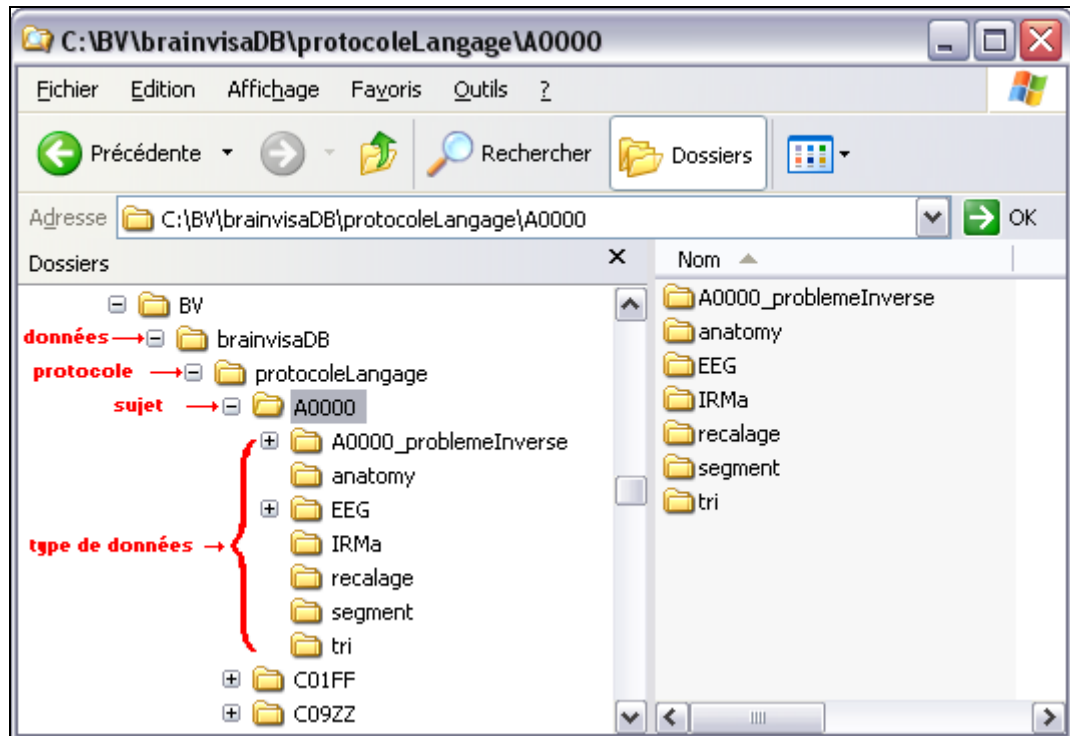
IMPORTANT Avant de commencer à utiliser les fonctionnalités de BV, l'utilisateur doit indiquer une base de données dans laquelle se trouve les fichiers de données, et dans laquelle il pourra stocker les fichiers de sorties. Pour cela, il faut dans un premier temps mettre les données brutes dont on dispose (EEG, IRM, MEG, etc.) dans une arborescence de type :

<nom_de_protocole>/<identifiant_patient>/<type_de_données>.

Exemple :

Pour que le système intègre une base de données il faudra procéder selon la méthode suivante:

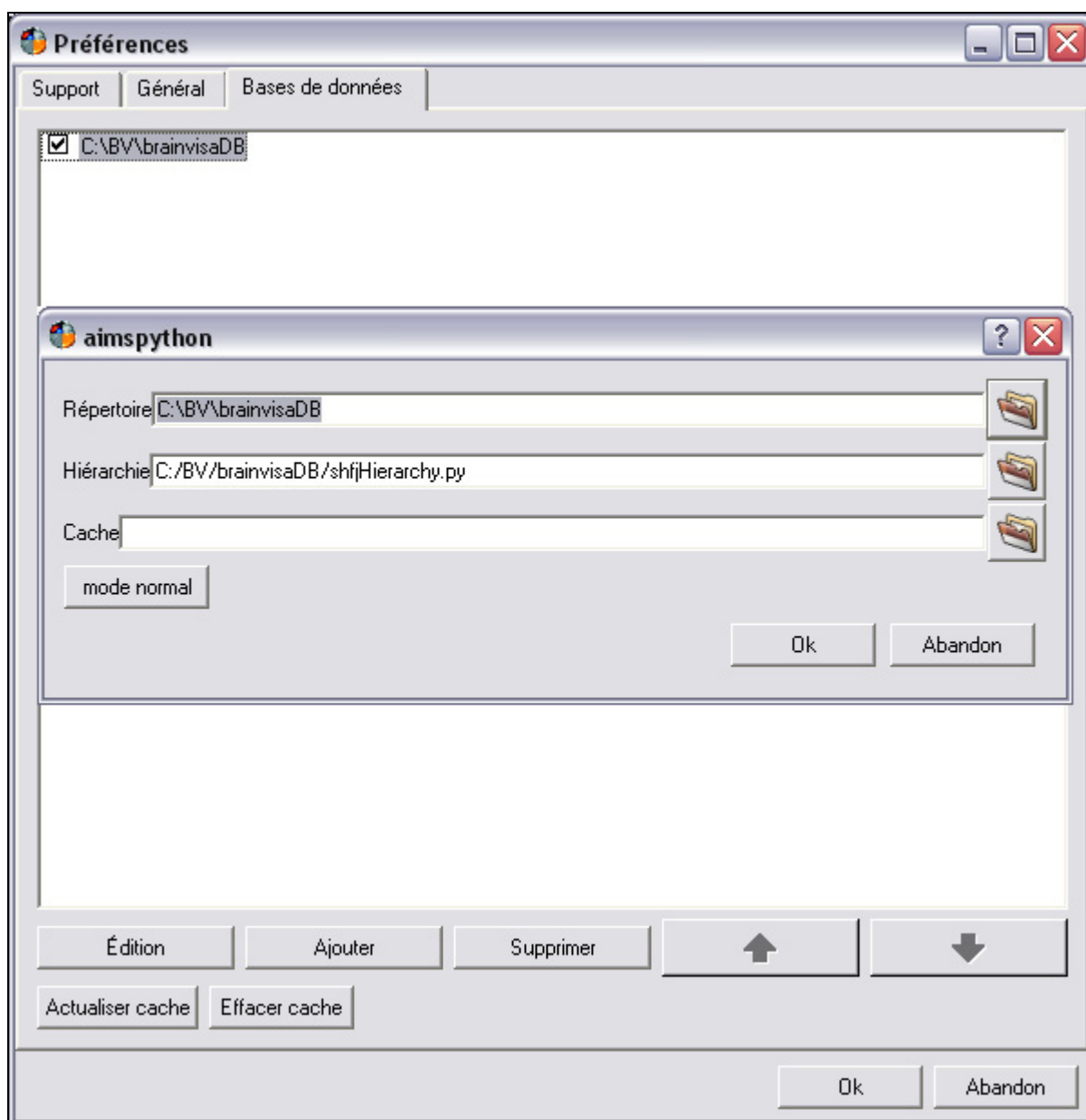
Dans Préférences > Bases de données, faire " ajouter ". Deux champs sont à remplir : "répertoire" et "hiérarchie".



Organisation des données

Dans le champs "répertoire", indiquer le répertoire contenant la base de données tel que le montre la figure suivante.

On doit ensuite lier une hiérarchie (fichier : shfjHierarchy.py) à cette base de données, pour cela presser le bouton " mode expert " et spécifier le chemin d'accès du fichier de la hiérarchie liée à la base de données. On place généralement ce fichier dans le répertoire de la base de données. Vous pouvez garder le même fichier hiérarchie pour chaque nouvelle base de données mais il est important de le spécifier à nouveau à chaque fois. Quand les champs répertoire et hiérarchie sont complétés appuyer sur OK, et valider les préférences. Si vous n'avez pas scrupuleusement suivi les indications ci-dessus et qu'un message d'erreur apparaît, vous devrez alors supprimer la base de données que vous venez de créer et presser le bouton " vider le cache". Recommencez ensuite l'opération de remplissage des champs "répertoire" et "hiérarchie".



Configuration d'une base de données dans les préférences.

ANNEXE III

CLASSES ET ATTRIBUTS DE BRAINENTROPY

1) La classe Modele

```
function M = Modele( pathname ) if nargin == 0;
    Modele.repertoires = '';
    Modele.parametres = [];
    Modele.sources = [];
    Modele.parcelles = struct([]);
    Modele.selection = struct();
    Modele.H0 = struct();
    Modele.headmesh = [];
    Modele.mesh = [];
    Modele.modalites = struct([]);
    Modele.pb_inverses = [];
    Modele.unites = struct([]);
    Modele.type = '';
    M = class(Modele,'Modele');
else
    load( pathname );
    M = actual_model;
end
```

Repertoires	Chemins des fichiers qui ont construit le modèle. (Voir 5.2)
Parametres	Indique le nombre de surfaces corticales
sources	Tableau de classes surface corticale
parcelles	Parcellisation pour MEM.
selection	
H0	
headmesh	maillage de la tête
mesh	
modalites	Tableau de classes Modalites
pb_inverses	Classe pb_inverses pour la sauvegarde des intensités selon la méthode.
unites	
type	'EEG' 'MEG' ou 'MEEG'

Description des attributs de la classe modèle.

2) La classe Modalite

```

function M = Modalite( obj )
%constructeur d'une modalite ou copie d'une modalite
if nargin == 0
    mod.gain = [];
    mod.data = [];
    mod.type = '';
    M = class(mod,'modalite');
elseif isstruct(obj)
    mod.gain = obj.gain;
    mod.data = obj.data;
    mod.type = obj.type;
    M = class(mod,'modalite');
elseif isa(obj,'modalite')
    M = obj;
end

```

gain	Matrice de Gain de la modalité
data	Classe Data
type	Type de modalité, 'EEG' ou 'MEG'

Description des attributs de la classe modalité.

3) La classe Data

```

function D = Data( obj )
if nargin == 0
    data.Capteurs = [];
    data.Sampling = 0;
    data.Mesures = [];
    data.Mesures_simul = [];
    data.Duree = 0 ;
    data.N_c = 0 ;
elseif isa(obj,'Capteur')
    data.Capteurs = obj;
    data.Sampling = 0;
    data.Mesures = [];
    data.Mesures_simul = [];
    data.Duree = 0 ;
    data.N_c = length(obj);
else
    disp('mauvais appel du constructeur data');
end

```

```
end D = class( data , 'Data');
```

Capteurs	Tableau de Classe Capteur
Sampling	Taux d'échantillonnage des données
Mesures	Mesures réelles relatives à la modalité
Mesures_simul	Mesures générées par le simulateur.
Duree	Nombre d'échantillons
N_c	Nombre de capteurs de la modalité

Description des attributs de la classe data.

4) La classe Capteur

```
function capteur = Capteur( val , name, par)
if nargin==0
    cap.nom = '';
    cap.ID = 0;
    cap.Position1 = [0 0 0];
    cap.Position2 = [0 0 0];
    cap.Orientation1 = [0 0 0];
    cap.Orientation2 = [0 0 0];
elseif nargin>1
    cap.ID = val;
    cap.nom = name;
    cap.Position1 = par.position1;
    cap.Position2 = par.position2;
    cap.Orientation1 = par.orientation1;
    cap.Orientation2 = par.orientation2;
end capteur = class(cap,'Capteur');
```

nom	nom du capteur
ID	identification du capteur.
Position1	Position X Y Z
Position2	
Orientation1	Orientation O1x O1y O1z (MEG)
Orientation2	Orientation O2x O2y O2z (MEG)

Description des attributs de la classe capteur.

5) La classe Surface corticale

```

function Surf_Cort = Surface_Corticale( obj )
    surf_cort.Dipoles = [];
    surf_cort.Adjacence = sparse([]);
    surf_cort.selection = [];
    surf_cort.alpha = [];
    surf_cort.intensites = [];
    surf_cort.simulation = [];
    if nargin == 0
        surf_cort.TailleP = 0;
        surf_cort.TailleD = 0;
        surf_cort.Parcelle = [];
        surf_cort.Pactives = [];
        surf_cort.Dipoles = [];
        surf_cort.ParcelleDe = [];
        surf_cort.Maillage = struct();
        surf_cort.Surface = struct();
    elseif isstruct( obj )
        surf_cort.TailleP = obj.tailleP;
        surf_cort.TailleD = obj.tailleD;
        surf_cort.Pactives = obj.actives;
        for i = 1:surf_cort.TailleP
            surf_cort.Parcelle(i) = Parcelle;
        end
        surf_cort.ParcelleDe = zeros(1:obj.tailleD);
        surf_cort.Surface = obj.surf;
    end Surf_Cort = class(surf_cort,'Surface_Corticale');

```

Dipoles	Tableau de classes dipoles
Adjacence	Matrice symétrique contenant la distance entre chaque vertex du maillage cortical
selection	
alpha	
intensites	Matrice Sources*échantillons. Une colonne est la carte d'activation des sources pour un temps donné provenant du problème inverse .
simulation	Matrice Sources*échantillons. Une colonne est la carte d'activation des sources pour un temps donné provenant du simulateur.
TailleP	Nombre de parcelles
Taille D	Nombre de dipoles Modalites
Parcelle	Tableau de classes parcelle.
ParcelleDe	Vecteur indiquant les parcelles corespondant aux dipôles

Maillage	Maillage du cerveau ou d'une hémisphère
Surface	
Pactive	Parcelles actives

Description des attributs de la classe surface corticale.

6) La classe Dipôle

```

function Dipole = Dipole( obj )
if nargin==0
    Dip.Position = [0.0 ; 0.0 ; 0.0];
    Dip.Orientation = [1 ; 1 ; 1 ];
    Dip.Index = 0;
    Dip.Voisinage = [];
    Dip.Lead_Field = [];
    Dipole = class(Dip,'Dipole');
elseif isnumeric( obj )
    Dip.Position = [0.0 ; 0.0 ; 0.0];
    Dip.Orientation = [1 ; 1 ; 1 ];
    Dip.Index = obj;
    Dip.Voisinage = [];
    Dip.Lead_Field = [];
    Dipole = class(Dip,'Dipole');
elseif isstruct(obj)
    Dip.Position = obj.Position;
    Dip.Orientation = obj.Orientation;
    Dip.Index = obj.Index;
    Dip.Voisinage = obj.Voisinage;
    Dip.Lead_Field = obj.Lead_Field;
    Dipole = class(Dip,'Dipole');
elseif isa(obj,'Dipole')
    Dipole = obj;
end

```

Position	Position X Y Z du dipole
Orientation	Orientation Ox Oy Oz du dipôle.
Index	Numero du dipôle
Voisinage	Index des dipoles voisins
Lead_Field	Lead_field de la matrice de gain du dipôle.

Description des attributs de la classe dipôle.

BIBLIOGRAPHIE

- [1] M. Clerc, T. Papadopoulo. INRIA, Inverse problems in fonctionnal brain imaging. Cours donné à l' ENS CACHAN, (2007).
- [2] J. Kybic, M. Clerc, O. Faugeras, R. Keriven, T. Papadopoulo. Generalized head models for MEG/EEG : boundary element method beyond nested volumes. *Physics in Medicine and Biology*, volume 51, pages 1333–1346, (2006).
- [3] J. Kybic, M. Clerc, T. Abboud, O. Faugeras, R. Keriven, T. Papadopoulo. A Common Formalism for the Integral Formulations of the Forward EEG Problem, *IEEE Transactions on medical imaging*, volume 24, pages 17–28, (2005).
- [4] B. Renault. *Imagerie cérébrale fonctionnelle électrique et magnétique*. Hermes sciences publication, Lavoisier, (2004).
<http://e2phy.in2p3.fr/2002/presentations/renault.pdf>
- [5] Mark. F. Bear, Barry. W. Connors, Michael A. Paradiso, *Neuroscience à la découverte du cerveau*. Éditions Pradel, (2002).
- [6] B. D. Rao, K. Kreutz-Delgado. An Affine Scaling Methodology for Best Basis Selection. *iee Transaction on signal processing*, volume 47, No.1, (1999).
- [7] E. Lapalme, JM. Lina, J. Mattout. Data-driven parceling and entropic inference in MEG. *NeuroImage* 30, pages 160–171, (2006). www.elsevier.com/locate/ynimg
- [8] O. David, L. Garnero. Time-Coherent Expansion of MEG/EEG Cortical Sources. *NeuroImage* 17, pages : 1277–1289, (2002).
- [9] L. Garnero. *Les bases physiques et physiologiques de la Magnétoencéphalographie et de l'Électroencéphalographie*. CNRS-UPR640-LENA
- [10] A. H. Register. *A guide to MATLAB Object-Oriented Programming*, Chapman & Hall/CRC, (2007).
- [11] <ftp://ftp-sop.inria.fr/odyssee/Team/Sylvain.Vallaghe/grenoble.ppt>
- [12] <http://geometryalgorithms.com>
- [13] <ftp://ftp-sop.inria.fr/odyssee/Publications/2007/clerc-gramfort-et-al :07.pdf>
- [14] S. Baillet, L. Garnero, G. Marin, and JP. Hugonin. Combined MEG and EEG Source Imaging by Minimization of Mutual Information, *iee transaction on biomedical engineering*, Vol. 46, No. 5, (1999).

- [15] C. Grova, J. Daunizeau, J.-M. Lina, C.G. Bénar, H. Benali, and J. Gotman. Evaluation of EEG localization methods using realistic simulations of interictal spikes *NeuroImage* 29, 734 – 753, (2006).
- [16] http://lsiit-miv.u-strasbg.fr/contenu/fichiers_page/MEG-EEG-online.pdf