

Simulation des déformations du cerveau.

Projet de Recherche M1if - année 2011-12

Contexte :

L'utilisation croissante de systèmes de navigation pour l'aide à la chirurgie a permis de faciliter les interventions ainsi que la planification des gestes chirurgicaux. Néanmoins, dans le cas de la neurochirurgie où le geste opératoire doit être très précis, les systèmes actuels sont limités par les déformations survenant en per-opératoires. Ce phénomène, connu sous le terme de "Brain Shift" traduit le mouvement des structures cérébrales après ouverture de la boîte crânienne. Le recalage entre les examens pré-opératoires et la position du patient en salle d'opération est donc imprécis, et les informations fournies par le système de navigation deviennent partiellement obsolètes.

Problématique :

Au LIRIS, nous avons développé un ensemble d'outils permettant de prendre en compte ce phénomène, améliorant ainsi toute la chaîne de traitement [1]. Ces outils ont été développés et validés en Matlab pour des raisons de rapidité de conception, mais il sera nécessaire de les intégrer dans une application en langage C/C++ ou Python, avec utilisation des bibliothèques graphiques classiques.

À partir d'images IRM pré-opératoires du cerveau, nous obtenons une première segmentation qui est ensuite raffinée pour obtenir un maillage volumique des différentes structures cérébrales. Nous disposons d'un modèle bio-mécanique permettant de simuler les déformations et les interactions fluide/structure (analyse en éléments finis et résolution du système), mais les possibilités offertes restent encore trop limitées.

Le cadre de ce projet est donc la manipulation de maillages triangulaires et tétraédriques, utilisation de bibliothèques pour l'Algèbre Linéaire (LAPACK+BLAS par exemple). On abordera aussi : bio-mécanique, imagerie médicale, modèles physiques (discrets et continus). Il s'agit donc d'étudier cette première technique afin de l'améliorer pour obtenir un comportement plus réaliste.

Travail à réaliser, chacune des parties étant indépendante, bien que liée à toutes les autres :

1. *Partie MEPP:*

Environnement de développement C++ et MEPP (incluant les bibliothèques CGAL, OPENGL, QGLviewer) <http://liris.cnrs.fr/mepp/>

- Installation de la plateforme MEPP (et bibliothèque CGAL)
- Prise en main, regarder les tutoriels sur comment créer un nouveau composant, une nouvelle action
- Comprendre la structuration de données pour un maillage, étudier en particulier la représentation demi-arête, sommets et faces (intégrée dans MEPP)

- Créer une petite démo pour charger et parcourir les éléments d'un maillage 2D dans MEPP, avec cette structuration
- Représenter dans MEPP le résultat d'une simulation (cf fichiers de sortie de la simulation sous Matlab)
 - le maillage déformé suite à l'application de forces
 - le déplacement des points sous forme de vecteur, ou de code couleur

2. Partie Génération de Maillages

Attention, ici, les maillages doivent considérer des interactions entre le cerveau (solide) et les ventricules (liquide). Il faut donc s'assurer de la bonne gestion du contact entre les deux, de manière géométrique (en réalité => 1 maillage, 2 labels ou +).

- Dans l'existant, les maillages 2D sont générés à partir des contours externes, à l'aide du logiciel Triangle (<http://www.cs.cmu.edu/~quake/triangle.html>).
- Les fichiers sont de type .poly en entrée et ceux obtenus grâce à l'option -en en sortie.
- Convertir de manière automatique ces maillages en sortie dans un format compatible avec MEPP et idéalement avec le programme Matlab existant (cf ci-dessous, mais en gros : 1 maillage pour le solide et 1 pour le liquide, avec gestion adéquate de l'interface).
- [+] Intégrer des trous ou divisions internes à l'objet (les ventricules, membranes du cerveau)

3. Partie Matlab/Octave :

- Étudier le code de simulation existant en Matlab. En identifier le fonctionnement dans ses grandes lignes, les parties utiles au chargement et manipulation des maillages (se restreindre à l'interface entre les données et la sortie de la simulation).
- Porter le code sous Octave, si nécessaire
- Intégrer la représentation des maillages 2D en demi-arête dans le programme existant
 - pour la simulation
 - [+] pour le calcul de l'estimateur d'erreur
- [+] Créer une interface graphique pour faciliter le choix des différentes options de la simulation (mode compressible, incompressible, caractéristiques des matériaux, choix des fichiers de configuration, gestion des boucles d'optimisation...)

4. Partie Raffinement/Maillages Non Conformés :

On aimerait gérer des maillages cerveau/ventricules de différentes résolutions. En pratique, on souhaite raffiner uniquement le maillage (liquide ou solide) qui pose problème, et ne pas transmettre le raffinement si ce n'est pas nécessaire. Conséquence : un triangle du solide peut partager une arête avec plusieurs triangles du liquide, et vice-versa.

- Mettre en place une procédure pour raffiner des maillages (sous dans MEPP, soit dans le pgm Matlab - via le logiciel Triangle)
 - raffinement régulier (on divise chaque arête en 2)
 - [+] raffinement adaptatif (seuls les triangles qui posent problème sont subdivisés). Cela sous-entend calculer un estimateur d'erreur de la simulation mécanique (méthode existante).

- Étudier comment générer des maillages non-conformes, modifier (compléter) éventuellement la structure de données en demi-arête
- Modifier le programme Matlab/Octave pour prendre en compte cette nouvelle possibilité (gestion de voisins multiples pour une arête), en 2D ([+] en 3D).

5. Partie Comparaison de Maillages

Dans cette partie, on souhaite comparer des maillages du même objet, à différentes résolutions, ou plutôt surtout le résultat de la simulation avec ces différents maillages.

- Utiliser soit :
 - Triangle pour raffiner les maillages
 - ce qui aura été développé en Partie 4.
- Pour un sommet dans un maillage, identifier le point correspondant (soit un sommet, soit le point le plus proche dans l'objet) dans le maillage de résolution la plus fine, considéré comme référence.
- Pour ces 2 points, comparer les données de la simulation si elles existent, sinon estimer une valeur en fonction des valeurs des voisins (interpolation).
- [+] Calculer la norme H1 ou L2 de l'erreur, pour montrer la convergence.

6. [+] Partie Modification de l'IRM initiale

- À partir de l'IRM du cerveau (cela peut être sur une seule coupe 2D pour commencer), segmenter la surface des différentes parties (cortex, ventricules), utiliser pour cela un logiciel existant.
- Générer les maillages internes
- Associer chaque sommet d'un triangle/tétraèdre aux pixels de l'image
- Simuler les déformations du maillage, obtenir un champ de déformation
- Déformer l'image initiale pour reporter ces déformations dans l'IRM, et fournir au médecin une vue actualisée du cerveau pendant l'opération.

Mots-clés :

Modélisation de Maillages, Animation et Simulation Physique 2D/3D, Imagerie Médicale, Réalité Augmentée.

Coordonnées du responsable de stage :

Fabrice JAILLET - fabrice.jaillet@liris.cnrs.fr
 Laboratoire LIRIS -Equipe SAARA
<http://liris.cnrs.fr/saara/>
 Bâtiment Nautibus, 8 bd Niels Bohr
 Université Lyon I - 69622 Villeurbanne Cedex

Références:

[1]. Araya, Barrenechea, Galdames, Jaillet and Rodríguez: Maillage adaptatif et analyse éléments finis du couplage fluide/structure : applications aux déformations du cerveau, SURGETICA'2007, pp 117-121

