

Charlotte BRICOUT  
*Charlotte.bricout@polytech-lille.net*

Nathan MARTIN  
*Nathan.martin@polytech-lille.net*

Informatique-Microélectronique-Automatique  
Ecole Polytechnique Universitaire de Lille

Année universitaire 2014-2015

---

Rapport intermédiaire de Projet de fin d'études  
**Modélisation d'un robot chirurgical déformable pour la  
simulation et le contrôle**

---



Tuteur pédagogique : M. Jérémie DEQUIDT

# Table des matières

<b>Introduction</b>	<b>3</b>
<b>1 . Contexte et cadre du projet</b>	<b>4</b>
1.1 L’Institut National de Recherche en Informatique et Automatique . . . . .	4
1.1.1 Présentation de l’INRIA . . . . .	4
1.1.2 Le centre Inria Lille - Nord Europe . . . . .	4
1.1.3 L’équipe-projet Shacra . . . . .	4
1.2 Le King’s college de Londres . . . . .	4
1.2.1 Présentation du King’s college . . . . .	4
1.2.2 Le centre de recherche en robotique . . . . .	5
1.3 Présentation du projet . . . . .	5
<b>2 . Contraintes matérielles</b>	<b>6</b>
2.1 Le logiciel Blender . . . . .	6
2.2 Le logiciel GID . . . . .	6
2.3 La plate-forme logicielle SOFA . . . . .	6
<b>3 . Réalisations techniques</b>	<b>7</b>
3.1 Contrôle des robots . . . . .	7
3.1.1 La commande des robots rigides . . . . .	7
3.1.2 La commande de robot déformable . . . . .	7
3.2 Elaboration d’un modèle pour la simulation . . . . .	7
3.2.1 Modélisation du prototype . . . . .	7
3.2.2 Mise en simulation avec SOFA . . . . .	8
3.3 Création d’un prototype . . . . .	10
<b>4 . Travail prévisionnel</b>	<b>11</b>
4.1 Amélioration de la simulation . . . . .	11
4.2 Tests réels en pression . . . . .	11
4.3 Interfaçage entre la simulation et le prototype . . . . .	11
4.4 Etude de faisabilité de migration de technologie de contrôle . . . . .	11
4.5 GANTT du projet . . . . .	11
<b>5 Annexes</b>	<b>13</b>
5.1 Annexe 1 : Scène SOFA . . . . .	13
5.2 Diagramme GANTT du projet . . . . .	16

# Introduction

Dans le cadre de notre cinquième année à Polytech Lille en spécialité Informatique-Microélectronique-Automatique, nous réalisons actuellement un projet de fin d'études. Ce projet, nous le menons en collaboration avec l'équipe Shacra de l'INRIA Lille ainsi qu'avec l'équipe du centre de recherche en robotique du King's college de Londres. En effet, les deux équipes entament un partenariat sur un projet de création et de contrôle de robot déformable à visée chirurgicale. Le King's college possède actuellement un prototype et nous sollicite pour l'implémenter en simulation sur la plate-forme logicielle SOFA ainsi que pour réaliser son contrôle pneumatique. Par ailleurs, on envisage, à terme, de le contrôler hydrauliquement. Ce rapport est un rendu intermédiaire permettant de suivre la progression de notre travail ainsi que de fournir un planning pour la suite du projet. Nous rendrons compte, ici, du contexte de ce projet, des avancées techniques que nous avons réalisé jusqu'à présent ainsi que du planning pour les deux prochains mois.

# 1. Contexte et cadre du projet

## 1.1 L’Institut National de Recherche en Informatique et Automatique

### 1.1.1 Présentation de l’INRIA

L’Institut National de Recherche en Informatique et en Automatique (INRIA) est un institut de recherche public français qui oriente ses recherches dans les domaines de l’informatique et les mathématiques des sciences du numérique. Les activités scientifiques d’Inria sont regroupées en cinq domaines de recherche :

- STIC (Sciences et Technologies de l’Information et de la Communication) pour les sciences de la vie et de l’environnement
- Mathématiques appliqués, calcul et simulation
- Perception, cognition et interaction
- Réseaux, systèmes et services, calcul distribué
- Algorithmes, programmation, logiciels et architectures

L’institut est composé de 8 centres de recherche répartis dans toute la France et nous travaillerons, durant ce projet, avec le centre Lille - Nord Europe.

### 1.1.2 Le centre Inria Lille - Nord Europe

Le centre de recherche Inria Lille - Nord Europe, situé à la Haute Borne, compte 17 équipes-projet. Celles-ci se concentrent sur les Sciences et Technologies de l’Information et de la Communication. Elles conçoivent des logiciels innovants pour le commerce ou la logistique, ou encore, mettent au point des simulateurs médicaux ou des interfaces facilitant l’interaction entre l’humain et la machine. Elles orientent leurs recherches sur quatre grandes thématiques :

- Internet des données et Internet des objets
- Génie logiciel pour systèmes éternels
- Modèle patient dynamique
- Couplage perception/action pour l’interaction homme-machine

### 1.1.3 L’équipe-projet Shacra

Parmi les dix-sept équipes du centre de Lille, l’équipe Shacra oriente ses recherches sur le thème Images, modèles et algorithmes pour la médecine et les neurosciences. Le rôle de cette équipe-projet est de mener des recherches sur des problèmes scientifiques liés à la simulation médicale numérique. Pour ce faire, ils utilisent une plate-forme logicielle : SOFA et collaborent avec des instituts chirurgicaux. L’objectif commun des projets de l’équipe est de fournir des outils de diagnostics, de planification ou de guidage basés sur la simulation physique.

## 1.2 Le King’s college de Londres

### 1.2.1 Présentation du King’s college

Le King’s College de Londres est un établissement d’enseignement supérieur britannique et cofondateur de l’Université de Londres. Il est également un membre fondateur du Russell Group, qui constitue le plus grand centre dédié à l’éducation des professionnels de la santé en Europe. Ce complexe regroupe six centres de conseil en recherche médicale. Aujourd’hui, Le King’s College est organisé en neuf écoles réparties en quatre campus à Londres.

L’équipe Shacra est en collaboration avec le centre de recherche en robotique (Centre of Robotics Research).

### 1.2.2 Le centre de recherche en robotique

Les activités du centre de recherche en robotique portent sur :

- la robotique et automatisation
- la chirurgie et la réadaptation robotique
- les systèmes robotisés médicaux
- la manipulation des systèmes robotiques
- la cinématique et les mécanismes
- l'art mimétique inspiré des mécanismes et des robots
- le développement de systèmes d'inspection et de surveillance
- les systèmes biomimétiques, Neuraux et cognitifs



Bon nombre des projets de recherche sont inter-disciplinaires et le centre entretient de nombreux partenariats avec des entreprises industrielles et des universités telles que The Imperial College, Newcastle, Birmingham ou encore l'Université de Birkbeck de Londres.

Shacra entretient une collaboration étroite avec l'équipe dirigée par le professeur Kaspar Althoefer. Cette équipe a pour thématique de recherche :

- *Robotics and Intelligent Systems*, Robotique et systèmes intelligents
- *Embedded Sensing and Classification*, Classification de systèmes intégrés
- *Medical and Industrial Applications*, Applications médicales et industrielles

## 1.3 Présentation du projet

Les équipes Shacra et du centre de recherche en robotique vont travailler pendant deux ans sur ce projet de robot déformable. Actuellement, les robots se constituent de squelettes rigides dont les articulations sont mises en mouvement par des moteurs. Cependant, il est également possible de mettre une structure en mouvement en la déformant. Les robots déformables présentent un avantage non négligeable par rapport aux robots rigides : leur flexibilité. En effet, cette caractéristique leur confère un faible encombrement et une capacité à se faufiler dans des endroits très étroits.

L'objectif de ce projet est de contrôler des robots non-rigides qui pourraient faire progresser la chirurgie non invasive. A terme, le laboratoire envisage de créer un endoscope pneumatique. Actuellement, le King's college possède un prototype destiné à un contrôle pneumatique. En effet, le contrôle est effectué en emplissant ou vidant des volumes d'air. Le projet pourrait évoluer avec un contrôle hydraulique qui serait plus précis grâce à une mesure du liquide injectée dans les cavités.

On se propose d'intégrer le projet afin de réaliser le contrôle pneumatique du prototype par le biais d'une simulation temps réel grâce à la plate-forme logicielle SOFA.

Pour cela, on se fixe les objectifs suivants :

- Développer un modèle 3D du robot et le mailler en éléments finis grâce à un logiciel de Conception Assistée par Ordinateur
- Mettre en place une simulation SOFA
- Faire un prototype pneumatique simplifié de l'endoscope afin de valider le contrôle du robot par le biais d'une simulation informatique. Pour mener à bien le projet, une partie de contrôle bas-niveau des régulateurs de pression sera nécessaire
- Réaliser le contrôle pneumatique du prototype
- Interfacer la simulation avec le prototype réel
- Valider le modèle sur le robot développé au King's College de Londres
- Faire une étude de faisabilité pour passer de la technologie pneumatique à hydraulique

## 2. Contraintes matérielles

Nous souhaitons dans un premier temps réaliser un modèle du prototype. C'est à dire créer par le biais d'un logiciel de CAO un modèle en 3 dimensions et le mailler. Le maillage est une modélisation géométrique du modèle par des éléments finis d'une certaine forme géométrique (Ici, nous utiliserons des tétraèdres). Il est utile pour simplifier un système dans l'optique de simulations, de calculs ou de représentation graphique.

Pour créer notre modèle, nous avons à disposition les logiciels Blender et GiD.

### 2.1 Le logiciel Blender

Blender est un logiciel libre de modélisation, d'animation et de rendu en 3D. Il dispose de fonctions avancées de modélisation, de sculpture 3D, de texturage, d'animation 3D et de rendu. Le logiciel prend en charge diverses simulations physiques telles que les particules, les corps rigides, les corps souples et les fluides. Par ailleurs, ce logiciel ne permet pas le maillage du modèle et il est donc nécessaire d'utiliser SOFA en complément de Blender.



### 2.2 Le logiciel GiD

GiD est un logiciel adaptatif universel pré et post-processeur pour les simulations numériques. Il a été conçu pour couvrir tous les besoins en simulation numérique de pré à post-traitement : modélisation géométrique, définition efficace des données d'analyse, maillage, transfert de données à un logiciel d'analyse et visualisation des résultats numériques.



Afin de mettre en simulation le modèle 3D, nous utilisons SOFA, décrite ci-dessous.

### 2.3 La plate-forme logicielle SOFA

SOFA est une plate-forme de recherche et développement dédiée aux simulations physiques interactives et en particulier à la simulation médicale. La plate-forme regroupe de nombreux algorithmes issus de domaines de recherche variés. Des modèles de nature différente peuvent être combinés de façon à produire des simulations complexes qui restent compatibles avec des temps de calcul optimaux.



## 3. Réalisations techniques

### 3.1 Contrôle des robots

#### 3.1.1 La commande des robots rigides

Un système articulé rigide est caractérisé par une structure arborescente articulée simple ou multiple dont les liaisons sont mobiles les unes par rapport aux autres. Ce système a pour objectif de mener l'organe terminal vers un lieu géométrique imposé par la tâche.

La synthèse de la commande du robot nécessite la connaissance des relations entre ses grandeurs d'entrées et de sorties. L'ensemble de ces équations constitue le modèle mathématique du robot. Si les équations sont extraites de la physique, le modèle est appelé modèle de connaissance, et si ces équations découlent des observations disponibles sur le système, le modèle s'appelle modèle de représentation.

Les modèles dynamiques des bras manipulateurs sont décrits par un ensemble d'équations mathématiques qui portent des informations dynamiques de ces manipulateurs. Ils peuvent être simulés sur un ordinateur dans le but de synthétiser une commande conditionnée par des performances désirées. L'ensemble des équations dynamiques peut être déterminé par des lois mécaniques classiques Newtoniennes et Lagrangiennes. Les approches d'Euler Lagrange et Newton-Euler permettent d'aboutir aux équations du mouvement des robots.

#### 3.1.2 La commande de robot déformable

Les robots déformables possèdent un grand nombre de degrés de liberté. Il faut, de ce fait, aborder leur contrôle d'une toute autre manière. En effet, il faut intégrer une modélisation précise des déformations dans le processus de contrôle. Pour cela, il est nécessaire d'utiliser des méthodes numériques précises telles que les éléments finis (FEM). Étant donné le grand nombre de degrés de liberté des modèles, il est impossible d'utiliser les méthodes classiques de contrôle robotique.

Nous allons donc créer un modèle 3D par CAO, qui sera maillé en éléments finis. Nous intégrerons ce modèle sur la plate-forme logicielle SOFA où nous utiliserons la FEM temps réel.

### 3.2 Elaboration d'un modèle pour la simulation

#### 3.2.1 Modélisation du prototype

Grâce à une publication, on connaît les dimensions et spécifications du robot. Celui-ci est représenté sur la figure ci-dessous.

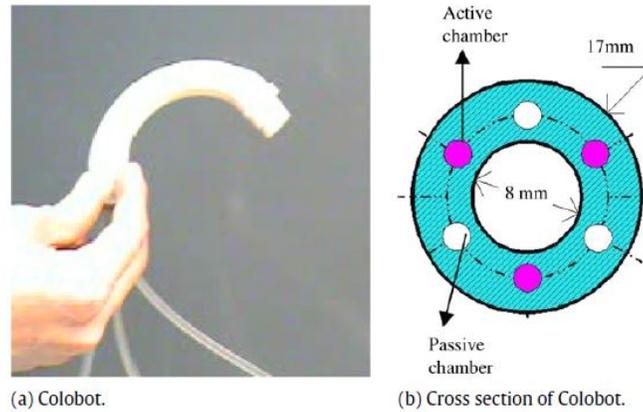


FIGURE 3.1 – Robot déformable et sa section

Le robot est constitué de 6 cavités très fines mais seules trois cavités recevront un volume d'air. La forme particulière du prototype nous oblige à porter une attention toute particulière au maillage des cavités. On se propose dans un premier temps, d'élaborer deux modèles :

- un modèle comportant 2000 à 3000 noeuds qui permettra une simulation temps réel
- un modèle comportant 10 000 à 11 000 noeuds qui permettra d'évaluer les besoins en pression.

On décide d'abord d'utiliser Blender pour élaborer le modèle simplifié. On l'extrait avec l'extension '.obj' afin de réaliser son maillage sur SOFA. Cependant, on s'aperçoit que le maillage est rendu impossible par le trop grand nombre de noeuds et nos machines ne supportent pas les calculs. De plus, un nombre de tétraèdres moins important ne fait pas apparaître les cavités sur le prototype maillé. Par conséquent, on décide d'utiliser le logiciel GID qui permet non seulement de concevoir mais aussi de mailler le système. Cependant, même en prenant soin de mailler plus précisément les cavités par rapport au reste du modèle, le maillage n'est pas non plus concluant avec GID. Finalement, nous décidons, en accord avec le King's college, de créer un prototype deux fois plus large que celui prévu initialement. Ainsi, les cavités sont plus simples à mailler et nous générons deux prototypes : un à 2600 noeuds et un à 10 000 noeuds.

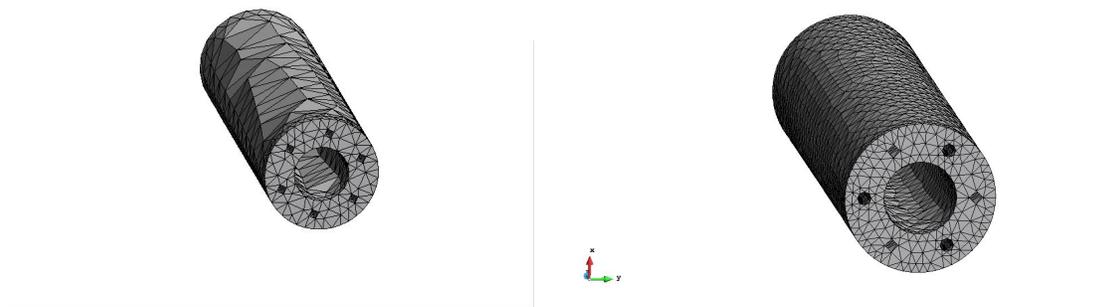


FIGURE 3.2 – Modèle à 2600 noeuds et modèle à 10 000 noeuds

### 3.2.2 Mise en simulation avec SOFA

On souhaite, à présent, mettre en simulation notre modèle exporté de GID au format ".msh". Ce fichier mesh contient les coordonnées des points ainsi que les éléments de l'objet maillé. A partir de ce mesh, on cherche à générer un fichier au format ".xml" afin que les données contenues dans le mesh

soient exploitables dans une scène Sofa. Pour ce faire, on utilise le script suivant qui génère le fichier souhaité :

```
awk 'BEGIN { print "<Mesh position=\""; section=0 } /Coordinates/ { section=1 } (
NF==4&&section==1) { print $2 " " $3 " " $4 } /Elements/ { section+=1; if (
section==2) { print "\\n tetrahedra=\"\" } } (NF==5&&section==2) { print $2-1 " "
$3-1 " " $4-1 " " $5-1 } END { print "\" />" }'
```

On souhaite, dans un premier temps, valider notre modèle et le contrôler grâce à la technologie pneumatique. On crée donc une scène SOFA (Voir Annexe 1) qui permettra de jouer une simulation. SOFA permet le contrôle d'un modèle par le biais d'une interface graphique. Sur cette interface, on peut manuellement définir une trajectoire pour le robot. Grâce à une scène, on calcule la quantité d'air à injecter dans les cavités pour obtenir le déplacement demandé.

SOFA utilise la méthode des éléments finis (FEM) qui permet de résoudre de manière discrète des équations différentielles partielles dont on cherche une solution approchée. Cette méthode directe s'appuie sur le principe fondamental de la dynamique :

$$\frac{M\partial^2 x}{\partial t^2} + F\left(\frac{\partial x}{\partial t}, x\right) = F_{ext}$$

Dans cette méthode, on connaît les forces appliquées au modèle et on cherche la position qu'il atteindra sous la contrainte de ces forces. Notre problématique est légèrement plus complexe car ici, on cherche à connaître la force à appliquer pour atteindre une position donnée. On utilise pour cela une méthode inverse temps réel qui repose sur les éléments finis non-linéaires, permettant une estimation du module de Young. En effet, on ajoute à l'équation précédente des contraintes :

$$\frac{M\partial^2 x}{\partial t^2} + F\left(\frac{\partial x}{\partial t}, x\right) = F_{ext} + \lambda H^T$$

On utilise alors un algorithme itératif d'analyse numérique afin d'estimer la valeur de la contrainte. Cette méthode repose sur une projection du modèle dans l'espace des variables d'optimisation. On valide la méthode par des exemples numériques que l'on applique au modèle.

Dans notre scène, on cherche tout d'abord à sélectionner les trois cavités qui nous intéressent et pour ce faire, on utilise des 'BoxROI'. Les 'BoxROI' sont des boîtes permettant de sélectionner les tétraèdres d'un volume. Ici, on en crée quatre : la première permet de fixer la base du robot et les trois autres permettent de sélectionner les 3 cavités à contrôler. Une fois que les tétraèdres sont sélectionnés, on récupère tous les triangles qui composent les cavités. Cette opération permet de sélectionner la surface au lieu du volume et nous permettra le contrôle par pression. On fixe alors un point que l'on nommera "Goal" et un second point nommé "Effector". "Goal" est le point que va suivre le modèle. "Effector" est un point du modèle qui cherchera à converger vers le "Goal". Une fois ces deux points définis, il nous reste à tester différentes pressions pour connaître la tolérance de notre système ainsi que plusieurs valeurs du module de Young pour avoir la rigidité souhaitée.

Pour mettre en simulation le modèle à 10 000 noeuds, nous ne pouvons pas observer la simulation. En effet, le modèle étant très précis, nous ne pouvons pas utiliser une simulation temps réel car notre machine ne calcule pas assez vite. Nous allons donc programmer une trajectoire pour la sphère "Goal" et faire des captures de la simulation à chaque pas de calcul. Il nous suffira ensuite de convertir l'ensemble des photos en une vidéo. Nous avons, pour cela, modifié le fichier scène afin qu'il génère un fichier texte contenant les coordonnées de la sphère "Goal" durant la simulation. Il nous a ensuite fallu modifier le fichier texte pour y coder la trajectoire désirée et le fichier scène pour qu'il la lise. Ainsi lors du déroulement de la simulation, la sphère adoptera successivement les positions définies dans ce fichier texte et notre robot essaiera de rapprocher son centre outil de celle-ci.

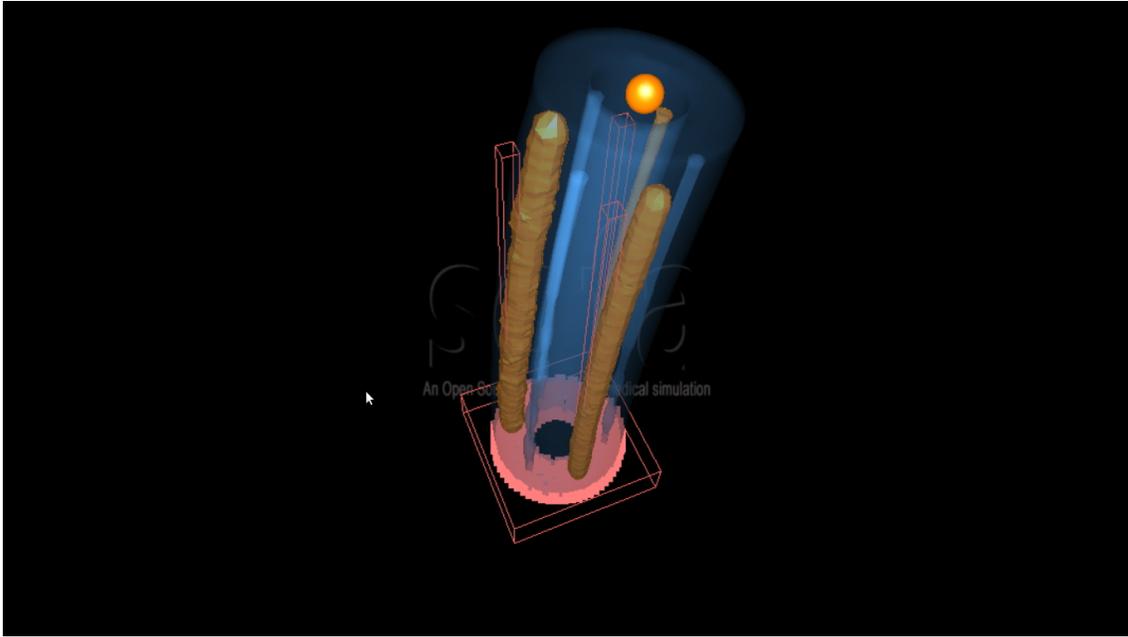


FIGURE 3.3 – Robot simulé sous pression

### 3.3 Création d'un prototype

Nous sommes chargés de réaliser un premier prototype du robot déformable car nous n'avons pas à disposition le prototype du King's college. Pour ce faire, il nous faut concevoir un moule pour couler du silicone et obtenir la géométrie souhaitée. Nous réaliserons ce moule à partir de tuyaux pour la forme du robot et de tiges filetées pour délimiter les cavités. Dans ce moule, nous allons couler du silicone "Dragon Skin" et laisser prendre pour enfin démouler la première partie du prototype. La seconde partie est un disque de 1 cm d'épaisseur qui vient fermer le haut des cavités pour empêcher l'air de passer.

## 4. Travail prévisionnel

### 4.1 Amélioration de la simulation

Actuellement, la simulation est imparfaite car nous rencontrons quelques problèmes de convergence du modèle en simulation. Il sera nécessaire par la suite de tester dans SOFA le modèle pour corriger les erreurs. Il nous faudra également pousser plus loin les tests d'évaluation des besoins en pression.

### 4.2 Tests réels en pression

Il existe actuellement un dispositif de contrôle pneumatique non fonctionnel à l'INRIA de Lille. Il sera nécessaire de le rendre opérationnel et de tester, indépendamment de la simulation, notre prototype avec différente pression. Nous pourrons ainsi ajuster, dans la scène SOFA, les paramètres de pression à appliquer pour être au plus proche de la réalité.

### 4.3 Interfaçage entre la simulation et le prototype

Il est nécessaire d'interfacer le prototype physique avec la simulation pour réaliser le contrôle pneumatique par simulation. C'est à dire, de connecter la partie software avec la partie hardware.

### 4.4 Etude de faisabilité de migration de technologie de contrôle

Un contrôle hydraulique du robot déformable a été envisagé dans le but d'obtenir plus de précision dans ses mouvements. En effet, l'eau est un liquide non compressible que l'on va injecter dans les cavités pour obtenir la déformation voulue. On connaîtra alors précisément la quantité d'eau insérée dans l'une ou l'autre des cavités et on pourra être plus précis quand au mouvement du robot déformable. Nous étudierons la possibilité de migration d'une technologie à l'autre à la fin du projet.

### 4.5 GANTT du projet

Le diagramme GANTT du projet est présenté en Annexe 2. Il reprend les différentes tâches à effectuer durant les mois de Décembre, Janvier et Février.

## Conclusion

Nous aurons d'ici peu un prototype physique du robot ainsi qu'une simulation de celui-ci. La prochaine étape consiste à interfacier les deux parties et à contrôler le robot par le biais de sa simulation. La phase qui va alors avoir lieu peut être délicate. Nous devons en effet nous assurer du bon fonctionnement du prototype ainsi que de sa précision. Nous réaliserons pour cela une série de tests afin d'évaluer les performances de notre système. Nous espérons pouvoir ainsi présenter le système complet à l'équipe du King's college avant la fin de notre projet de fin d'études. De plus, nous envisageons de leur fournir une étude de faisabilité de migration de la technologie pneumatique vers hydraulique.

## 5. Annexes

### 5.1 Annexe 1 : Scène SOFA

```
<?xml version="1.0" ?>
<Node name="root" dt="0.01" gravity="0 -9.81 0">

  <RequiredPlugin pluginName="SoftRobots" />
  <VisualStyle displayFlags="showVisualModels showCollisionModels hideForceFields
showInteractionForceFields showBehaviorModels" />

  <FreeMotionMasterSolver />
  <GenericConstraintSolver maxIterations="500" printLog="1" tolerance="0.00000001"/>
  <CollisionPipeline verbose="0" />
  <BruteForceDetection name="N2" />
  <CollisionResponse response="FrictionContact" />
  <LocalMinDistance name="Proximity" alarmDistance="0" contactDistance="0" />

  <Node name="settings_video" >
  <ViewerSetting resolution="1280 720"/> <!-- 640 480 -->
  </Node>

  <!-- Interactive Mode -->
  <Node name="goal" activated="1" >
  <EulerImplicitSolver firstOrder="1" />
  <CGLinearSolver iterations="100" />
  <MechanicalObject name="goalM0" position="0 0 110"/> <!-- -->
  <!-- to write or read in a text file in order to move our sphere -->
  <ReadState filename="trajectoire.ws" />
  <!-- <UniformMass totalmass="0.0" /> -->
  <Sphere radius="3" group="3"/>
  <UncoupledConstraintCorrection />
  </Node>

  <Node name="VisuGoal" activated="0">
  <!--<IdentityMapping name="default17" />-->
    <OglShader
      vertFilename="shaders/shaderLibrary.gls1"
      fragFilename="shaders/shaderLibrary.gls1" />
  <OglFloat3Variable name="AmbientColor" value="0.0 0.0 0.0" />
  <OglFloat3Variable name="DiffuseColor" value="0.8 0.2 0.2" />
  <!-- <OglFloat3Variable name="SpecularColor" value="1 0.8 0.8" /> -->
  <OglFloatVariable name="SpecularRoughness" value="0.25" />
  <OglFloatVariable name="SpecularReflectance" value="0.05" />
  <OglFloat3Variable name="LightColor" value="1 1 1" />
  <OglFloat3Variable name="LightPosition" value="10 10 -60" />
  <OglFloat3Variable name="LightDirection" value="-0.1 -0.1 0.6" />
  </Node>
```

```

<Node name="forme_U">
<EulerImplicit name="odesolver" />
<SparseLDLSolver name="preconditioner" />

<include href="Mesh_vf.xml" filename="Mesh_vf.xml" name="loader" />
<TetrahedronSetTopologyContainer src="@loader" name="Container"/>
<TetrahedronSetTopologyModifier name="Modifier" />
<TetrahedronSetTopologyAlgorithms template="Vec3d" name="TopoAlgo" />
<TetrahedronSetGeometryAlgorithms drawTetrahedra="false" template="Vec3d"/>

  <!-- ici tu peux appliquer des paramètres pour changer la position de l'objet
avec rx, ry, rz ou dx, dy, dz -->
<MechanicalObject name="tetras" template="Vec3d" showIndices="false"
showIndicesScale="4e-5"/><UniformMass totalmass="0.5" />
<TetrahedralCorotationalFEMForceField template="Vec3d" name="FEM" method="large"
poissonRatio="0.4" youngModulus="180"/> <!--youngModulus="18000000 -->

<!-- ici tu peux modifier les paramètres de la box :
xmin ymin zmin xmax ymax zmax box="-0.1 -0.1 -1.001 0.01 0.01 -0.99" -->
<BoxROI name="boxROI" box="-20 -20 -5 20 20 5" drawBoxes="true" />
<FixedConstraint indices="@boxROI.indices" /> -->

<LinearSolverConstraintCorrection solverName="preconditioner" />

<Node name="surface">
<TriangleSetTopologyContainer name="Container" />
<TriangleSetTopologyModifier name="Modifier" />
<TriangleSetTopologyAlgorithms template="Vec3d" name="TopoAlgo" />
<TriangleSetGeometryAlgorithms template="Vec3d" name="GeomAlgo" />
<Tetra2TriangleTopologicalMapping name="Mapping" object1="../../Container"
object2="../../Container" />

<BoxROI name="cylinder1ROI" template="Vec3d" box="-1.5 -10.75 0 1.5 -13.75 100"
drawBoxes="true" /> <!-- Cavity number 4 -->
<BoxROI name="cylinder5ROI" template="Vec3d" box="-11.97 4.45 0 -8.93 7.55 100"
drawBoxes="true" /> <!-- Cavity number 6 -->
<BoxROI name="cylinder3ROI" template="Vec3d" box="8.97 4.55 0 11.97 7.55 100"
drawBoxes="true" /> <!-- Cavity number 2 -->

<Node name="Visu" activated="true">
<OglModel template="ExtVec3f" name="Visual" color="0.2 0.4 0.6 0.3" />
<IdentityMapping name="default17" />
</Node>
</Node>

<Node name="Effector" activated="true" >
<MechanicalObject name="effectorPoint" position=" 0 0 110 " />
<!-- <QPConstraint indices="0" effectorGoal="@../../goal/goalM0.position"
numActuators="1" actuatorMax="500" actuatorMin="0" actuatorValues="0"/> -->
<ActuatorConstraint index="0" actuator="false"
effectorGoal="@../../goal/goalM0.position" />
<BarycentricMapping name="mapping1" mapForces="false" mapMasses="false"/>
</Node>

```

```
<Node name="Cavity1" activated="true">
<Mesh position="@../..../forme_U/tetras.rest_position"
triangles="@../..../forme_U/surface/cylinder1ROI.trianglesInROI" name="topo"/>
<MechanicalObject name="cavity" />
<SurfacePressureConstraintNoQP triangles="@topo.triangles" actuator="true"
maxPressure="2" visualization="1" showVisuScale="0.0001"/>
<Triangle group="3"/>
<BarycentricMapping name="mapping2" mapForces="false" mapMasses="false"/>
</Node>
```

```
<Node name="Cavity2" activated="true">
<Mesh position="@../..../forme_U/tetras.rest_position"
triangles="@../..../forme_U/surface/cylinder3ROI.trianglesInROI" name="topo"/>
<MechanicalObject name="cavity" />
<SurfacePressureConstraintNoQP triangles="@topo.triangles" actuator="true"
maxPressure="2" visualization="1" showVisuScale="0.0001"/>
<Triangle group="3"/>
<BarycentricMapping name="mapping2" mapForces="false" mapMasses="false"/>
</Node>
```

```
<Node name="Cavity3" activated="true">
<Mesh position="@../..../forme_U/tetras.rest_position"
triangles="@../..../forme_U/surface/cylinder5ROI.trianglesInROI" name="topo"/>
<MechanicalObject name="cavity" />
<SurfacePressureConstraintNoQP triangles="@topo.triangles" actuator="true"
maxPressure="2" visualization="1" showVisuScale="0.0001"/>
<Triangle group="3"/>
<BarycentricMapping name="mapping2" mapForces="false" mapMasses="false"/>
</Node>
</Node>
</Node>
```

## 5.2 Diagramme GANTT du projet

