
Compte-rendu des séances de TP d'haptique

Étudiant :
Idriss CHELIKH

Numéro étudiant :
3704957

Janvier 2021

Table des matières

1	Introduction	2
2	Identification des paramètres mécaniques du Novint Falcon	3
2.1	Estimation des capacités mécaniques du robot	3
2.2	Estimation des paramètres dynamiques de l'interface	3
3	Travaux sur les figures virtuelles	5
3.1	Conception d'un retour de force fidèle au contact mécanique	5
3.1.1	Retour de force par calcul d'intersections	5
3.1.2	Retour de force par calcul de Force élastique	5
3.1.3	Application à la perception d'un mur à épaisseur finie	6
3.2	Création d'une surface plan avec une normale	7
3.3	Création d'une boule/sphère virtuelle	7
3.4	Création d'une glissière	8
3.5	Intégration des frottements visqueux	10
4	Utilisation avec un simulateur 3D	11
4.1	Intégration de scripts	11
4.2	Interaction avec les objets Blender	11
5	Conclusion	13

Partie 1

Introduction

Ce rapport a pour but de synthétiser le travail effectué dans le cadre des séances de TP en Haptique. Le but premier de ces TP était de réaliser une chaîne de téléopération entre un opérateur utilisant un contrôleur et un environnement de simulation pour y manipuler des objets voire faire de la téléopération de robot. Cette chaîne de téléopération contiendra une interface robotique tout en devant être de la plus grande transparence possible pour garantir une efficacité maximale. Cette transparence implique donc un besoin de faire ressentir le plus de sensations possibles par le retour haptique donné à l'interface. Il faut donc coupler l'interface avec la simulation, ceci est possible par deux méthodes :

- le couplage en admittance : Le robot prend en entrée les forces/couples soumis par l'utilisateur pour en déduire les déplacements avec un moteur physique intégré à l'interface. Cette méthode peut sembler attrayante pour sa grande précision. En pratique, les capteurs de forces et couples sont compliqués à embarquer (volumineux ou chers), elle induit aussi un retard entre les mouvements réalisés et leur réponse ce qui peut causer des instabilités dans la réponse.
- le couplage en impédance : le robot prend en entrée les déplacements et vitesses de l'effecteur et retourne les forces associées à ces déplacements dans l'environnement virtuel. Bien plus simple à mettre en oeuvre matériellement (raison pour laquelle elle est populaire dans l'industrie) perd en précision et donc en réalisme.

L'interface robotique utilisée tout le long des TPs sera un Novint FALCON. cette interface conçue pour le jeu vidéo utilise la méthode du couplage en impédance. Il est cependant nécessaire d'estimer en premier lieu, les paramètres physiques de l'interface, comme la masse de l'effecteur et les raideurs des axes. Puis une chaîne de téléopération sera conçue pour des formes physiques simples. la chaîne de téléopération sera ensuite intégrée dans un simulateur avec un moteur physique complet.



FIGURE 1.1 – Contrôleur Falcon

Partie 2

Identification des paramètres mécaniques du Novint Falcon

2.1 Estimation des capacités mécaniques du robot

Un premier travail à effectuer est d'estimer pour quel ordre de grandeur de forces et déplacements notre interface est viable pour la chaîne de téléopération. L'ensemble des positions atteignables définit l'espace de travail du robot. Ce dernier étant un manipulateur parallèle Delta, l'espace de travail sera donc assez restreint. Avec un script d'enregistrement des coordonnées de l'effecteur, on en déduit un espace de travail d'une forme sphérique centrée autour de l'origine de l'espace du robot. d'où E, espace du travail peut être défini comme suivant : $E(x, y, z) := \{\sqrt{x^2 + y^2 + z^2} < 0.04m\}$.

Il est maintenant nécessaire d'estimer l'intervalle de force pour lequel l'interface permet une fidélité de reproduction de force. En d'autres termes, il faut trouver l'ordre de grandeur pour laquelle les efforts retournés à l'utilisateur seront les mêmes que ceux calculés par la simulation. Un moyen d'estimer le maximum réalisable est de demander au robot de déployer une force élevée. Avec la fonction `dhdSetForce()`, une force de 100 Newtons (équivalent à une charge de 10kg) puis récupérons la mesure de force déployée par le robot grâce à la fonction `dhdGetForce()`. La force maximale déployable pour chaque axe du robot est comprise entre 15 et 20 Newtons.

2.2 Estimation des paramètres dynamiques de l'interface

Les limites des capacités du Falcon connues, il est nécessaire d'identifier les paramètres dynamiques et inertiels de l'effecteur. La méthode est la suivante :

- un ressort virtuel est défini puis appliqué entre l'effecteur et l'origine de l'espace de travail du robot. on applique l'équation suivante pour un axe : $m_{effecteur}\ddot{x} + c\dot{x} + k_{axe}x = 0$
- le terme d'amortissement c est fixé tel que le régime transitoire du système soit sous amorti
- le résultat du retour élastique est affiché pour estimer graphiquement $m_{effecteur}$ en imposant une valeur à k_{axe}

L'estimation de $m_{effecteur}$ se fait par identification des coefficients dynamique d'un système masse ressort classique. Ainsi ω est mesuré et sachant $m_{effecteur} = \frac{k_{impos}}{\omega^2}$, nous en déduisons, en prenant $k_{impos} = 1000N/m$: $m_{effecteur} = 300g$ à partir des mesures $T = 0.33s$, $\omega = 6\pi rad/s$.

Avec c l'amortissement, m la masse de l'objet et k la raideur du ressort. Cela se traduit par l'expression de la pulsation $\omega^2 = \frac{k}{m}$ avec $\omega = 2\pi F$ où F est la fréquence.

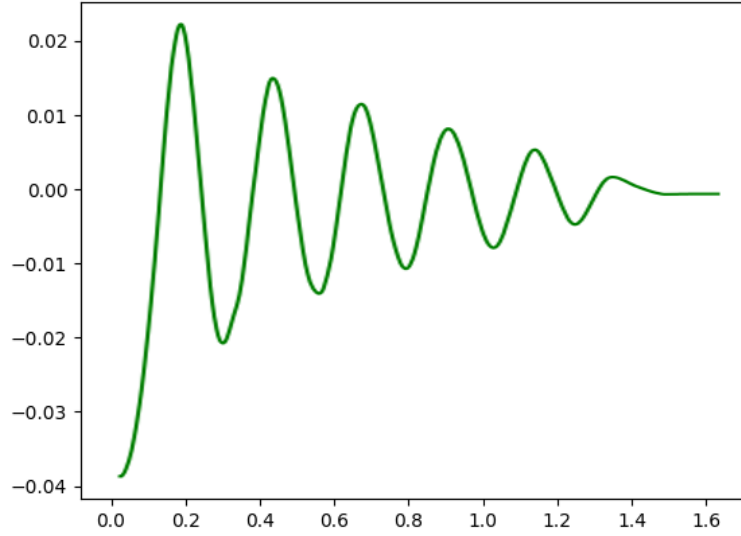


FIGURE 2.1 – déplacement y de l'effecteur dans une simulation de système masse-ressort avec $k = 1000$

Partie 3

Travaux sur les figures virtuelles

L'identification des paramètres de l'interface robotique faite, il est possible de garantir un domaine pour lequel, une chaîne de téléopération sera suffisamment transparente. Cette partie va donc se concentrer sur la création du retour haptique ainsi que de son utilisation pour la simulation de contact sur des formes simples et des effets physiques liés au mouvement dans l'environnement.

3.1 Conception d'un retour de force fidèle au contact mécanique

3.1.1 Retour de force par calcul d'intersections

Afin d'avoir un retour tactile fidèle à la réalité pour l'utilisateur, une démarche généraliste doit être effectuée pour modéliser et transmettre une force de contact à l'interface. Une première approche fut pensée :

- la chaîne de téléopération se faisant par impédance, les positions sont récupérées
- si la position concorde avec une surface définie dans le simulateur, une force F constante est appliquée pour ne pas passer au travers

Cette méthode, bien qu'intuitive n'est pas réaliste et produit un retour surprenant pour l'utilisateur. Il se mettra à ressentir des impulsions fortes et périodiques assimilables à des vibrations comme décrit dans la figure 3.1 dans le cas d'une surface verticale infinie située en $y = -0.02m$:

en représentant l'évolution de cette force de retour dans le temps, cette dernière est discontinue et n'est donc pas adéquat pour la chaîne de téléopération comme visible sur la figure 3.2.

3.1.2 Retour de force par calcul de Force élastique

Le monde réel n'est pas constitué d'objets parfaitement rigides. Chaque matériau possède des propriétés mécaniques de déformation qui lui sont propre. Ces déformations induisent une force de rappel élastique qui vient de manière réciproque au déplacement/déformation induite du contact. Ce phénomène peut donc être reproduit Le procédé de retour de force est donc le suivant :

- la distance entre la surface et l'effecteur est calculée
- si l'effecteur "rentre" dans la surface, alors une force de rappel est calculée

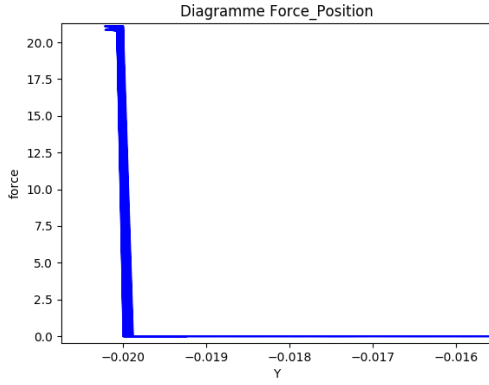


FIGURE 3.1 – Mur selon l'axe \vec{y}

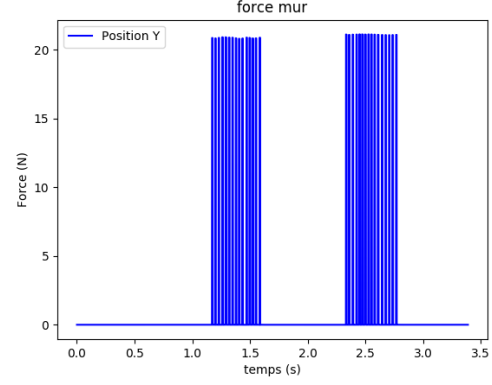


FIGURE 3.2 – évolution de la force de retour dans le temps

— la force de rappel est assimilable à celle de l'élongation d'un ressort : $F_{\text{retour}} = k\Delta\vec{d}$

Cette approche est alors utilisée pour produire un retour de force sur un mur identique à celui testé sur la figure 3.1. Le résultat obtenu en 3.3 montre alors que pour une raideur suffisamment élevée, le diagramme force-position s'avère proche du ressenti réel. Le retour de contact par force élastique est donc éprouvé et nous servira pour toute la suite de ce rapport.

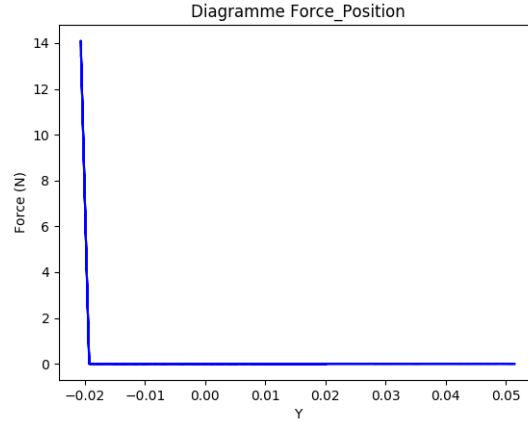


FIGURE 3.3 – Mur selon l'axe \vec{y} avec retour élastique adéquat

3.1.3 Application à la perception d'un mur à épaisseur finie

Un retour de force sur une surface plan a été produite et éprouvé, il est alors judicieux de simuler la présence d'un mur à épaisseur finie et de le faire ressentir à l'utilisateur.

La procédure utilisée est la suivante dans le cas d'un mur sur l'axe y :

- le mur est défini par son centre x_0 et son épaisseur l
- si $x_{\text{effecteur}}$ est compris dans l'intervalle $[x_0 - \frac{l}{2}, x_0 + \frac{l}{2}]$, un retour élastique est appliqué
- le sens du retour élastique est donné alors par le signe de la différence $x_0 - x_{\text{effecteur}}$

Cette procédure peut être adaptée pour des murs similaires sur les 3 axes de l'espace 3D. La figure 4.1 est le résultat en force-position pour un mur à épaisseur de 0.02 m centrée en $y = 0$:

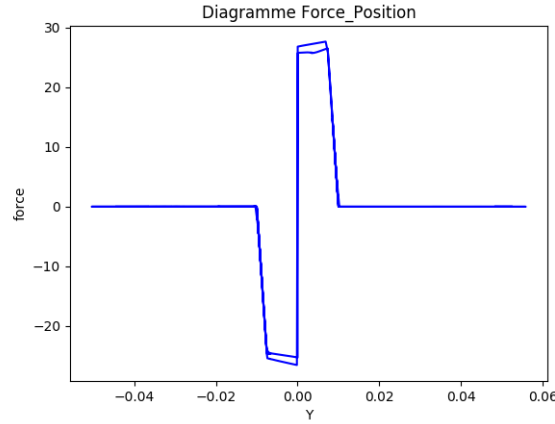


FIGURE 3.4 – diagramme force-position d'un mur à épaisseur finie

Une amélioration possible de ce mur fini est d'ajouter des informations supplémentaires pour le délimiter dans l'espace avec des sommets. Malheureusement, cette tentative fut infructueuse car un cas limite de dépassement du mur fut trouvé. Si l'effecteur se trouvait sur une arête, il ne savait pas quelle direction prendre ce qui avait pour effet de perturber le retour haptique.

3.2 Création d'une surface plan avec une normale

L'approche développée jusqu'ici pour créer des surfaces plan était de prendre ceux ci affines aux axes principaux de l'espace 3D. Dans le cas d'une génération de surface plane à orientation quelconque, un moyen efficace est de définir ce plan par un vecteur normal et un point appartenant au plan. La méthode de détection et de calcul de retour de force diffère légèrement :

- la normale \vec{n} et un point M appartenant au plan sont connus
- avec P, position de l'effecteur, \vec{MP} est déduit
- si $\vec{MP} \cdot \vec{n} = 0$, alors P appartient au plan et une force de retour élastique est appliquée

Cette méthode permet donc une superposition plus simple de plusieurs plans aux paramètres différents dans un seul même espace 3D. Elle généralise la création de tout murs infinis inclinés ou non visible sur la figure 3.5. Une limitation de l'air peut être faite par spécification des sommets. Mais cette fonctionnalité ne fut pas ajoutée pour la même raison que les plans délimités à épaisseurs finies : il y a conflit sur les arêtes du plans lors du choix de la normale.

3.3 Création d'une boule/sphère virtuelle

La création d'objets plan maîtrisée, il est nécessaire de pouvoir appliquer le retour de contact élastique à des volumes comme une sphère. Une boule/sphère physique dans un espace 3D est alors définie par :

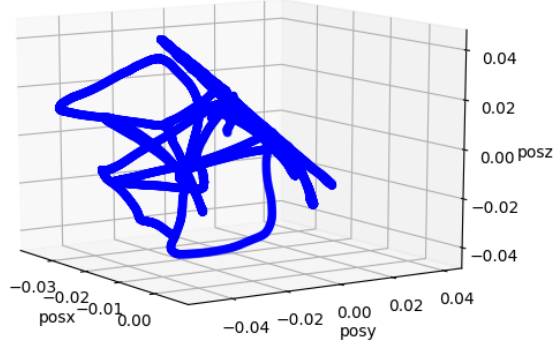


FIGURE 3.5 – visualisation d’une surface plan inclinée dans l’espace

- la position de son centre C dans le repère cartésien
- son rayon R
- sa rigidité de surface k

La sphère étant un objet fini dans l’espace, les conditions de contacts sont plus compliquées à formuler qu’un mur infini comme précédemment. Le processus est alors le suivant :

- prendre la position de l’effecteur \vec{OP} avec O origine de l’espace 3D
- calculer le vecteur de différence de position entre l’effecteur et le centre de la sphère \vec{CP}
- si $\|\vec{CP}\| < R$, alors une force de retour doit être calculée telle que $\vec{F} = -k(\|\vec{CP}\|)\vec{e}_r$

La subtilité réside dans le calcul du vecteur normal au point de contact \vec{e}_r . Celui-ci n’est autre que le vecteur unitaire sur \vec{CP} d’où $\vec{e}_r = \frac{\vec{CP}}{\|\vec{CP}\|}$

Cette approche analytique permet un ajout de plusieurs sphères distinctes dans l’espace virtuel de simulation, voire de les superposer partiellement comme montré à la figure 3.6 :

3.4 Création d’une glissière

Dans cette partie, la création et le retour de force pour simuler une liaison glissière seront traités. Une glissière est caractérisée par un nombre de contraintes de liaisons assez élevé. Dans le cadre de l’interface Falcon qui fonctionne uniquement en translation de l’effecteur, la glissière contraint le mouvement sur la droite directrice de la glissière. Il faut donc appliquer une force élastique normale à la droite comme suivant :

$$\vec{F} = -k(\vec{OP}' - \vec{OP})$$

Où P' est le projeté de P sur la droite, O point de référence sur la droite directrice définie et k la raideur nécessaire pour simuler l’interaction de contact élastique. Connaissant \vec{n} , vecteur normal à la droite, le projeté est alors calculé de la sorte :

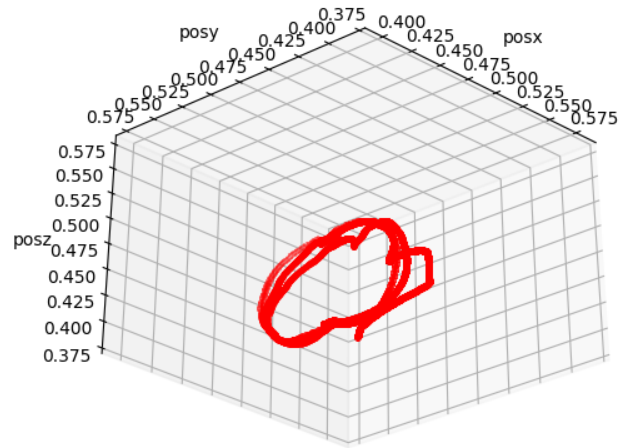


FIGURE 3.6 – Mouvement de l'effecteur autour de 2 sphères

$$\vec{OP'} = (\vec{OP} \cdot \vec{n}) \vec{n}$$

Avec cette méthode, il est alors possible d'avoir un plus large choix d'objet directeur, il est de fait possible d'établir une courbe directrice si la normale est connue en tout point de la courbe. Le calcul de la force serait alors réadapté dans un repère de Serret-Frenet comme suivant :

- la position de l'effecteur est enregistrée
- le point P' le plus proche appartenant à la courbe est recherché
- le calcul de la force de retour est similaire pour p' trouvé

De fait, il est possible d'ajouter des crans voire de simuler une liaison glissière hélicoïdale par cette méthode.

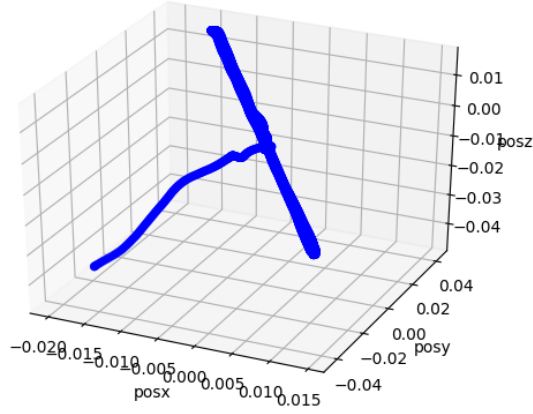


FIGURE 3.7 – Représentation des positions contraintes avec une glissière linéaire sur une droite quelconque

3.5 Intégration des frottements visqueux

Tout objet évoluant dans un fluide (gaz ou liquide) est soumis à une action non conservatrice liée à l'interaction de sa surface avec le fluide. Cette interaction induit un frottement en opposition au mouvement du solide et est modélisé sous plusieurs façons :

- $F_{frot}^{\vec{v}} = -c v_{objet}$ dans le cas de vitesses faibles (écoulement laminaire)
- $F_{frot}^{\vec{v}} = -c ||v_{objet}||^2 \vec{e}_v$ dans le cas de vitesses élevée (ou nombre Re fort/turbulent) et calculant au préalable $\vec{e}_v = \frac{v_{objet}}{||v_{objet}||}$, vecteur unitaire suivant le vecteur vitesse de l'objet.

Un tel effet serait judicieux à implémenter pour garantir la transparence de la chaîne de téléopération ou amplifier les sensations de résistance des fluides pour certaines tâches de manipulation. Devant la simplicité du cas laminaire, ce dernier fut implémenté grâce à la fonction `dhdGetLinearVelocity` pour récupérer les informations de vitesse de l'effecteur.

Cet effet physique étant distinct des interactions de contact entre solides, il peut être utilisé systématiquement comme force intrinsèque appliquée à l'effecteur, en spécifiant le coefficient de viscosité du milieu de simulation.

Partie 4

Utilisation avec un simulateur 3D

Il est aisé de définir et de déterminer analytiquement les retours de force sur des objets simples. Dans le cas d'une téléopération sur un robot réel, ce dernier est souvent amené à interagir avec des objets à la géométrie complexe, l'approche de simulation via un simple script python n'est plus approprié d'où l'utilisation d'un outil de simulation tel que Blender, plus complet et prenant en charge la dynamique des objets complexes.

4.1 Intégration de scripts

Un simulateur fonctionne de la façon suivante :

- L'environnement de simulation est visible par une caméra fixant une scène
- Dans cette scène 3D sont présents des objets. Ces derniers sont définis par leur géométrie, leur couleur mais également leurs paramètres dynamiques tels que leur masse et leur matrice d'inertie
- Ces objets peuvent être liés par des scripts leur imposant des forces, des déplacements ou des interactions entre objets
- Les scripts développés précédemment peuvent donc être repris et réadaptés pour l'environnement Blender

Blender effectuant une boucle interne de simulation, il faut donc retirer les boucles de fonctionnement des scripts python et ne garder que les fonction de calcul de retour de force. De plus, les échelles de déplacement entre la scènes et le Novint étant différentes (rapport de déplacement de 0.1 entre le Novint et Blender), un facteur de correction d'échelle est appliqué sur les déplacements de l'effecteur dans le monde virtuel.

Cependant, l'import de scripts est une approche peu efficace, elle nécessite une création d'un nombre important de fonctions pour prendre en charge le plus de formes et d'effets possible. Une autre méthode consiste à utiliser les propriétés des objets Blender comme cela est décrit en partie suivante.

4.2 Interaction avec les objets Blender

La simulation et le retour mécanique appréhendés avec des objets simples, il est judicieux d'inclure ces objets dans un environnement plus complexe avec un moteur physique fidèle et éprouvé comme Blender.

Cet environnement de simulation physique permet dans une scène, une simulation d'interaction de forces entre objets et champs de forces comme la gravité. néanmoins, notre effecteur n'étant pas un objet considéré comme physique sous Blender, il faut définir un objet fantôme suivant la position de l'effecteur (qui est exploitable par Blender) et pouvant faire office d'effecteur dans la scène virtuelle.

Pour cela, nous allons créer un corps rigide simple voire ponctuel "proxy" qui sera lié à l'effecteur en récupérant les positions de l'effecteur et liant ces dernières au proxy par un système de masse-ressort couplés et amortis. cet objet suivra les mouvements de l'effecteur avec un certain retard de phase dû à l'amortissement nécessaire pour garantir la stabilité de suivi. Le proxy étant un objet défini dans Blender, il pourra interagir avec les autres objets à la place de ce dernier et faire les retours de forces adéquats à l'interface. Le système masse-ressort créé appliquera une force attractive sur proxy ainsi qu'une force de réaction à l'effecteur pour garantir le retour haptique avec les équations suivantes :

$$\begin{cases} \vec{F}_{proxy} &= k(x_{effecteur} - x_{proxy}) - c\vec{v}_{effecteur} \\ \vec{F}_{effecteur} &= -\vec{F}_{proxy} \end{cases}$$

Avec k la raideur fixée autour de 200, Δ_{pos} la différence de position entre le proxy et l'effecteur du robot, c l'amortissement et v la vitesse du proxy.

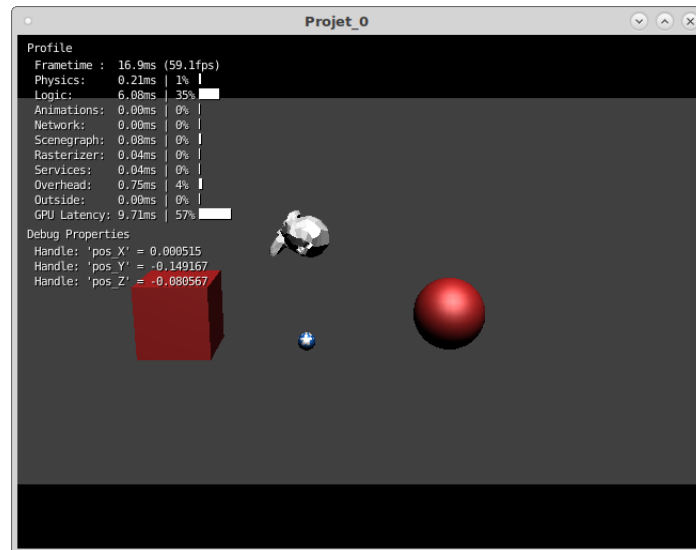


FIGURE 4.1 – scène Blender utilisée avec le proxy en bleu confondu avec la position de l'effecteur

Un constat est à faire au vu de la perception de l'utilisateur. Le proxy étant l'objet en réelle interaction et non plus l'effecteur, en dépit de l'optimisation de l'amortissement vers un régime critique une latence entre l'effecteur et le proxy subsiste. Ce retard peut perturber l'utilisateur lors de la visualisation et manipulation simultanée. Une possibilité pour diminuer cette gêne serait de ne plus afficher la position de l'effecteur, ainsi que de compenser l'inertie de celui ci afin de réduire grandement le retard de phase du proxy.

Partie 5

Conclusion

Malgré ses capacités de transparence limitées, l'interface Falcon a permis de nous sensibiliser aux méthodes sous-jacentes présentes dans les dispositifs de téléopération par retour haptique.

Les travaux effectués sous python illustrent les techniques utilisées sous des environnements plus complexes comme Blender. Les objets à ressentir sont de fait définis par une "collide-box" et un calcul de force de retour est appliqué par modélisation de contact identique au monde réel.

Il aurait été intéressant de poursuivre les travaux sous Blender afin d'augmenter la transparence de l'interface par ajout d'autres effets complexes (frottements secs, état de surface, inertie..) plus simples à émuler sous Blender quand celui-ci s'occupe déjà de la modélisation physique d'effets de base (contacts mécaniques, gravité...).