

N° d'ordre: 653

THÈSE

École Doctorale Mathématiques, Sciences de l'Information et de l'Ingénieur

UdS

présentée pour obtenir le grade de

Docteur de l'Université de Strasbourg
Discipline Informatique
Spécialité Informatique

par

Manuel Veit

Amélioration du contrôle de l'utilisateur dans des applications de modélisation géométrique en environnement de réalité virtuelle

Soutenue publiquement le **20 Septembre 2010** devant la commission d'examen composée de :

Pr. Dominique Bechmann *Directeur de thèse*
Professeure à l'Université de Strasbourg

Dr. Antonio Capobianco *Encadrant et Examineur*
Maître de conférences à l'Université de Strasbourg

Pr. Pascal Guitton *Rapporteur*
Professeur à l'Université de Bordeaux 1

Dr. Anatole Lécuyer *Rapporteur*
Chargé de Recherche à l'INRIA de Rennes

Dr. Daniel Mestre *Examineur et Président du Jury*
Directeur de Recherche au CNRS, Institut des Sciences du Mouvement

**Laboratoire des Sciences de l'Image, de
l'Informatique et de la Télédétection**

UMR 7005 – CNRS

Remerciements

Tout d’abord, je tiens à remercier Dominique Bechmann pour avoir accepté d’encadrer cette thèse tout au long de ces trois années. Son expérience et ses conseils avisés m’ont été d’une grande aide. Je remercie également Pascal Guitton, Anatole Lécuyer et Daniel Mestre pour avoir accepté d’évaluer ce travail.

Je tiens à remercier tout particulièrement Antonio Capobianco (également connu sous le nom de « Grand Maître »). Malgré ses responsabilités, il a toujours su se rendre disponible. Nos nombreuses discussions (dans ou hors des locaux du laboratoire) m’ont permis de mener à bien ce travail. À son contact, j’ai beaucoup appris.

Je remercie également l’ensemble de l’équipe d’Informatique Géométrique et Graphique, pour la bonne humeur omniprésente mais également pour les nombreuses discussions, plus ou moins sérieuses, mais toujours enrichissantes. Je remercie tout particulièrement les actuels occupants du bureau de la fête (Lucas et Simon) pour les moments de détente et de franche rigolade. Je remercie également, Thomas Jund (plus connu sous le nom de Never Wrong Man), avec qui j’ai partagé le bureau, toujours dans la joie et la bonne humeur. Pour eux trois, je ne dirai qu’une chose : « Foi et Courage ».

Je tiens également à remercier ma famille pour m’avoir soutenu et pour m’avoir encouragé durant ces nombreuses années d’études.

Je remercie également toute ma belle famille, qui par un dimanche ensoleillé, a accepté et eu le courage de relire intégralement cette thèse.

Mon dernier remerciement va à celle qui deviendra très prochainement ma femme. Son soutien sans failles et sa présence, tout au long de ces trois années, m’ont permis de surmonter les nombreux moments de doute. Je la remercie également pour sa contribution à la rédaction de ce mémoire.

Table des matières

I	Interaction 3-D : théories et applications	5
1	Interaction et modélisation géométrique	7
1.1	Introduction	8
1.2	Applications de modélisation en réalité virtuelle	10
1.2.1	Interaction libre	10
1.2.2	Interaction contrainte	14
1.2.3	Bilan	19
1.3	Manipulation en réalité virtuelle	20
1.3.1	Techniques existantes	20
1.4	Analyse	25
1.4.1	Une manipulation plus directe	25
1.4.2	Contraindre la manipulation	26
1.4.3	Interfaces tactiles et interaction 3-D	27
1.5	Approche proposée	28
1.6	Conclusion	30
2	Contexte théorique	31
2.1	Modèles connexes	32
2.1.1	Degrés de liberté et dimensions	32

2.1.2	<i>Control - Display Space Mapping</i>	33
2.1.3	Structure perceptuelle	35
2.2	Interaction et degrés de liberté	38
2.2.1	Technique d'interaction	39
2.2.2	Caractérisation de la tâche	43
2.2.3	Conclusion	46
3	Une nouvelle métrique de coordination	49
3.1	Mesures existantes	50
3.1.1	Mesure d'inefficacité	51
3.1.2	$\mathcal{M}$ – metric	53
3.2	NDC : Nombre de Degrés de liberté Combinés	54
3.2.1	Exemple : cas de l'orientation 3-D	55
3.2.2	Formulation générale pour n DDL	58
3.2.3	Implémentation	60
3.2.4	Comparaison	60
3.2.5	Validation	61
3.3	Conclusion	63
II	Expérimentations et résultats	65
4	Décomposition et orientation 3-D	69
4.1	Métaphore du potier	70
4.1.1	Mise en oeuvre	70
4.1.2	Fonctionnement	71
4.2	Protocole expérimental	72
4.2.1	Matériel	72
4.2.2	Techniques comparées	74
4.2.3	Procédure expérimentale	76

4.2.4	Mesures	78
4.2.5	Hypothèses	78
4.3	Résultats	79
4.3.1	Performances	79
4.3.2	Manipulation des degrés de liberté	80
4.4	Analyse et discussions	81
4.4.1	Performances	81
4.4.2	Coordination des degrés de liberté	83
4.5	Conclusion	84
5	Décomposition et positionnement 3-D	87
5.1	Introduction	88
5.2	Protocole expérimental	89
5.2.1	Techniques comparées	89
5.2.2	Tâches	92
5.2.3	Procédure expérimentale	93
5.2.4	Mesures	94
5.2.5	Hypothèses	94
5.3	Résultats	95
5.3.1	Performances	95
5.3.2	Analyse comportementale	97
5.3.3	Évaluation subjective	100
5.4	Discussions	102
5.4.1	Décomposition en fin de tâche	102
5.4.2	Évaluation subjective	103
5.4.3	Précision	104
5.5	Conclusion	108
6	Contrainte et tâche à contrôle continu	109
6.1	Introduction	110

6.2	Protocole expérimental	111
6.2.1	Techniques comparées	111
6.2.2	Tâche	112
6.2.3	Procédure expérimentale	113
6.2.4	Mesures	114
6.2.5	Hypothèses	114
6.3	Résultats	115
6.3.1	Erreurs	115
6.3.2	Temps de réalisation	116
6.3.3	Évaluation subjective	116
6.4	Analyse	117
6.4.1	Performances	117
6.4.2	Confort d'utilisation	119
6.5	Conclusion	120
7	Recommandations de conception pour l'interaction 3-D	123
7.1	Bilan	123
7.2	Recommandations pour des tâches d'interaction 3-D	125
7.3	Exploitations des recommandations	125
III	Amélioration du contrôle de l'utilisateur	129
8	DIOD	131
8.1	Mise en oeuvre	132
8.2	Évaluation de DIOD	135
8.2.1	Techniques comparées	136
8.2.2	Protocole expérimental	137
8.3	Résultats	140
8.3.1	Positionnement de précision	140

8.3.2	Positionnement répété	143
8.3.3	Évaluation subjective	146
8.4	Analyse	147
8.4.1	Stratégie	147
8.4.2	Performances	148
8.4.3	Évaluation subjective	150
8.5	Conclusions	150
9	CrOS : Cursor On Surface	153
9.1	Introduction	154
9.2	Contexte	156
9.2.1	Méthodes existantes	156
9.2.2	Bilan	159
9.3	CrOS	159
9.3.1	Principe de la méthode	159
9.3.2	Compatibilité directionnelle	161
9.3.3	Extensions	161
9.4	Conclusions	164
10	Application	167
10.1	Applications de modélisation en réalité virtuelle	168
10.1.1	VRModelis	168
10.1.2	VR Terrain Editor	172
10.1.3	Conclusion	177
10.2	Intégration des techniques d'interaction	177
11	Conclusion	179
11.1	Bilan	179
11.2	Perspectives	182
11.2.1	Repère et construction mentale de la tâche	182

11.2.2	Interaction hybride par l'utilisation d'un terminal mobile tactile	182
11.2.3	Information visuelle	183
A	Implémentation de CrOS	197
A.1	Définition du plan géodésique	198
A.2	Calcul de l'intersection	198
B	Questionnaire suivi de trajectoires	201
C	Questionnaire DIOD	205

Table des figures

1	Exemple de manipulation en environnement de réalité virtuelle	1
1.1	Exemple de déformation d'un objet	9
1.2	Exemple d'un réseau de courbes dans un espace 3-D	11
1.3	Schéma de subdivision d'une face carré	12
1.4	Un exemple de création d'une surface à l'aide d'un pinceau surfacique .	13
1.5	Un exemple de construction avec VLEGO	13
1.6	Illustration du <i>clay-modelling</i>	14
1.7	Exemple d'un réseau de courbes 3-D	15
1.8	Création d'un cylindre dans l'application développée par Stork <i>et al.</i> . .	16
1.9	Édition des différents paramètres d'un cylindre	18
1.10	Résultat obtenu avec l'application InTouch	18
1.11	Métaphore des baguettes chinoises	22
1.12	Le CAT	22
1.13	Le périphérique DigiHaptic	23
1.14	La technique du <i>Balloon Selection</i>	24
1.15	La technique <i>Multi-View</i>	28
1.16	La technique <i>Magnitude Filtering</i>	29
2.1	Le <i>3-state model</i>	32

2.2	Les trois DDL de la souris	33
2.3	Une illustration de l'effet miroir	35
2.4	La structure perceptuelle	36
2.5	Exemple d'une structure perceptuelle intégrable et d'une structure perceptuelle séparable	37
2.6	La relation entre degrés de liberté des périphériques et degrés de liberté de la tâche	40
2.7	La technique du Go-Go	42
2.8	Synthèse du modèle théorique	47
3.1	La mesure <i>Target Re-entry</i> durant une sélection 2-D	50
3.2	La mesure d'inefficacité	51
3.3	La mesure d'inefficacité et le chemin optimal	52
3.4	La coordination des DDL dans une tâche de positionnement 2-D	54
3.5	La décomposition et l'intégration des DDL	56
3.6	La mesure MDS et ses valeurs clés	57
3.7	Les trajectoires de test	61
4.1	La mise en oeuvre proposée pour la simulation d'une interface tactile	71
4.2	La métaphore du potier	73
4.3	Un exemple de manipulation sur le workbench	74
4.4	Les gants de données	75
4.5	La technique de rotation avec intermédiaire	76
4.6	La tâche d'orientation 3-D	77
5.1	La technique de manipulation directe avec séparation de la profondeur	90
5.2	La technique de manipulation directe avec séparation de la hauteur	91
5.3	La technique de manipulation tactile avec séparation de la profondeur	91
5.4	La technique de manipulation tactile avec séparation de la hauteur	92
5.5	La tâche de positionnement de précision	93

6.1	Contrainte virtuelle et physique pour le suivi de trajectoires	112
6.2	Les trois trajectoires utilisées lors de la troisième étude	113
8.1	Effet du seuillage basé sur la vitesse minimale	134
8.2	Exemple de transition entre la phase de manipulation libre et la phase de manipulation contrainte	134
8.3	La représentation graphique du composant GearBox	135
8.4	La tâche de positionnement répété	138
8.5	La tâche de positionnement de précision	139
8.6	Un exemple de courbe 3-D	151
9.1	Contrainte virtuelle pour la manipulation sur des surfaces	154
9.2	Transformation de déplacements dans un plan en déplacements géodé- siques sur une surface 3-D	155
9.3	Navigation dans des scènes 3-D à l'aide d'une souris	157
9.4	Illustration d'une surface dépliée et son utilisation pour la navigation . .	158
9.5	Un exemple de situation où les mouvements sur la surface sont inversés par rapport aux mouvements du périphérique	158
9.6	Le déplacement sur la surface est défini par l'intersection entre l'objet et un plan	160
9.7	Exemple d'incompatibilité directionnelle entre le déplacement effectué par l'utilisateur et le déplacement du curseur	161
9.8	Mise en oeuvre du retour pseudo-haptique	163
9.9	Courbure et retour pseudo-haptique	163
9.10	Calcul de la courbure en utilisant une moyenne pondérée	164
10.1	Création d'une forme 3-D en utilisant une extrusion	169
10.2	Un exemple de déformation directe basée sur l'utilisation de contraintes	171
10.3	Un exemple de déformation colorée	171
10.4	Coloration et placage de texture sur une surface	172
10.5	Exemple de l'extrusion d'une face	173
10.6	Exemple d'une succession d'extrusions	173

10.7	La structure de données utilisée pour représenter le terrain	174
10.8	Dispersion et érosion du roc sur une partie du terrain	175
10.9	Création d’une arche sur le terrain	176
10.10	Ajout d’arbres dans une scène naturelle	176
A.1	Intersection entre le plan géodésique et la face courante	199

Introduction

En exploitant les récentes avancées technologiques, la réalité virtuelle (notée RV) permet d'immerger les utilisateurs dans des environnements 3-D. Ces environnements, artificiellement créés, peuvent être inspirés du monde réel ou totalement imaginaires. Nous illustrons, en Figure 1, un utilisateur immergé dans une scène 3-D simple constituée d'un seul objet virtuel. Associés à des périphériques d'interaction, tels que les gants de données ou le Wand, ce type d'environnement permet une manipulation directe et intuitive [Shneiderman 87].

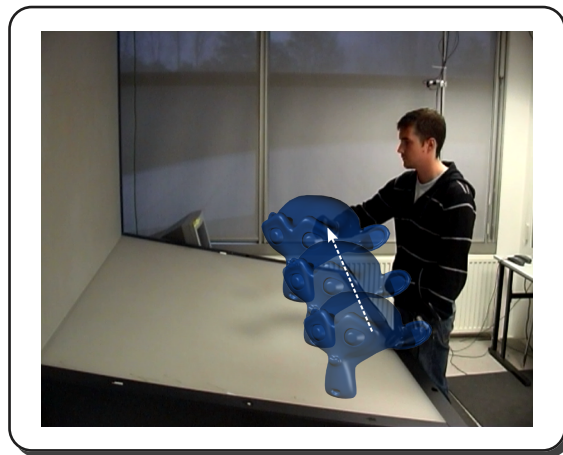


FIGURE 1 – Sur le Workbench, l'utilisateur est ici immergé dans un environnement 3-D dans lequel il peut manipuler directement les objets à l'aide de gants de données.

Par rapport à des stations de travail classiques (constituées d'un écran standard, d'un clavier et d'une souris), l'utilisateur bénéficie d'une visualisation et d'une interaction tridimensionnelle. Dans ce nouveau contexte, la perception et l'interaction sont enrichies par l'ajout d'une troisième dimension, la **profondeur**. Cet enrichissement a sou-

vent été présenté comme un avancement majeur vers la mise en oeuvre d'interfaces plus intuitives et plus efficaces. En effet, l'interaction 3-D semble particulièrement adaptée aux tâches de nature 3-D. Prenons l'exemple du positionnement 3-D dans une application de modélisation géométrique. Dans ce type d'application, l'utilisateur manipule habituellement dans quatre vues distinctes : trois vues pour chacune des projections et une vue perspective. Pour positionner l'objet dans l'espace 3-D, l'utilisateur manipule successivement dans les trois différentes vues projetées. En utilisant une technique de manipulation directe avec un périphérique à trois ou six degrés de liberté (DDL), le positionnement peut alors être effectué directement dans l'espace 3-D. Cela a pour avantage, non seulement d'offrir une manipulation intuitive, mais également de permettre une manipulation simultanée de l'ensemble des dimensions de l'espace.

Ces technologies sont aujourd'hui exploitées dans le milieu industriel, notamment dans le milieu automobile. Les environnements de RV sont utilisés pour visualiser à échelle réelle les prototypes des voitures en cours de conception. La visualisation à échelle réelle permet une économie substantielle durant la phase de conception en réduisant les coûts liés à la réalisation de prototypes physiques. Toutefois, lors de la visualisation d'un prototype en milieu immersif ou semi-immersif¹, il n'est pas possible de modifier la géométrie de la carrosserie dans l'environnement virtuel. En effet, une grande partie des phases de modélisation sont, jusqu'à présent, effectuées sur stations de travail classiques, hors de l'environnement virtuel. L'intégration des phases de modélisation et des phases de visualisation permettrait de réduire les temps de réalisation des prototypes.

Toutefois, il semblerait que la perception et l'interaction en environnement 3-D restent difficiles à comprendre. De nombreux facteurs, tels l'utilisation d'ombres projetées, la présence de phénomènes d'occlusion entre les différents objets de la scène ou encore l'utilisation de transparence [Interrante 07], semblent influencer la perception, et par conséquent, les performances, des utilisateurs [Ryu 05] [Interrante 06]. Du point de vue de l'interaction, de nombreuses techniques proposées jusqu'à présent utilisent des gestes libres de toutes contraintes. Cette nouvelle liberté a été initialement perçue comme un facteur pouvant améliorer significativement les performances des utilisateurs [Chen 88]. En permettant la manipulation simultanée des différents DDL de la tâche, l'utilisation de l'interaction permet généralement de réduire les temps de réalisation. Toutefois, l'ajout de la profondeur dans l'espace de manipulation semble introduire de nouvelles difficultés [Fiorentino 03].

L'objectif de notre travail est de proposer des outils d'interaction permettant de pallier à certaines des difficultés liées à la manipulation en environnement 3-D. Ces outils

1. Par immersion, nous entendons ici immersion visuelle. Un environnement immersif, comme un CAVE, dans lequel l'utilisateur est totalement immergé dans une scène 3-D. Dans un environnement semi-immersif, comme le Workbench, l'immersion n'est que partielle.

doivent permettre aux utilisateurs de modéliser des objets 3-D en environnement de RV. Notre travail s'inscrit dans la mise en place d'une application de modélisation géométrique en environnement virtuel. Dans ce cadre applicatif, l'utilisation d'interaction 3-D semble être en adéquation avec la nature tridimensionnelle des tâches impliquées (positionnement dans un espace 3-D, orientation, déformation d'un objet, sculpture, *etc.*). Néanmoins, de telles applications requièrent un niveau de précision élevé, que les techniques actuelles ne semblent pas à même de fournir.

Dans un premier temps, nous identifions les possibles difficultés introduites par l'interaction 3-D. Pour cela, nous proposons un modèle théorique permettant de mieux comprendre le rôle de la technique d'interaction durant la manipulation en environnement de RV. Par la suite, nous proposons et évaluons des solutions adaptées à notre environnement de travail. En particulier, nous proposons de passer par la simulation d'une interface tactile² pour interagir avec l'environnement 3-D :

- Elle introduit un **support physique** au cours de la manipulation. Le support physique peut influencer les performances en donnant à l'utilisateur un meilleur contrôle du mouvement. De plus, il peut améliorer le confort de manipulation en réduisant la fatigue lors de longues sessions de manipulation ;
- Elle permet une **réduction de la dimension de la tâche** à effectuer. Par exemple, en utilisant une interface tactile, le positionnement d'un objet 3-D peut se décomposer en trois sous-tâches, chacune correspondant au positionnement de l'objet selon un des axes du repère du monde virtuel. Cette décomposition peut permettre de mieux appréhender des difficultés liées à la manipulation 3-D.

Dans une série d'études expérimentales, nous souhaitons évaluer les apports respectifs de l'introduction d'un support physique et de la décomposition induite par ce support. Nous utilisons les résultats obtenus afin de proposer de nouvelles techniques d'interaction adaptées aux difficultés de manipulation en environnement de RV.

Dans la première partie, nous présentons, dans un premier temps, un état de l'art concernant les applications de modélisation géométrique en environnement de RV ainsi qu'un état de l'art de l'interaction 3-D. Nous présentons, par la suite, le modèle théorique que nous proposons afin de caractériser le lien entre le périphérique d'interaction, la technique d'interaction et la tâche à réaliser. Nous proposons ensuite une nouvelle métrique permettant de quantifier la coordination d'un geste dans l'espace de manipulation. Cette métrique nous permet d'étudier les stratégies de réalisation basées sur l'intégration et la décomposition des DDL.

Dans la seconde partie, nous utilisons cette métrique dans une série de trois expérimentations :

- La première expérimentation concerne la tâche d'orientation 3-D

2. Une interface tactile est un dispositif qui va permettre d'interagir avec une Machine en permettant une manipulation par simple toucher de ce dispositif avec les doigts.

- La seconde expérimentation concerne la tâche de positionnement 3-D
- La dernière expérimentation concerne la tâche de suivi de trajectoires

Ces études expérimentales ont pour objectif de comprendre l'influence de la décomposition et du support physique sur les performances des utilisateurs. Après un bilan des résultats obtenus, nous proposons un ensemble de guides de conception.

Dans la dernière partie, nous utilisons ces guides de conception afin de proposer deux nouvelles techniques d'interaction. La première, appelée *Decomposition and Integration Of Degrees of freedom*, est une technique de positionnement 3-D. La seconde, appelée *Cursor On Surface*, est une technique de suivi de trajectoires sur la surface d'un objet 3-D. Nous décrivons également la manière dont ces techniques sont intégrées dans deux applications de modélisation en environnement de RV. Nous finirons par quelques conclusions et perspectives que nous proposons dans le cadre de ce travail.

Première partie

Interaction 3-D : théories et applications

État de l'art : interaction et modélisation géométrique

Sommaire

1.1	Introduction	8
1.2	Applications de modélisation en réalité virtuelle	10
1.2.1	Interaction libre	10
1.2.2	Interaction contrainte	14
1.2.3	Bilan	19
1.3	Manipulation en réalité virtuelle	20
1.3.1	Techniques existantes	20
1.4	Analyse	25
1.4.1	Une manipulation plus directe	25
1.4.2	Contraindre la manipulation	26
1.4.3	Interfaces tactiles et interaction 3-D	27
1.5	Approche proposée	28
1.6	Conclusion	30

Nous présentons dans ce chapitre un état de l'art de l'interaction en réalité virtuelle (notée RV). Nous nous intéressons, dans un premier temps, aux techniques d'interaction liées à la modélisation géométrique, le contexte applicatif dans lequel se place notre travail. Nous nous intéressons, par la suite, aux techniques d'interaction proposées dans le cadre de tâches plus fondamentales comme la sélection ou la manipulation en environnement de RV [Bowman 98]. Dans la suite de ce document, nous ne nous intéressons pas aux techniques d'interaction liées aux tâches de navigation ou aux tâches de contrôle d'application.

1.1 Introduction

Afin d'identifier l'apport potentiel de la RV dans le contexte de la modélisation, nous utilisons la définition de la modélisation géométrique donnée par le Groupe de Travail en Modélisation Géométrique de l'AFIG (Association Française d'Informatique Graphique).

Définition : *La modélisation géométrique est l'ensemble des outils mathématiques, numériques et informatiques qui combinés permettent de construire un modèle virtuel (ou modèle informatique) d'un objet réel. Cet objet peut être plus ou moins complexe, plus ou moins schématisé. Il peut être le fruit de l'imagination, d'une tendance ou plutôt une solution plus ou moins exacte d'un problème physique donné, voire même un compromis entre les deux.*

La modélisation géométrique sous-entend d'être en mesure de réaliser :

- *La construction de formes élémentaires ;*
- *L'assemblage de formes élémentaires pour créer des objets de plus en plus complexes en respectant des contraintes topologiques ;*
- *Des manipulations géométriques pour représenter, modifier, analyser ;*
 - *Processus « élémentaires » (transformations géométriques, calcul, ...),*
 - *Processus complexes (intersection),*
 - *Processus spécifiques (objets décalés (fabrication), raccordements et congés).*

De cette définition, nous retenons que la modélisation géométrique doit permettre de construire des modèles d'objets issus du monde réel ou de l'imagination du créateur. Pour des modèles simples, la construction peut être directe. Pour des modèles plus complexes, la construction peut être effectuée notamment par assemblage de modèles plus simples. Ces modèles, une fois construits, doivent être par la suite modifiables (déformation, modification topologique, etc.).

Généralement, les outils de modélisation géométrique sont utilisés afin de créer / modifier des objets 3-D et nécessitent donc de pouvoir manipuler dans un espace 3-D. L'utilisation de tels outils se fait habituellement sur stations de travail classiques, à l'aide d'un clavier et d'une souris. La souris est un périphérique à deux degrés de liberté (noté DDL), avec lequel l'utilisateur doit effectuer des tâches allant de trois (positionnement simple) à six DDL (comme par exemple le *docking*¹). Ce problème est résolu en proposant à l'utilisateur un système de visualisation à quatre vues : une vue perspective et trois vues proposant, chacune, une projection différente. La vue perspective sert essentiellement à visualiser l'objet. Les trois vues restantes sont quant à elles principalement utilisées lors des phases de modélisation. Le choix de la vue la plus appropriée à la tâche à effectuer ainsi que la nécessité de passer d'une vue à l'autre rend la modélisation 3-D

1. La tâche de *docking* est une tâche d'assemblage consistant à positionner et orienter un objet par rapport à un autre.

fastidieuse sur stations de travail standards.

Partant de ce constat, le contexte applicatif de la modélisation géométrique s'est très rapidement développé dans le domaine de la RV. Il a donné lieu au développement de nombreuses applications [Sachs 91, Butterworth 92, Liang 94]. En proposant une visualisation stéréoscopique 3-D, ainsi que des périphériques à six DDL (permettant de contrôler simultanément la position et l'orientation d'un objet virtuel), la RV permet l'adéquation entre l'espace de visualisation et l'espace de manipulation. Cette adéquation a été initialement perçue comme une avancée vers des applications de modélisation plus directes et plus intuitives. Effectuer une tâche 3-D, dans un espace 3-D, à l'aide de périphériques 3-D, paraît plus naturel qu'effectuer la même tâche avec une souris sur une station de travail classique. Par exemple, pour effectuer une déformation, l'utilisateur peut, en un seul geste, désigner la partie de l'objet à déformer et définir la trajectoire de déformation. Un exemple est donné en Figure 1.1.

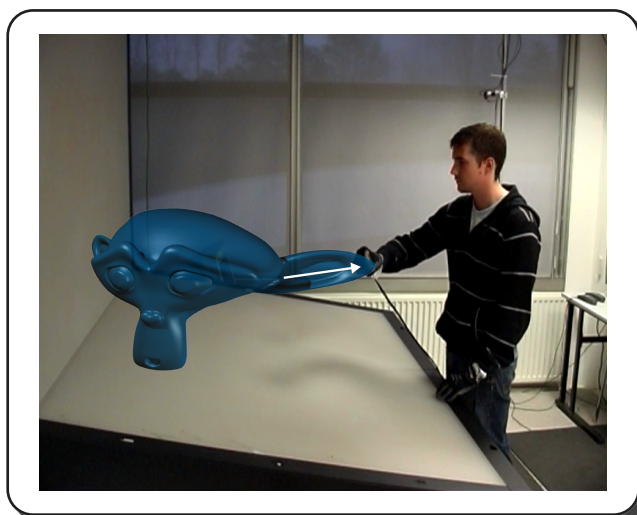


FIGURE 1.1 – L'utilisateur déforme l'objet directement avec la main.

Dans la section suivante, nous présentons et analysons l'ensemble des applications de modélisation proposées en RV. Au delà des outils de modélisation proposés, nous nous intéressons plus particulièrement aux techniques d'interaction utilisées dans la cadre de ces différentes applications. Nous ne nous intéressons pas ici aux différents travaux consistant à fournir des modèles mathématiques permettant la manipulation directe de modèles 3-D [Botsch 04].

1.2 Applications de modélisation en réalité virtuelle

Cette section regroupe une grande partie des applications de modélisation géométrique développées dans des environnements de RV. Nous différencions les applications en fonction des techniques d'interaction utilisées pour créer / manipuler les modèles 3-D. Dans un premier temps, nous présentons les applications où la façon d'interagir avec l'environnement est basée majoritairement sur des interactions libres (*i.e.* des techniques interactions avec lesquelles l'utilisateur manipule librement dans l'espace 3-D). Dans un second temps, nous décrivons les applications introduisant divers types de contraintes au cours des différentes phases de manipulation des modèles 3-D. Nous parlerons alors d'interaction contrainte.

1.2.1 Interaction libre

Depuis les années 90, de nombreuses applications de modélisation ont fait leur apparition en RV. 3-Draw [Sachs 91] est une des premières applications de modélisation exploitant l'utilisation de périphériques 3-D. Elle permet la création de formes 3-D directement dans l'espace de travail. Dans le cadre de ce travail, Sachs *et al.* utilisent des capteurs à six DDL. Ces capteurs sont incorporés, à la fois dans un stylet, mais également dans une tablette. L'utilisateur peut alors créer des courbes de deux manières différentes. La première consiste à définir les courbes directement dans l'espace 3-D en effectuant un geste libre avec le stylet. L'utilisateur positionne des points de contrôle qui sont utilisés pour définir une spline interpolant ces différents points. Définir une courbe à l'aide d'un geste libre permet de donner une grande liberté à l'utilisateur. L'autre manière de créer des courbes consiste simplement à utiliser la tablette comme support physique pour le stylet. Cela permet de créer des **courbes contraintes**². Dans 3-Draw, nous trouvons également des outils (par exemple de déformation) permettant d'éditer les différentes courbes. Les courbes ainsi créées permettent de constituer un réseau de courbes utilisé pour générer des surfaces 3-D.

De nombreuses autres applications se sont inspirées des travaux de Wesche *et al.* [Wesche 00]. Ils proposent des outils de manipulation directe permettant aux utilisateurs de créer / modifier des courbes. Les opérations de modifications consistent à pouvoir lisser ou accentuer localement la courbure. D'une manière similaire à 3-Draw, l'utilisateur peut également attacher des courbes ensembles afin de créer des réseaux de courbes. Une fois le réseau défini, l'application peut générer automatiquement la surface 3-D correspondante. Ici encore, l'objectif est de fournir des outils directs et intuitifs pour la

2. Ce sont des courbes contenues dans un plan. Cette technique a été introduite en raison de la difficulté rencontrée par les utilisateurs à pouvoir se concentrer sur la manipulation de nombreux DDL à la fois.

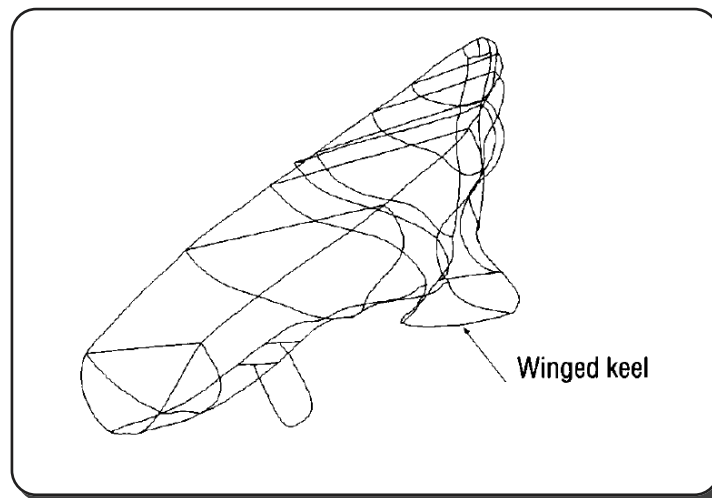


FIGURE 1.2 – Un réseau de courbes 3-D. Ce réseau sert à définir les bords d'une surface. Image extraite de [Sachs 91].

modélisation, un pré-requis essentiel pour les applications de modélisation en environnement immersif [Deisinger 01]. Les travaux de Wesche *et al.* [Wesche 01] ont mené à la mise en place de *FreeDrawer*, une application de modélisation basée sur la création de réseaux de courbes 3-D. Dans cette application, nous retrouvons tous les opérateurs définis dans [Wesche 00] ainsi que des opérateurs permettant de déformer directement les surfaces 3-D. L'utilisateur peut, en désignant directement avec le périphérique d'interaction, déformer localement la surface en la tirant ou en la poussant.

Les applications présentées jusqu'à présent se basent sur la création de surfaces à travers la mise en place de réseaux de courbes 3-D. D'autres applications proposent des approches différentes à la création de modèles 3-D. Dans THRED [Shaw 94], l'utilisateur débute avec une surface carrée. À partir de cette surface, il est possible d'ajouter des carrés adjacents ou encore de supprimer des carrés afin de modéliser une surface grossière. Le travail d'édition se fait à différents niveaux de détails. Le raffinement de la surface se fait à l'aide d'une méthode simple de subdivision, illustrée en Figure 1.3. En travaillant à différents niveaux d'édition, l'utilisateur peut effectuer des modifications plus ou moins fines sur la surface. L'édition se fait en déplaçant les sommets du maillage à l'aide d'une technique de manipulation directe. L'utilisateur n'a donc qu'à saisir le point à manipuler et à le déplacer directement dans l'espace 3-D. Toutefois, la manipulation directe étant jugée comme imprécise, la manipulation des sommets peut également se faire à l'aide d'outils contraignant la manipulation à un plan ou encore à un axe particulier du repère du monde virtuel. Le type de contrainte est, ici, défini par la main non-dominante. Lorsque l'utilisateur active la contrainte selon un axe, l'axe utilisé est défini comme l'axe du repère virtuel le plus proche de la direction donnée par l'axe Z

du périphérique. De même, dans le cas où l'utilisateur active la manipulation contrainte à un plan, le plan est défini par l'orientation du périphérique.

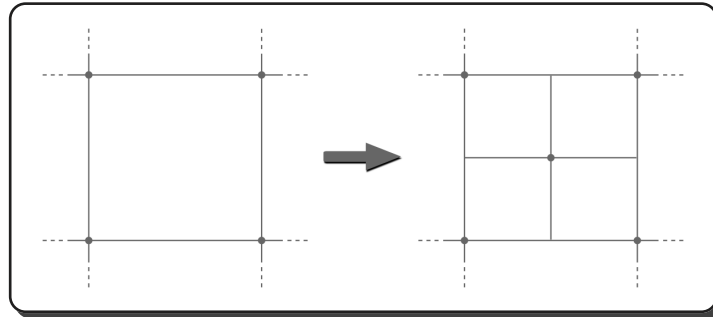


FIGURE 1.3 – La passage d'un niveau de détail au niveau de détail suivant se fait en subdivisant chaque carré de la surface à l'aide du schéma de subdivision ci-dessus.

Dans l'application proposée par Schkolne *et al.* [Schkolne 01], l'utilisateur peut créer directement des surfaces 3-D en manipulant des pinceaux 3-D. Une illustration de cette technique est donnée en Figure 1.4. Une fois la surface initiale créée, l'utilisateur peut manipuler cette surface au travers d'interfaces tangibles. En manipulant un périphérique physique particulier pour chacun des outils de modélisation, l'utilisateur peut modifier la surface afin de créer des modèles plus complexes. L'utilisateur peut mettre à l'échelle une partie du modèle 3-D, pousser localement le modèle ou encore supprimer des parties de la surface. L'objectif principal de cette application est de fournir des outils de manipulation libre, permettant d'accroître le pouvoir d'expression des utilisateurs, notamment dans la phase de conception initiale.

VLEGO [Kiyokawa 97] a été développée par Kiyokawa *et al.* Le principe de construction des modèles 3-D qu'ils proposent est basé sur l'assemblage de blocs géométriques simples. L'utilisateur dispose d'un ensemble de briques de base qu'il peut alors assembler pour créer des modèles 3-D plus complexes. Un exemple de construction est donné en Figure 1.5. Dans le cadre de l'utilisation de VLEGO, Kiyokawa *et al.* ont remarqué que l'assemblage des briques – une tâche à six DDL – est complexe à effectuer en utilisant une technique de manipulation directe. Pour simplifier l'assemblage, ils proposent d'utiliser une technique de *snap-to-grid*³, à la fois pour le positionnement – dont la grille a une résolution de 1 cm – et pour l'orientation en limitant les orientations possibles à 0 °, 90 °, 180 ° et 270 °. Dans la version 2.0 de l'application de modélisation VLEGO [Hiraki 98], Hiraki *et al.* introduisent la notion de **contrainte géométrique**. Les contraintes géométriques sont des éléments – 1 DDL en translation, 2 DDL en

3. L'outil de *snap-to-grid* consiste à définir une grille régulière dans l'espace 3-D ayant une résolution donnée. Les positions alors possibles pour l'ensemble des objets se trouvant dans l'espace 3-D sont définies par les sommets de la grille.



FIGURE 1.4 – L'utilisateur crée une surface 3-D en manipulant un pinceau 3-D à l'aide d'un gant de donnée. Image extraite de [Schkolne 01].

translation, 1 DDL en rotation et 2 DDL en rotation – qui peuvent être associés à des briques. Toutefois, l'utilisation de ces contraintes n'est pas proposée pour la tâche d'assemblage mais plutôt pour définir des comportements d'objets dans l'environnement virtuel (par exemple un tiroir ne pouvant se déplacer que selon un axe de translation et non dans l'espace 3-D).

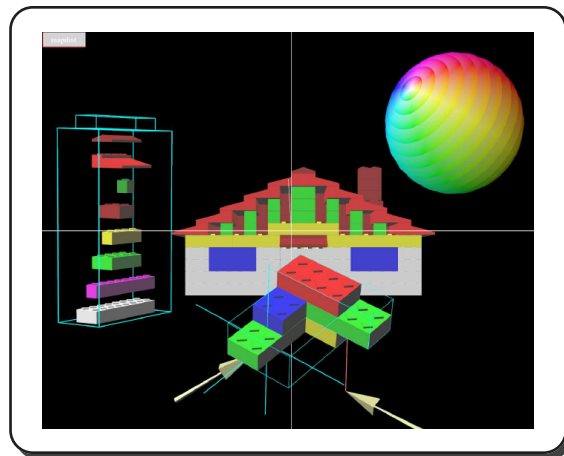


FIGURE 1.5 – Un exemple de construction effectuée avec VLEGO. Image extraite de [Kiyokawa 97].

Une dernière catégorie est celle des applications de modélisation basées sur le *clay modelling*. Dans ces applications, l'utilisateur modélise des objets 3-D en les manipu-

lant comme s'ils étaient fait de glaise. En Figure 1.6, nous pouvons voir le type de comportement que peut avoir l'objet au contact du curseur 3-D sphérique. Dans ce type d'application les concepteurs essaient de tirer profit de l'aspect intuitif que propose la manipulation directe. Dans les différentes applications de *clay modelling*, développées en environnement de RV, la manipulation proposée se fait exclusivement en utilisant la manipulation directe [Kameyama 97, Moritz 00, Ueda 05]. Elle permet, dans ce contexte, de fournir une technique d'interaction plus fidèle à la manière dont nous manipulons dans la réalité.

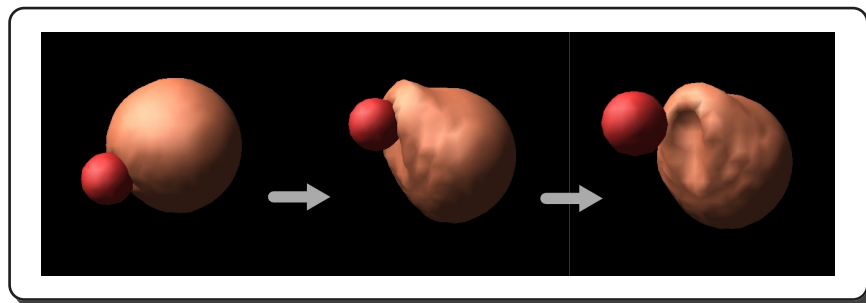


FIGURE 1.6 – La sphère représente le curseur 3-D utilisé pour modeler l'objet 3-D.

L'objectif de ces travaux est de proposer des techniques d'interaction adaptées permettant une manipulation naturelle et directe des entités géométriques. Néanmoins, dans les exemples précédents, aucune évaluation quantitative des performances offertes par les différentes techniques proposées n'a été effectuée. Toutefois, dans ce type d'applications, il est nécessaire d'avoir un contrôle précis des actions effectuées dans l'environnement. Dans un contexte de manipulation précise, il semblerait que l'interaction 3-D ne soit pas totalement adaptée. Une des solutions proposées par les auteurs consiste à introduire des contraintes durant certaines phases de la manipulation. Nous présentons, à présent, des applications où la notion de contraintes est mise en avant dans la manipulation.

1.2.2 Interaction contrainte

Dans leur travaux, Wesche *et al.* [Wesche 00] proposent une modalité d'interaction à deux mains où la main non-dominante va positionner et orienter un plan virtuel sur lequel l'utilisateur va pouvoir dessiner des courbes 2-D. Ce type de modalité est quelque peu en contradiction avec les objectifs initiaux qui sont de proposer des moyens de créer et manipuler des objets directement dans l'espace 3-D. Dans 3DM [Butterworth 92], Butterworth *et al.* introduisent des outils de *snap-to-grid* ainsi que des outils de *snap-*

*to-plane*⁴ afin de « combler le manque de contraintes » [Butterworth 92].

Dans l'application proposée par Grossman *et al.* [Grossman 01] [Grossman 02], l'utilisateur peut là aussi créer des surfaces 3-D en définissant des réseaux de courbes. La technique proposée est celle du *Digital Tape Drawing*. Cette technique est inspirée des méthodes utilisées pour dessiner les courbes principales de carrosseries dans le milieu automobile. La principale méthode consiste à dessiner, à l'aide d'un scotch, le profil de la carrosserie sur une surface plane – en général un mur. Avec le *Digital Tape Drawing*, la création de courbe se fait dans des plans de manipulation que l'utilisateur peut définir et déplacer. La technique permet alors de définir les courbes sur les plans de manière similaire à l'utilisation d'un scotch. Ce qui est intéressant ici, ce n'est pas la technique permettant de définir les courbes dans les plans, mais l'utilisation même des plans. Ces derniers permettent de contraindre l'interaction à un espace bidimensionnel et évite ainsi la création directe de courbes dans l'espace 3-D.

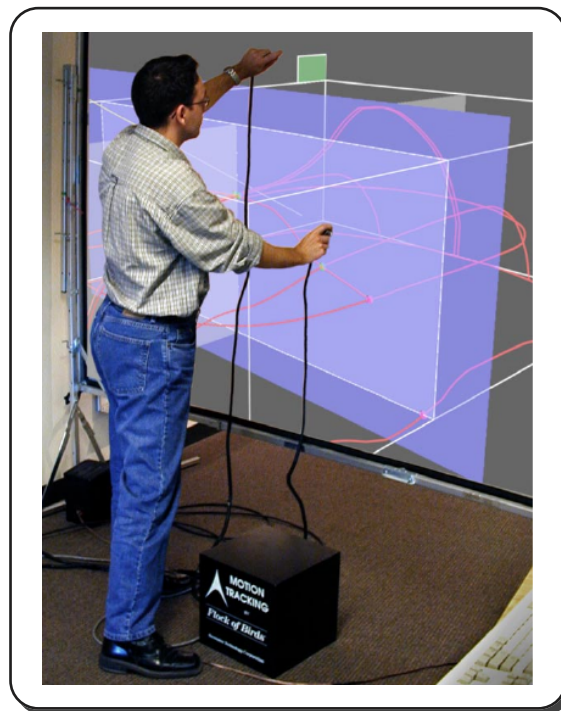


FIGURE 1.7 – Création d'un réseau de courbes 3-D. Image extraite de [Grossman 02].

Stork *et al.* [Stork 97] ont développé une application de conception assistée par ordinateur (CAO) exploitant l'utilisation de périphériques 3-D. L'objectif de cette application est de tirer profit de l'interaction 3-D dans les tâches où elle semble appropriée.

4. Le principe est similaire à un outil de type *snap-to-grid*. La position d'un objet dans l'espace 3-D est contraint à un plan défini par l'utilisateur.

L'interaction est ensuite contrainte lorsque c'est nécessaire. Par exemple, l'utilisateur peut créer des modèles librement dans l'espace. Dans ce cas, l'interaction libre est avantageuse puisqu'elle permet de définir de nombreux paramètres en un seul geste. Nous pouvons voir, en Figure 1.8, qu'en un seul geste, l'utilisateur peut définir, la position du centre du cylindre, la hauteur du cylindre ainsi que son rayon. Ce principe est utilisé pour la création de nombreuses primitives géométriques 3-D. Toutefois, certaines tâches – comme la définition d'une cote – demandent une manipulation précise. Cette précision ne pouvant pas être apportée par la manipulation directe, Stork *et al.* proposent d'intégrer des outils permettant de contraindre le mouvement de l'utilisateur. Parmi ces outils, nous retrouvons le *snap-to-grid* 3-D ou encore la possibilité de pouvoir contraindre le mouvement du curseur 3-D à un axe ou encore à un plan. Ce sont les mêmes types d'outils que ceux proposés dans THRED.

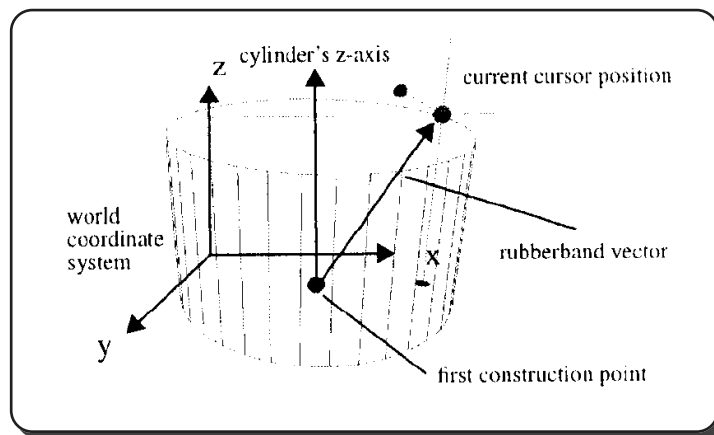


FIGURE 1.8 – La position initiale du geste définit la position du centre du cylindre. En déplaçant la main, l'utilisateur définit à la fois la hauteur ainsi que la largeur du cylindre créé. Image extraite de [Stork 97].

S'ajoutent à cela, des outils de contrainte du mouvement basés sur la géométrie du modèle 3-D. Stork *et al.* proposent de contraindre de manière dynamique les gestes des utilisateurs en fonction de la partie du modèle sur laquelle ils travaillent. Par exemple, lorsque l'utilisateur travaille au niveau d'une arête du modèle celui-ci peut vouloir que le déplacement du curseur 3-D soit contraint sur l'arête. De même lorsque celui-ci travaille à proximité d'une face, les déplacements du curseur 3-D doivent être limités au plan contenant la face. Ces contraintes sont appelées **contraintes contextuelles basées sur la topologie**. Nous retrouvons ces contraintes, basées sur la topologie, dans des travaux ultérieurs tels que ceux effectués par De Amicis *et al.* [De Amicis 01].

Ces outils trouvent leur origine dans JDCAD [Liang 94], une des applications de modélisation que nous jugeons comme parmi les plus abouties. Cette application, déve-

loppée par Liang *et al.*, fonctionne sur une station de travail permettant une visualisation stéréoscopique. L'interaction se fait en utilisant le Bat, un périphérique à six DDL. Avec JDCAD, l'utilisateur peut instancier des primitives géométriques de complexités différentes – cubes, cônes ou encore trapézoïdes. D'une manière similaire aux travaux précédents, tous les paramètres nécessaires à la création d'un objet sont spécifiés au travers d'un seul geste 3-D lorsque c'est possible. Pour des objets complexes dépendant de plus de trois paramètres, l'utilisateur ne contrôle directement que trois paramètres, les autres étant calculés automatiquement mais pouvant être modifiés par la suite. Les bénéfices de l'interaction 3-D qu'ils espèrent ne sont pas attendus au niveau de la création des modèles 3-D, mais plutôt au moment des phases de modification de ces modèles. Ces modifications sont souvent nombreuses et nécessitent de la précision. L'interaction 3-D ne semble pas adaptée à de telles modifications. Sont mises en cause notamment les difficultés à percevoir la position en profondeur du curseur 3-D par rapport au modèle. Liang *et al.* proposent alors une méthode appelée *region-based reshaping*. Elle consiste à identifier les paramètres modifiables d'un objet en fonction de la position du curseur 3-D. Ainsi, si le curseur 3-D se trouve à proximité du centre d'une des extrémités d'un cylindre, alors l'utilisateur pourra modifier la hauteur du cylindre, si le curseur se trouve à proximité du centre du cylindre alors l'utilisateur pourra en modifier le rayon. Lorsque le curseur se trouve à proximité du bord d'une des extrémités du cylindre, l'utilisateur pourra en modifier la hauteur et le rayon simultanément. L'apparence visuelle du curseur est modifiée afin d'indiquer les modifications possibles. La Figure 1.9 illustre le fonctionnement de cette technique dans le cas de la modification des différents paramètres d'un cylindre. Cette méthode permet de réduire le nombre de DDL manipulés simultanément afin de permettre à l'utilisateur de se concentrer sur la manipulation de dimensions spécifiques (comme par exemple la hauteur du cylindre).

L'apparition de dispositifs à retour d'effort a permis le développement d'applications de modélisation introduisant un nouveau type de contraintes que nous appellerons **contraintes physiques**. À l'opposé des contraintes virtuelles, que nous avons vu précédemment – généralement implémentées sous la forme de fonctions filtrant les données renvoyées par les périphériques 3-D, les contraintes physiques sont appliquées au niveau du contrôle moteur et peuvent être ressenties physiquement par l'utilisateur. Dans l'ensemble des applications utilisant un périphérique à retour d'effort, le périphérique est utilisé afin de permettre à l'utilisateur de toucher la surface des modèles 3-D. Le fait de pouvoir toucher et se déplacer sur les surfaces 3-D permet de réduire la dimension⁵ de la tâche à effectuer. En effet, déplacer le périphérique sur la surface d'un objet 3-D revient à se déplacer localement sur une surface isomorphe à un plan. Cette technique d'interaction est utilisée par exemple dans InTouch [Gregory 00] ainsi que dans ArtNova [Foskey 05]. Au travers de l'utilisation de pinceaux virtuels, l'utilisateur peut

5. Nous définissons la dimension de la tâche comme le nombre de DDL à manipuler pour effectuer la tâche. Le positionnement 3-D est une tâche à trois dimensions.

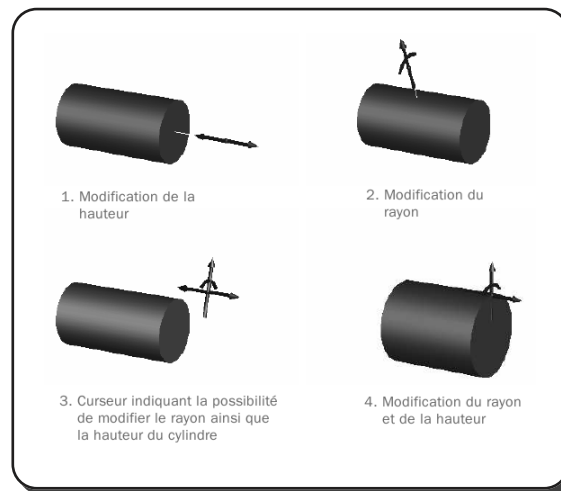


FIGURE 1.9 – Les différentes modifications possibles en fonction de la position du curseur 3-D par rapport au modèle manipulé. Image extraite de [Liang 94].

alors déformer localement le modèle 3-D en poussant ou en tirant la surface. Le retour d'effort permet ici de contraindre la désignation à la surface de l'objet. Les pinces peuvent également servir à dessiner sur la surface même de l'objet [Gregory 00] ou encore à appliquer des textures [Foskey 05].



FIGURE 1.10 – Résultat obtenu avec l'application InTouch. Image extraite de [Gregory 00].

La dernière application exploitant un périphérique à retour d'effort a été développée par Keefe *et al.* [Keefe 05, Keefe 07]. Cette dernière permet de faire ce qu'on appelle du **dessin virtuel**. La technique proposée mobilise les deux mains de manière asymétrique pour dessiner dans l'espace 3-D. La main non dominante manipule un capteur permettant de définir l'orientation d'un plan de dessin virtuel. L'utilisateur peut alors dessiner sur ce plan avec la main dominante en utilisant un PhantomTM. En plus de définir une courbe dans l'espace 3-D, l'utilisateur peut faire varier la largeur de la courbe en fonction de la pression qu'il exerce sur le plan de dessin virtuel avec le Phantom. En contraignant le geste sur un plan 2-D, le dessin ne se fait plus à main levée dans l'espace 3-D. Cela permet à la fois de réduire la dimensionnalité de la tâche et de bénéficier d'un support physique au cours de l'interaction. L'évaluation de cette technique dans le cadre d'une tâche de suivi de courbe dans un espace 3-D a montré que ses performances en matière de précision sont meilleures que celles offertes par une technique de dessin 3-D à main levée.

1.2.3 Bilan

Nous avons présenté une grande partie des applications de modélisation géométrique développées dans le cadre de la RV. Ces applications, bien que reposant sur différents procédés de création de modèles 3-D (*solid modelling* ou *clay modelling*), proposent d'utiliser les périphériques 3-D comme moyen de rendre le processus de création plus direct et plus intuitif. Cela a été le cas notamment dans les premières applications de modélisation qui ont été proposées (3-Draw ou encore 3DM).

Toutefois, dans les applications nous pouvons voir apparaître l'utilisation de contraintes de différentes natures :

- **Contraintes virtuelles explicites** : l'utilisateur spécifie explicitement – par un geste ou encore par l'activation d'un item dans un menu – le type de contrainte qu'il souhaite utiliser. C'est le cas notamment dans THRED [Shaw 94] ;
- **Contraintes implicites basées sur la géométrie ou la topologie** : l'utilisateur n'a pas à spécifier explicitement la contrainte, elle est déduite automatiquement en fonction de la position du curseur 3-D par rapport à l'objet manipulé. On retrouve ce type de contraintes dans JDCAD [Liang 94] ou dans l'application développée par Stork *et al.* [Stork 97] ;
- **Contraintes physiques** : l'interaction est contrainte au travers du périphérique d'interaction, qui est en général, un périphérique à retour d'effort. On les retrouve dans les application InTouch [Gregory 00] ou encore ArtNova [Foskey 05].

Quelle que soit la manière dont les contraintes sont appliquées, l'objectif est le même : réduire temporairement ou au cours de l'ensemble de la tâche, la dimension de l'espace de manipulation afin d'augmenter les performances de l'utilisateur. Ce phénomène

n'apparaît pas seulement dans le contexte des applications de modélisation, mais dans l'ensemble des tâches fondamentales en RV [Bowman 99] : la navigation, la sélection, la manipulation et le contrôle de système.

1.3 Manipulation en réalité virtuelle

Nous nous intéressons, ici, uniquement aux techniques liées à la manipulation en RV. Les techniques proposées dans ce cadre sont susceptibles d'être utilisées dans un grand nombre de tâches de modélisation géométrique telles que le positionnement (par exemple les points de contrôle d'une courbe) et l'orientation d'entités géométriques, le suivi de trajectoire dans un espace 3-D à l'aide d'un curseur, la définition de trajectoires de déformation, *etc.* Dans la prochaine section, nous allons décrire les techniques usuelles de manipulation en RV.

1.3.1 Techniques existantes

Par soucis de clarté nous distinguons ici les techniques **égocentriques**, *i.e.* celles avec lesquelles l'utilisateur manipule directement à partir de l'intérieur du monde virtuel, des techniques **exocentriques**, *i.e.* celles avec lesquelles le monde virtuel est manipulé de l'extérieur. Cette distinction est issue de la classification proposée par Bowman *et al.* [Bowman 98].

Techniques egocentriques

Une des techniques les plus répandues est la technique dite de la **main virtuelle** [Sturman 89]. Elle permet à la fois de sélectionner et de manipuler un objet virtuel. L'utilisateur sélectionne un objet en y plaçant sa main, la validation se faisant soit en fermant la main, soit après un temps donné passé à l'intérieur de l'objet. Une fois l'objet sélectionné, l'utilisateur peut le manipuler en déplaçant et en tournant sa main. Les déplacements et rotations sont alors appliquées à l'échelle 1 : 1. Cela permet une manipulation directe [Shneiderman 87] de l'objet dans le monde virtuel. Une manipulation à l'échelle 1 : 1 est naturelle pour les utilisateurs mais elle peut poser un certain nombre de problèmes, notamment dus à des contraintes anatomiques limitant l'amplitude ou encore la précision des mouvements effectués.

Frees *et al.* proposent une technique appelée PRISM. Il s'agit d'une méthode inspirée de la main virtuelle. La méthode de sélection / manipulation est similaire à celle utilisée dans la métaphore de la main virtuelle. La différence réside dans la relation qu'il existe entre les déplacements du périphérique d'interaction et les déplacements effec-

tifs de l'objet manipulé. La relation est définie par un facteur d'ajustement non-linéaire dépendant de la vitesse de déplacement du périphérique. La définition de la fonction calculant le facteur d'ajustement est basée sur le *Two-Component model* [Meyer 88] dont le principe est le suivant : lorsque l'utilisateur déplace le périphérique rapidement, il désire déplacer rapidement l'objet virtuel et n'a dans ce cas, pas besoin d'une précision élevée. Au contraire, lorsque celui-ci déplace le périphérique lentement, il désire être précis. La filtrage est appliqué de la manière suivante :

$$\vec{D}_{\text{objet}} = \vec{D}_{\text{périphérique}} * f(V_{\text{périphérique}})$$

où $\vec{D}_{\text{périphérique}}$ est le déplacement du périphérique et $V_{\text{périphérique}}$ est la vitesse de déplacement instantanée. La fonction f est définie de la manière suivante :

$$f(V) = \begin{cases} 0 & , \text{ si } V < V_{\min} \\ \min(1, \frac{V}{V_{\text{scale}}}) & , \text{ sinon} \end{cases} \quad (1.1)$$

$V_{\min}$ est la vitesse minimale nécessaire pour que l'objet soit déplacé. V_{scale} est la vitesse utilisée pour l'ajustement dynamique de la vitesse de l'objet. Ceci permet de donner une plus grande précision à l'utilisateur en phase finale de la tâche tout en permettant de traverser de grandes distances lors de la phase d'approche initiale.

Kitamura *et al.* [Kitamura 99] proposent une technique de manipulation basée sur l'utilisation de baguettes chinoises. À l'aide de ces baguettes, l'utilisateur peut saisir, positionner et orienter des objets dans un espace 3-D. Une illustration de cette technique est donnée en Figure 1.11. Plusieurs adaptations ont été proposées afin de permettre, par exemple, de définir un axe de rotation avec une des baguettes ainsi que l'amplitude de la rotation en utilisant l'écartement entre les baguettes.

Des dispositifs matériels spécifiques permettent d'interagir avec un environnement de RV. C'est le cas notamment du CAT [Hachet 02] (représenté sur la partie gauche de la Figure 1.12), qui est un périphérique permettant à la fois de manipuler un objet et de naviguer dans une scène 3-D. Dans un contexte de manipulation, l'utilisateur peut translater l'objet virtuel dans les trois dimensions de l'espace individuellement ou simultanément. Il en va de même pour orienter un objet dans l'espace 3-D. Les possibilités de manipulation sont illustrées sur la partie droite de la Figure 1.12.

Le DigiHaptic [Casiez 04] est un périphérique spécifique à trois DDL. Ce dernier est illustré en Figure 1.13. Il permet de contraindre l'interaction en facilitant la manipulation individuelle de DDL spécifiques. Ce périphérique est notamment utilisé dans des tâches de suivi de trajectoires et de navigation.

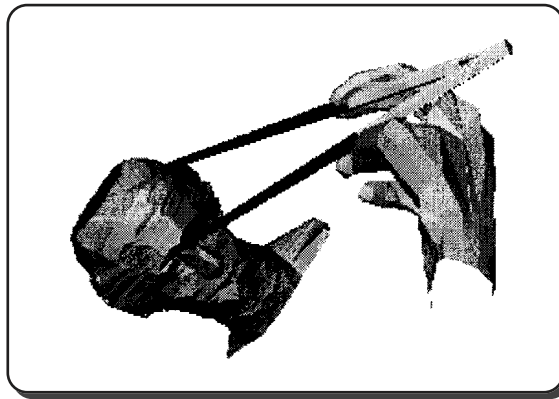


FIGURE 1.11 – Exemple de manipulation avec la métaphore des baguettes chinoises. Image extraite de [Kitamura 99].

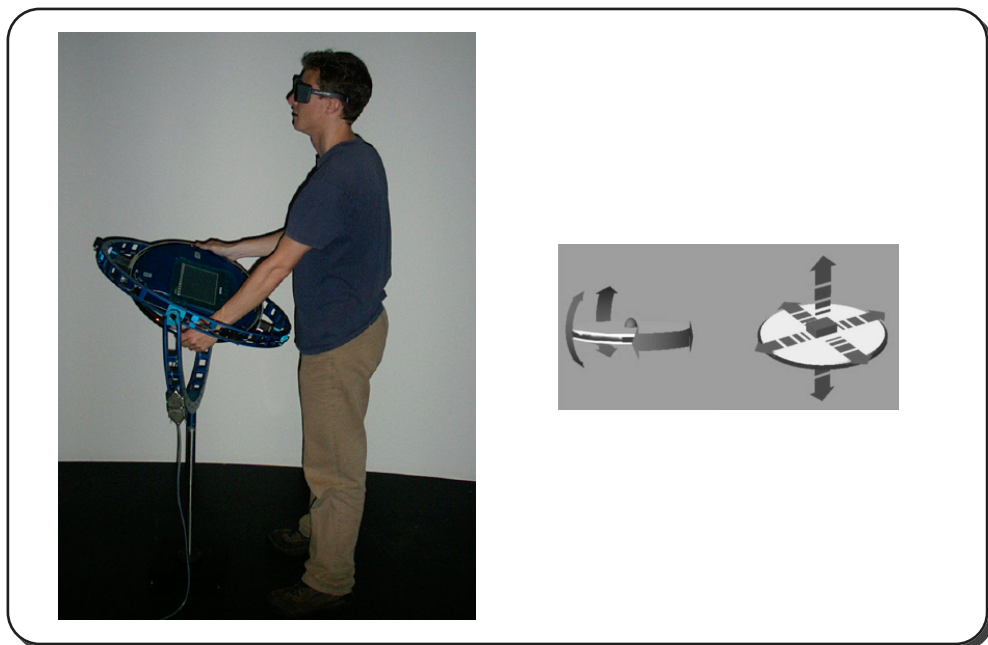


FIGURE 1.12 – Le dispositif du CAT. Image extraite de [Hachet 04].

Techniques exocentriques

Une des premières techniques exocentriques proposées pour la manipulation est celle du *World-In-Miniature* proposée par Stoackley *et al.* [Stoakley 95]. L'utilisateur va pouvoir sélectionner et manipuler indirectement les objets présents dans la scène au travers d'une maquette miniature du monde virtuel. Avec la main dominante, l'utilisateur peut

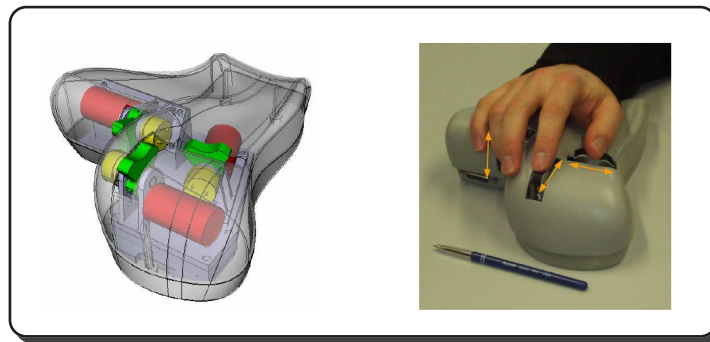


FIGURE 1.13 – Le périphérique DigiHaptic.

sélectionner et manipuler les objets virtuels par la méthode de la main virtuelle. Avec la main non dominante, l'utilisateur peut manipuler la représentation miniature du monde virtuel. En utilisant cette technique, la manipulation est souvent imprécise. Ceci est dû au fait que l'échelle de manipulation est souvent trop grande. De petits déplacements dans le monde miniature vont induire des déplacements de grandes amplitudes dans le monde virtuel.

Pierce *et al.* [Pierce 99] proposent la technique des *Voodoo Dolls* avec laquelle l'utilisateur peut créer ses propres représentations miniatures d'objets virtuels. La sélection se fait avec une technique appelée *head crusher* qui consiste à sélectionner l'objet non pas dans l'espace 3-D mais dans l'espace image. L'objet une fois sélectionné, une maquette miniature est créée au niveau de la main non dominante, la main dominante pouvant alors déplacer et orienter l'objet virtuel.

La technique *Balloon Selection* [Benko 07] est à la base une technique de sélection 3-D, toutefois, celle-ci nous semble également appropriée pour le positionnement d'objets. Cette technique simule l'utilisation d'une interface tactile à plusieurs points de contact. Lorsque l'utilisateur touche la surface avec la main dominante, un ballon apparaît au niveau du point de contact. En déplaçant la main dominante l'utilisateur peut définir la position du ballon dans le plan horizontal. Lorsque l'utilisateur touche la surface avec la main non dominante tout en restant en contact avec la main dominante, un fil reliant le ballon à la main non dominante apparaît. En modifiant la longueur du fil – en rapprochant ou éloignant les deux points de contact – l'utilisateur modifie la hauteur du ballon.

Techniques hybrides

D'autres techniques, à la fois exocentriques et egocentriques, ont été proposées. Parmi celles-ci nous retrouvons la technique HOMER [Bowman 97]. L'utilisateur sélectionne

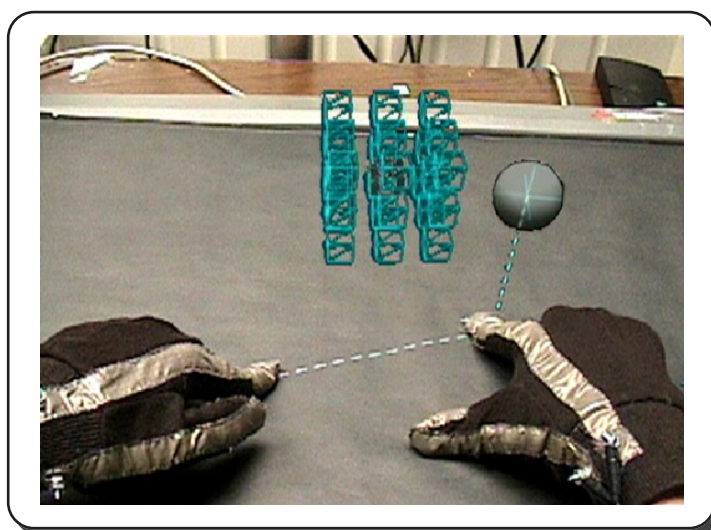


FIGURE 1.14 – Le point de contact de la main dominante définit la position dans le plan horizontal, la distance entre les deux points de contact permet de manipuler la hauteur du ballon. Image extraite de [Benko 07].

l'objet à manipuler avec une technique de *Ray-Casting*. Cette technique consiste à lancer un rayon émanant du périphérique d'interaction. Au moment de la validation, l'objet le plus proche intersectant le rayon est sélectionné. Une fois la sélection effectuée, les mains virtuelles se placent à proximité de l'objet afin d'en permettre la manipulation. Le but de cette technique est d'offrir une sélection simple et rapide en utilisant une technique de *ray casting* tout en permettant une manipulation efficace avec la technique de la main virtuelle. Toutefois, selon la distance à l'objet, la manipulation peut être plus ou moins aisée. Des difficultés de perception [Wanger 92] [Interrante 06] [Interrante 07], notamment en ce qui concerne la profondeur, peuvent rendre une manipulation à de longues distances inappropriée.

La technique *Scaled HOMER* [Wilkes 08] est une extension de la technique HOMER. Il s'agit d'une méthode hybride entre la technique HOMER et la technique PRISM. Elle permet une manipulation à distance combinée à une fonction de correspondance non-linéaire. Cette manipulation se révèle à la fois efficace dans les phases de positionnement grossier et précise dans les phases de correction de la position de l'objet manipulé.

La dernière technique présentée est la technique du **changement d'échelle** [Mine 97]. La sélection se fait ici, dans le plan image, en utilisant la direction du regard. Une fois l'objet sélectionné, le système diminue la taille du monde virtuel en considérant la position de la tête de l'utilisateur comme centre du monde. La taille du monde virtuel est réduite jusqu'à ce que l'objet sélectionné se trouve à portée de main. Une fois l'objet

relâché, le monde virtuel retrouve sa taille initiale.

1.4 Analyse

Nous avons décrit dans ce chapitre l'ensemble des techniques d'interaction que nous jugeons les plus pertinentes par rapport au contexte applicatif de notre travail. Nous allons, à présent, analyser l'ensemble des techniques proposées afin d'essayer de dégager un ensemble de méthodes utilisées pour améliorer le contrôle de l'utilisateur.

1.4.1 Une manipulation plus directe

Dans un premier temps, les techniques d'interaction proposées sont essentiellement basées sur une manipulation directe, tentant de supprimer l'utilisation de composants intermédiaires au cours de la manipulation. Dans le cadre des applications de modélisation géométrique, cet axe d'étude se traduit par la proposition de modèles mathématiques permettant une manipulation plus directe des entités géométriques. C'est le cas dans l'application *FreeDrawer* développée par Wesche *et al.* [Wesche 00]. Par rapport à des applications de modélisation classiques, les courbes 3-D sont représentées par des courbes de Bézier ou encore des NURBS. La création ainsi que la manipulation de ce type de courbes se faisant au travers de points de contrôle, ces modèles de représentation ne permettent pas une manipulation directe. Afin de permettre une manipulation directe de courbes 3-D, un ensemble d'opérateurs de modélisation, ne nécessitant pas la manipulation de points de contrôle, a été proposé et intégré dans *FreeDrawer*. On retrouve cette philosophie dans l'application THRED [Shaw 94] où l'utilisateur va pouvoir directement manipuler les sommets de la surface 3-D pour éditer le modèle 3-D.

On voit également apparaître des techniques d'interaction permettant de créer / manipuler directement des courbes et surfaces. C'est le cas notamment dans l'application proposée par Schkolne *et al.* [Schkolne 01] où l'utilisateur peut créer directement des surfaces dans l'espace 3-D à l'aide de pinceaux surfaciques. Le constat semble identique du côté des techniques de manipulation plus généralistes comme la **main virtuelle** [Sturman 89]. Les techniques permettent de manipuler directement les objets présents dans l'environnement virtuel. Il en va de même pour les techniques exocentriques telles que le *World-In-Miniature* [Stoakley 95] ou encore les *Voodoo Dolls* [Pierce 99]. Toutefois, les techniques proposées offrent très souvent un niveau de performance – en matière de précision – insuffisant. Cette difficulté revêt une importance toute particulière dans le cadre de la modélisation géométrique puisque dans ce cas, le besoin de précision est critique.

1.4.2 Contraindre la manipulation

Que se soit dans le cadre des applications de modélisation ou dans le cadre des techniques d'interaction décrites plus haut, nous pouvons remarquer que l'utilisation de contraintes – virtuelles ou physiques – a très rapidement fait son apparition. Le constat général semble unanime, l'utilisation d'une interaction plus libre permet de manipuler les objets d'une manière plus directe et plus intuitive, la contre partie étant une réduction significative de la précision des utilisateurs. L'introduction de contraintes a pour objectif d'accroître cette précision (*cf.* Section 1.2.2).

Les contraintes virtuelles prennent des formes diverses dans les applications que nous avons vus. Dans le cas de l'application VLEGO [Kiyokawa 97] ou de l'application développée par Grossman *et al.* [Grossman 02], les contraintes sont explicites. Dans VLEGO, l'utilisateur peut positionner les objets sur une grille magnétique et définir des orientations qui sont fixes et alignées sur les axes du repère du monde virtuel. Dans la deuxième application, l'utilisateur définit des courbes qui sont contraintes à des plans virtuels qui sont explicitement manipulés par l'utilisateur. Les contraintes peuvent être implicites, c'est le cas dans les applications de *clay-modelling* où l'outil de sculpture est contraint sur la surface de l'objet. Dans le cadre d'utilisation de contraintes physiques (via l'utilisation d'un périphérique à retour d'effort), les contraintes sont explicites.

En ce qui concerne les applications de modélisation, les périphériques à retour d'effort permettent généralement de toucher les surfaces 3-D. La manipulation est contrainte en obligeant l'utilisateur à toucher la surface pour effectuer des tâches de modélisation. Cela peut être également le cas pour la manipulation de courbes 3-D, comme par exemple avec la technique proposée par Keefe *et al.* [Keefe 05] où l'utilisateur dessine des courbes en dessinant sur un plan dont l'orientation est défini par la main non dominante.

Quel que soit le type de contraintes utilisées, il semblerait qu'elles permettent d'améliorer sensiblement les performances des utilisateurs. Kiyokawa *et al.* ont montré que l'utilisation de contraintes durant la tâche de *docking* permet à la fois de réduire les temps de réalisation mais également d'accroître la précision finale. Benko *et al.* lors de l'évaluation de la technique *Balloon Selection* [Benko 07] ont montré que cette technique permet de réduire le taux d'erreur tout en permettant d'effectuer la tâche dans le même temps qu'une technique de manipulation directe.

L'utilisation de contraintes semble donc accroître la précision des utilisateurs. C'est pourquoi nous proposons d'utiliser des contraintes (virtuelles et physiques) dans le cadre de la modélisation géométrique pour améliorer le contrôle de l'utilisateur. Dans le cadre de notre travail, nous proposons de simuler l'utilisation d'une interface tactile afin d'introduire des contraintes physiques au cours de la manipulation. Cette simulation exploite une spécificité du dispositif matériel. Sur le Workbench, un des deux

écrans se trouve à portée de bras. L'utilisateur peut toucher sa surface pour interagir avec l'environnement virtuel. Par ce biais, nous pouvons simuler l'utilisation d'une interface tactile. Avant de décrire les objectifs de notre travail, nous dressons un état de l'art de l'utilisation des interfaces tactiles dans le cadre de l'interaction 3-D.

1.4.3 Interfaces tactiles et interaction 3-D

Les interfaces tactiles sont maintenant accessibles à tous et bien acceptées par les utilisateurs surtout dans le domaine de la communication mobile. Toutefois, l'utilisation des interfaces tactiles ne se limite pas à ce cadre de travail. Certains travaux s'intéressent à l'utilisation de ce type d'interface pour des tâches 3-D. Martinet *et al.* [Martinet 09] [Martinet 10] proposent d'utiliser une surface tactile pour le positionnement 3-D. Ils proposent deux techniques : la *Z-technique* et *Multi-View*. La première consiste à manipuler l'objet dans le plan X-Y avec la main dominante et à isoler le paramètre de profondeur sur la main non dominante. Avec la seconde technique, l'utilisateur dispose de quatre vues. Il peut déplacer l'objet dans le plan de projection de la vue dans laquelle il manipule. En utilisant les deux mains, l'utilisateur peut manipuler dans deux vues simultanément permettant une manipulation 3-D des objets. Une illustration est donnée en Figure 1.15. Les résultats de leur étude semblent indiquer que les deux techniques offrent un même niveau de précision et permettent d'effectuer la tâche dans un laps de temps similaire. Toutefois, cette étude ne répond pas à une question fondamentale, est-ce que le choix de la décomposition a une influence sur les performances ? Est-ce qu'en isolant une autre dimension de l'espace 3-D les résultats auraient été les mêmes ?

Hancock *et al.* [Hancock 07] proposent un ensemble de techniques permettant de manipuler les 6 DDL d'un objet simultanément au travers d'une interface tactile multi-point. Les techniques proposent différentes correspondances entre les mouvements des points de contact associés à l'index et au pouce et les mouvements de l'objet. Dans leur étude, ils montrent qu'en permettant à l'utilisateur de manipuler l'ensemble des DDL, les performances sont meilleures en matière de précision et de temps de réalisation. Toutefois, dans leur étude Hancock *et al.* ne quantifient pas la coordination des DDL, il est donc impossible de savoir à quel point l'utilisateur a été capable de coordonner les DDL. L'ensemble des bénéfices observés peuvent provenir d'une stratégie basée sur la décomposition, spécialement en fin de tâche, stratégie facilitée par une technique plutôt qu'une autre.

Ces derniers travaux semblent s'orienter vers des techniques permettant d'intégrer le maximum de DDL à la fois dans un même geste. Toutefois, le fait de devoir (ou de pouvoir) décomposer les tâches 3-D peut être un facteur permettant d'influer positivement sur les performances des utilisateurs. Dans leur travail, Nacenta *et al.* [Nacenta 09] s'intéressent à la manipulation 2-D dans le cadre d'interfaces tactiles. Cette manipulation

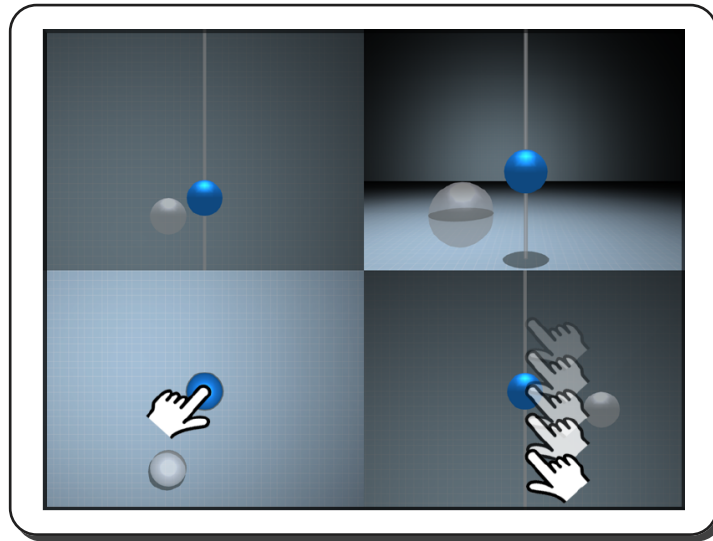


FIGURE 1.15 – L'utilisateur manipule dans les différents point de vue pour positionner l'objet dans l'espace 3-D. Image extraite de [Martinet 10].

implique, généralement, la manipulation simultanée de la position (2-D), de l'orientation (1-D) et du facteur de mise à l'échelle (1-D si uniforme, 2-D sinon). Dans ce cas, il est souvent difficile de simplement déplacer l'objet sans modifier les autres paramètres. Ils proposent alors plusieurs méthodes permettant d'éviter ce phénomène : *gesture matching* et *magnitude filtering*. La décomposition semble avoir un impact positif sur les performances en permettant un meilleur contrôle dans la tâche. Toutefois, cette étude se fait dans le cadre d'un environnement 2-D. Les conclusions semblent difficilement transposables à un environnement 3-D où l'utilisateur est confronté à des difficultés spécifiques, telle que la gestion de la profondeur.

1.5 Approche proposée

Dans le cadre des applications de modélisation géométrique, rares sont les applications exploitant l'utilisation des interfaces tactiles. Une des seules applications connues à ce jour est le *Origami Simulator* [Chang 09]. Cette application permet de créer des objets 3-D au travers de techniques inspirées de l'origami. L'intérêt croissant pour l'utilisation d'interfaces tactiles pour l'interaction 3-D [Martinet 10] semble indiquer que ce type d'interfaces est adaptée à la manipulation en environnement virtuel :

- Elles introduisent une **contrainte physique**. Cette contrainte peut être particulièrement intéressante dans le cadre des applications de modélisation géométrique. Dans de telles applications, les sessions de manipulation peuvent être longues.

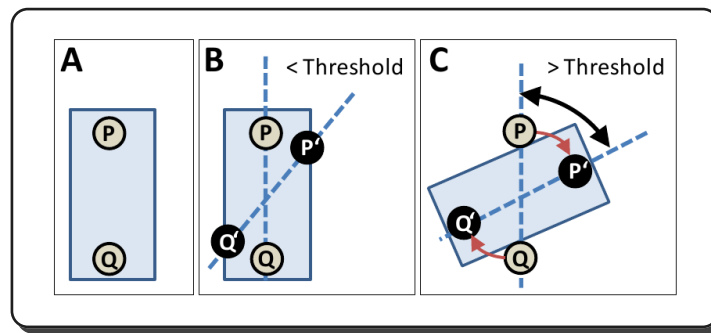


FIGURE 1.16 – La méthode *Magnitude Filtering* est un filtre sur l'amplitude des mouvements. Pour déplacer, tourner ou mettre à l'échelle un objet, il faut que le déplacement soit suffisamment important pour être appliqué. Image extraite de [Nacenta 09].

L'utilisation de techniques d'interaction basées sur l'interaction libre, telles que la main virtuelle, Go-Go, HOMER ou encore PRISM, peuvent ne pas être complètement adaptées à cette situation de manipulation. L'absence de support physique au cours de la manipulation peut introduire rapidement une fatigue importante. Cette fatigue peut se traduire par des mouvements parasites tels que des tremblements et constituer un facteur limitant la précision du geste ;

- Cette contrainte physique nécessite de **décomposer les tâches** d'interaction 3-D. La manipulation au travers d'une interface tactile ne permet de manipuler que deux DDL par point de contact. Il existe des solutions permettant d'intégrer la manipulation de tâches d'interaction 3-D dans un geste d'interaction tactile [Kruger 05] [Hancock 07]. Néanmoins, la décomposition peut être adaptée à des tâches d'interaction 3-D. C'est notamment le cas pour la technique *Balloon Selection* [Benko 07]. Il est possible que les utilisateurs mettent en place des stratégies basées sur la décomposition des tâches. Cela semble être le cas pour des tâches nécessitant la manipulation de nombreux DDL simultanément [Wang 98]. Cette décomposition peut constituer une stratégie visant à réduire la charge cognitive liée à la coordination des dimensions de l'espace de manipulation. Cette décomposition peut également constituer une stratégie visant à surmonter des difficultés liées au contexte de manipulation, comme la gestion de la profondeur.

Néanmoins, nous retrouvons dans les applications de modélisation une grande variété de tâches. En effet, les tâches habituellement utilisées au cours du processus de modélisation vont du positionnement et orientation d'objet, à des tâches plus complexes comme la déformation de modèle 3-D ou encore l'application d'opérations topologiques comme la découpe de surfaces. Les bénéfices que peut engendrer la simulation d'une interface tactile peuvent être différents en fonction des tâches. Afin de proposer des solutions qui soient généralisables à d'autres contextes de manipulation, il semble néces-

saire de mettre en place un cadre théorique nous permettant de caractériser la situation d'interaction. L'objectif est de comprendre l'influence que peuvent avoir des facteurs comme la présence d'un support physique et la décomposition induite par ce dernier sur les performances et le confort des utilisateurs. En effet, la mise en place et l'évaluation de solutions spécifiques sont difficilement généralisables à un autre ensemble de tâches.

1.6 Conclusion

Nous avons proposé une revue de la majeure partie des applications de modélisation existantes en environnement de RV. Au travers de l'analyse des techniques d'interaction habituellement employées dans ce type d'application, nous avons pu constater que :

- Les techniques utilisées sont généralement basées sur une manipulation libre de toutes contraintes. Cette situation d'interaction offre un cadre de manipulation direct et intuitif. Les utilisateurs ont la possibilité de manipuler des entités géométriques 3-D directement dans un espace 3-D ;
- La liberté de manipulation introduit de nouvelles difficultés comme l'appréhension d'une nouvelle dimension dans l'interaction, la profondeur. Ces difficultés semblent limiter les performances des utilisateurs. Les concepteurs ont alors recourt à de nouveaux outils afin d'essayer de surmonter ces difficultés. L'analyse des outils proposés jusqu'à présent indique que les contraintes virtuelles et physiques sont les solutions les plus répandues.

Nous proposons d'introduire des contraintes physiques au travers de la simulation d'une interface tactile. Les interfaces tactiles semblent constituer une solution intéressante aux problèmes de manipulation en environnement de RV. Toutefois, notre objectif n'est pas de proposer et d'évaluer des solutions à des problèmes spécifiques liés à la modélisation géométrique. Nous proposons un cadre théorique permettant de caractériser la situation d'interaction en tenant compte, non seulement de la technique d'interaction, mais également de la tâche effectuée. Nous voulons comprendre l'influence respective de l'introduction d'un support physique et de la décomposition sur des classes de tâches fondamentales. Les résultats obtenus seront alors généralisables à d'autres contextes applicatifs. Nous présentons ce contexte théorique dans le chapitre suivant.

Contexte théorique : degrés de liberté et interaction 3-D

Sommaire

2.1 Modèles connexes	32
2.1.1 Degrés de liberté et dimensions	32
2.1.2 <i>Control - Display Space Mapping</i>	33
2.1.3 Structure perceptuelle	35
2.2 Interaction et degrés de liberté	38
2.2.1 Technique d'interaction	39
2.2.2 Caractérisation de la tâche	43
2.2.3 Conclusion	46

Dans ce chapitre, nous proposons un modèle descriptif¹ dont l'objectif est de caractériser les différentes composantes intervenant dans la définition d'une technique d'interaction. La finalité de ce modèle est de comprendre, de manière plus pragmatique, les caractéristiques de la technique d'interaction ainsi que l'influence de ces caractéristiques sur la tâche effectuée. Avant de décrire le modèle que nous proposons, nous nous intéressons aux différents contextes théoriques existants. Que se soit sur des stations de travail classiques ou dans des environnements de RV, différents cadres théoriques ont été proposés dans le but de décrire l'ensemble des caractéristiques de l'interaction qui interagissent avec la manipulation des différents DDL de la tâche.

1. Les modèles descriptifs sont des modèles qui ne reposent pas sur des mesures quantitatives de performances. Ce type de modèle permet essentiellement de poser un contexte permettant de caractériser une problématique et de fournir un support de réflexion. C'est le cas du *3-state model* proposé par Buxton *et al.* [Buxton 90]. Ce dernier permet de décrire les différentes opérations associées à un périphérique d'interaction sous forme d'un automate donné en Figure 2.1.

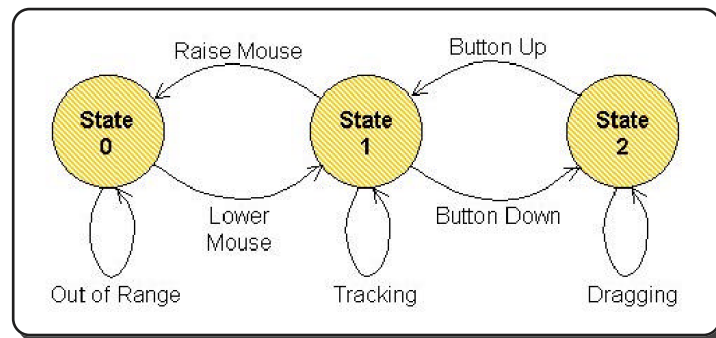


FIGURE 2.1 – L'automate représentant les différents états ainsi que les transitions entre les différents états de la souris. Image extrait de [Buxton 90].

2.1 Modèles connexes

De nombreux modèles tentent de décrire les aspects fondamentaux de l'Interaction Homme - Machine (noté IHM). Ils proposent notamment d'analyser de façon empirique les périphériques ainsi que les techniques d'interaction utilisées. Nous présentons dans cette section, les modèles descriptifs existants qui nous serviront de base afin de proposer un nouveau modèle s'articulant autour de la notion de DDL.

Nous ne mentionnons pas les modèles que l'on pourrait qualifier de plus haut niveau tel que *Goals, Operators, Methods, and Selection rules* [John 96] ou encore *Executive – Process / Interactive Control* [Kieras 97]. Ces modèles ont pour but de modéliser et d'analyser des systèmes interactifs dans leur globalité. Ils ne permettent pas de prendre en compte des phénomènes bas niveaux (par exemple le nombre de DDL à manipuler pour effectuer la tâche, les caractéristiques des DDL, etc.).

2.1.1 Degrés de liberté et dimensions

Le premier modèle, appelé *Model for mapping degrees of freedom to dimensions*, a été proposé par MacKenzie *et al.* [MacKenzie 03]. Dans leur modèle, ils opposent deux notions : les **degrés de liberté** et les **dimensions**. Les DDL sont des paramètres intrinsèques associés aux périphériques d'interaction. Par exemple, la souris est considérée, dans ce modèle, comme un périphérique à trois DDL :

1. Un DDL en translation pour les mouvements gauche – droite ;
2. Un DDL en translation pour les mouvements avant – arrière ;
3. Un DDL en rotation, noté θ_z sur la Figure 2.2, qui représente la rotation de la souris autour de l'axe vertical.

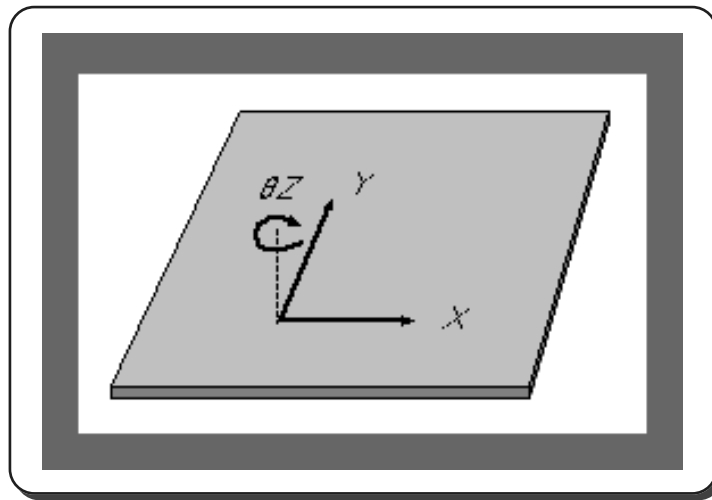


FIGURE 2.2 – Les trois DDL de la souris. Image de [MacKenzie 03].

Les dimensions, quant à elles, sont en quelque sorte les DDL associés à la tâche à effectuer. Prenons l'exemple de l'utilisation de la souris pour déplacer un curseur sur un écran (tâche à deux DDL). Dans ce cas de figure, les deux DDL en translation sont utilisés afin de manipuler les deux dimensions de la tâche. Dans ce cas, ce modèle considère que le DDL θ_z est ignoré.

Ce modèle permet principalement de caractériser les périphériques d'interaction. Toutefois, il considère de manière indépendante les DDL du périphérique et les dimensions de la tâche. En particulier, nous pouvons noter qu'il ignore la relation existant entre les DDL et les dimensions. Or, la correspondance définie, par la technique d'interaction, entre les DDL et les dimensions, peut avoir une influence sur les performances de l'utilisateur ou encore sur le confort d'utilisation [Keijser 07].

2.1.2 Control - Display Space Mapping

Afin de pallier cette lacune, Keijser *et al.* [Keijser 07] proposent un modèle (appelé *Control - Display Space Mapping*) dans lequel ils considèrent deux espaces de travail :

- **L'espace de contrôle** : il correspond à l'espace de travail du périphérique d'interaction. L'utilisation d'une souris fournit un espace de contrôle à deux dimensions, défini par le plan dans lequel la souris peut être déplacée ;
- **L'espace d'affichage** : il décrit l'espace dans lequel est effectuée la tâche. Positionner un objet dans une scène 3-D consiste à manipuler dans un espace d'affichage de dimension 3.

Le cadre théorique qu'ils proposent essaye de caractériser la correspondance entre ces deux espaces. Elle est définie par une fonction. Dans le cas général, si l'espace de contrôle est de dimension n et l'espace d'affichage de dimension m , alors la fonction est de type :

$$f : \mathbb{R}^n \mapsto \mathbb{R}^m$$

Dans le cas du positionnement 3-D d'un objet à l'aide d'un périphérique à trois DDL, la fonction de correspondance sera du type :

$$f : \mathbb{R}^3 \mapsto \mathbb{R}^3$$

Cette définition permet de prendre en compte de nombreux types de phénomènes comme le *Control / Display Gain*² (noté CDG) [MacKenzie 94]. Ce modèle permet également de différencier les types de périphériques utilisés.

Keijser *et al.* s'intéressent notamment au lien existant entre la correspondance des différentes dimensions – définies par la technique d'interaction – et les performances des utilisateurs. Dans une première étude, ils montrent les différences de performances qui peuvent apparaître lorsque la correspondance entre les paramètres de l'espace de contrôle et les paramètres de l'espace d'affichage est modifiée. Dans leur étude, ils comparent les performances obtenues dans une situation où un effet miroir est activé par rapport à une situation de référence où aucune transformation n'est appliquée. L'effet miroir est illustré en Figure 2.3. Les performances obtenues lorsque l'effet miroir est activé sont moindres.

Ces questions sont en lien étroit avec la théorie de la *Stimuli - Response Compatibility*³ (notée SRC) introduit par Fitts et Seeger [Fitts 53]. Dans le contexte de l'IHM, la théorie de la SRC établit que le niveau de difficulté associé à une tâche dépend de la complexité de la relation entre l'ensemble des stimuli (par exemple fournis par l'utilisateur au travers d'un périphérique d'interaction) et les réponses correspondantes (par exemple données par le système d'information au travers d'un retour visuel). Les stimuli peuvent être assimilés aux DDL accessibles au travers du périphérique d'interaction et la réponse correspond au retour visuel associé à la tâche effectuée.

Toutefois, le modèle proposé par Keijser *et al.* ne rend pas compte des caractéristiques de la tâche. En effet, deux tâches peuvent nécessiter la manipulation d'un même nombre

2. Le CDG définit la relation d'échelle existant entre les mouvements du périphérique et les mouvements de l'objet manipulé. Le CDG peut être constant, dans ce cas, les mouvements de l'objet sont ceux du périphérique à un facteur multiplicatif constant près. Le CDG peut également être une fonction de la vitesse du périphérique comme par exemple dans la technique PRISM [Frees 05].

3. En français, *compatibilité du stimulus et de la réponse*

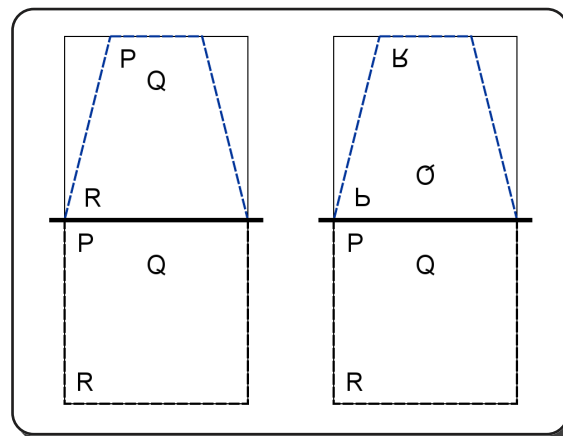


FIGURE 2.3 – À gauche, le résultat de la correspondance entre les positions de la souris et les positions du curseur lorsque l'effet miroir n'est pas appliqué. À droite de l'image, le résultat de la correspondance lorsque l'effet miroir est appliqué. Image extraite de [Keijser 07].

de DDL sans pour autant être de la même nature. C'est le cas, par exemple, pour les tâches de positionnement 3-D et de suivi de trajectoires en 3-D. Bien que les deux tâches nécessitent la manipulation simultanée de trois DDL, la nature de la tâche est différente et leur réalisation ne nécessite pas le même type de contrôle.

2.1.3 Structure perceptuelle

Mettre en avant la supériorité d'un périphérique ou d'une d'interaction pour une tâche particulière n'est pas suffisant. Il est plus intéressant d'en comprendre les raisons afin de pouvoir créer plus facilement des périphériques ainsi que des techniques qui soient en adéquation avec les besoins de l'utilisateur. Le modèle introduit par Jacob *et al.* [Jacob 94, Jacob 92] permet d'expliquer les différences de performances offertes, pour une même tâche, par différents périphériques d'interaction.

Dans leur modèle, ils présentent la notion d'*integrality* (ou intégrabilité) et de *separability* (ou séparabilité) des périphériques d'interaction et des tâches. La notion d'**attribut** (similaire à la notion de DDL) permet de décrire les caractéristiques d'un périphérique. Un périphérique est constitué d'un ensemble d'attributs : position, orientation, états des boutons, *etc.* Les éléments de l'interface ou encore les tâches sont également constitués d'attributs (position et orientation d'un objet manipulé par exemple).

Chaque attribut est plus ou moins **similaire** aux autres. Ainsi les attributs de position horizontale et de position verticale possèdent un fort degré de similarité car la relation sémantique entre ces deux attributs peut être considérée comme forte. Au contraire, les

attributs de couleur et d'orientation, ayant une relation sémantique moins importante, sont considérés comme ayant un faible degré de similarité. Il existe donc une métrique permettant de décrire cette **distance de similarité**. Cette métrique intègre notamment des notions liées à la sémantique des attributs.

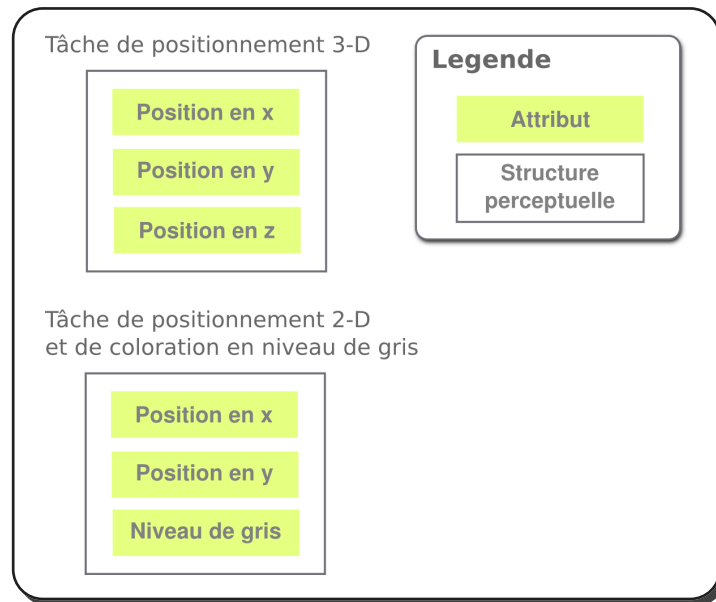


FIGURE 2.4 – La structure perceptuelle regroupe l'ensemble des attributs de la tâche ou du périphérique d'interaction.

Dans leur modèle, Jacob *et al.* considèrent que tous les attributs peuvent être regroupés en une **structure perceptuelle**. Ceci est illustré en Figure 2.4. La structure perceptuelle peut être **intégrable** ou **séparable**. Lorsque la structure perceptuelle est intégrable, l'utilisateur ne perçoit pas chaque attribut de la structure comme des éléments indépendants. Il perçoit les attributs de la tâche ou du périphérique comme un ensemble indissociable. À l'inverse, lorsque la structure perceptuelle d'une tâche ou d'un périphérique est séparable, l'utilisateur peut regrouper les attributs dans différents groupes. Les attributs au sein d'un même groupe sont similaires. En Figure 2.5, nous donnons l'exemple de deux tâches, la première ayant une structure perceptuelle intégrable, la seconde ayant une structure perceptuelle séparable.

Du point de vue de l'interaction, lorsque l'utilisateur effectue une tâche ayant une structure perceptuelle intégrable, il aurait tendance à manipuler l'ensemble des attributs simultanément. À l'opposé, il est difficile pour l'utilisateur de combiner dans un même geste de commande, l'ensemble des attributs d'une tâche ou d'un périphérique ayant une structure perceptuelle séparable.

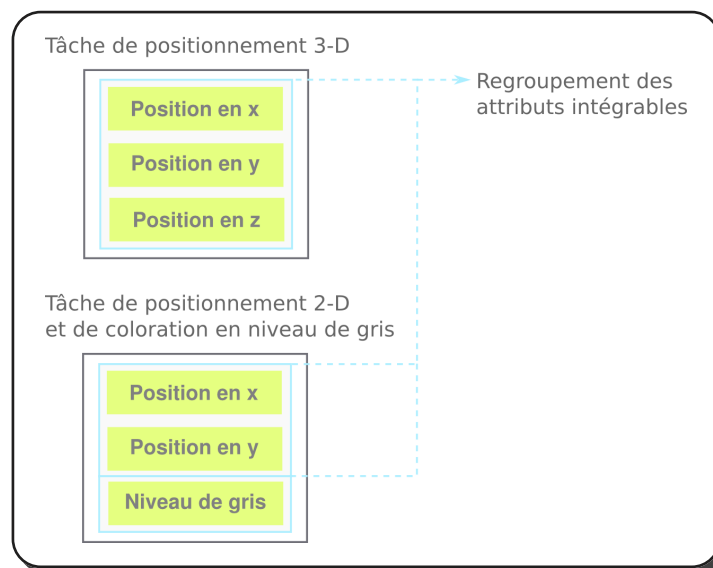


FIGURE 2.5 – Pour la tâche de positionnement 3-D, la sémantique des différentes dimensions étant très proche, les trois attributs peuvent être regroupés en un seul groupe d’attribut. La structure de cette tâche est à priori intégrable. Pour la seconde tâche, les attributs peuvent être regroupés en deux groupes distincts. La structure de la tâche est à priori séparable.

Bien que ce ne soit pas mentionné dans le modèle proposé par Jacob *et al.*, nous pensons que le regroupement d’un ensemble d’attributs fait intervenir une notion qui pourrait s’apparenter à une **intégration conceptuelle des attributs**⁴.

Jacob *et al.* ont étudié la relation existant entre la structure perceptuelle des périphériques d’interaction et la structure perceptuelle de la tâche effectuée. Dans leur étude, ils comparent les performances obtenues dans deux tâches :

1. Une tâche considérée comme ayant une structure perceptuelle intégrable : positionnement 2-D d’un objet sur une cible combinée à une tâche consistant à faire correspondre la taille de l’objet manipulé à la taille de la cible ;
2. Une tâche considérée comme ayant une structure perceptuelle séparable : positionnement 2-D d’un objet sur une cible combinée à une tâche consistant à faire correspondre le niveau de gris de l’objet manipulé au niveau de gris de la cible.

Les utilisateurs effectuent ces deux tâches en utilisant un capteur magnétique – structure perceptuelle intégrable, les trois attributs de positions étant intégrés dans un même

4. Cette notion est inspirée de l’**intégration conceptuelle des tâches** que nous retrouvons dans la théorie des interactions à deux mains [Guiard 87]. Lors d’une interaction à deux mains (symétrique ou asymétrique), on dit qu’il y a intégration conceptuelle des tâches lorsque l’utilisateur intègre les deux tâches effectuées par les deux mains dans une seule et unique tâche.

groupe d'attributs – et une souris avec molette – structure perceptuelle séparable, la molette étant considérée comme un attribut isolé des attributs de position de la souris. Les résultats montrent que les performances obtenues dans la première tâche sont meilleures avec le capteur magnétique qu'avec la souris. Dans la seconde tâche, les performances obtenues sont meilleures avec la souris qu'avec le capteur. Cela suggère que pour obtenir les meilleures performances, les structures perceptuelles de la tâche et du périphérique d'interaction doivent être identiques.

Ce modèle permet de tenir compte de la relation sémantique qui existe entre les attributs du périphérique et les attributs de la tâche. Toutefois, ce modèle ne tient pas compte de la charge cognitive liée à la manipulation simultanée de nombreux attributs. En effet, lorsque le nombre d'attributs à manipuler devient trop important, ce modèle ne tient pas compte de la capacité ou de l'incapacité des utilisateurs à les combiner. Une étude a été menée par Wang *et al.* [Wang 98] dans le cadre d'une tâche de *docking*. Cette tâche nécessite la manipulation de six DDL, trois DDL en orientation et trois DDL en translation. L'ensemble de ces DDL, étant à priori intégrables, l'utilisateur doit donc être capable de les combiner lors de la réalisation de la tâche. Or, Wang *et al.* montrent que les utilisateurs tendent à séparer le positionnement et l'orientation de l'objet. Ce phénomène peut être dû à la surcharge cognitive induite par la manipulation simultanée des six DDL.

De plus, la notion d'intégrabilité et séparabilité dépend de la similarité entre deux attributs. Or, la quantification de la similarité est un processus subjectif. Elle est donc potentiellement sujette à une forte variabilité entre les utilisateurs. Pour un même couple d'attributs, il est possible que certains utilisateurs puissent les percevoir comme étant intégrables, tandis que d'autres les définissent comme séparables.

De surcroît, ce modèle bien que suggérant l'existence d'une relation entre DDL du périphérique et DDL de la tâche, ne s'intéresse pas à l'influence de la relation entre les attributs (relation linéaire, non-linéaire, mise en relation d'un attribut de position avec un attribut d'orientation, *etc.*) sur les performances des utilisateurs. Ces relations sont souvent altérées par la technique d'interaction. Par exemple, la technique du Go-Go introduit une relation non-linéaire entre les attributs de position du périphérique et de la tâche. Par rapport à une technique comme la main virtuelle, la modification de cette relation peut altérer la représentation mentale (*i.e.* la structure perceptuelle) que peut se faire l'utilisateur de la tâche.

2.2 Interaction et degrés de liberté

Les modèles existants permettent d'avoir une vision pragmatique de certains mécanismes intervenant au cours de l'interaction. Nous proposons de réunir plusieurs de ces

modèles au sein d'un même contexte théorique. Notre modèle, nous le construisons autour de la notion de **manipulation de degrés de liberté**. Ceci nous permet de caractériser la situation d'interaction en tenant compte des propriétés de la technique. Nous proposons d'affiner cette caractérisation en tenant compte des propriétés de la tâche effectuée.

Dans un premier temps, nous nous intéressons aux propriétés de la technique d'interaction. Pour cela, nous considérons la technique d'interaction comme un composant abstrait qui va permettre de modifier la relation existant entre les DDL associés au périphérique d'interaction et les DDL associés à la tâche à effectuer. Cela nous permet de prendre en compte différents types de tâches en conceptualisant le problème de la manipulation en environnement virtuel à la manipulation de DDL. En utilisant cette notion, nous pouvons nous positionner à un niveau d'abstraction permettant de prendre en compte des phénomènes apparaissant à un bas niveau, comme l'introduction de contraintes virtuelles ou de contraintes physiques.

2.2.1 Technique d'interaction

Nous proposons, dans un premier temps, d'abstraire à la fois le périphérique d'interaction et la tâche comme deux ensembles de DDL distincts. Un capteur optique est considéré comme un ensemble de six DDL. Cet ensemble peut être utilisé afin d'effectuer des tâches d'interaction. Ces tâches peuvent également être considérées comme un ensemble de DDL. D'une manière similaire au modèle proposé par Keijser *et al.* [Keijser 07], nous considérons alors la technique d'interaction comme un composant abstrait qui va établir la relation existant entre les DDL du périphérique d'interaction et les DDL de la tâche. Nous notons DDL_p les DDL du périphérique d'interaction et DDL_t les DDL de la tâche.

Dans notre modèle, nous distinguons plusieurs composantes dans la relation des DDL :

- Les **contraintes physiques** : il s'agit de contraintes qui apparaissent au niveau du contrôle moteur du geste. Les gestes sont alors contraints physiquement par le périphérique (par exemple un périphérique à retour d'effort contraignant le point d'interaction sur la surface d'un objet en cours d'édition) ou par des éléments physiques présents dans l'environnement (par exemple la surface d'un écran tactile) ;
- La **fonction de contrainte** : il s'agit généralement de filtres qui vont ignorer certains DDL au cours de la réalisation d'une tâche. Dans ce cas, le mouvement n'est pas obligatoirement contraint physiquement ;
- La **fonction de correspondance** : pour un DDL_t , la fonction de correspondance va définir quel est le DDL_p qui va permettre de le contrôler ;
- La **fonction de transfert** : elle définit le CDG pour chaque DDL_p .

Ceci est illustré sur la Figure 2.6. De nombreuses études déjà menées se sont intéressées à l'influence de la fonction de correspondance et de la fonction de transfert [Casiez 08] sur les performances des utilisateurs. Nous discuterons de l'influence respective de ces différentes composantes par la suite.

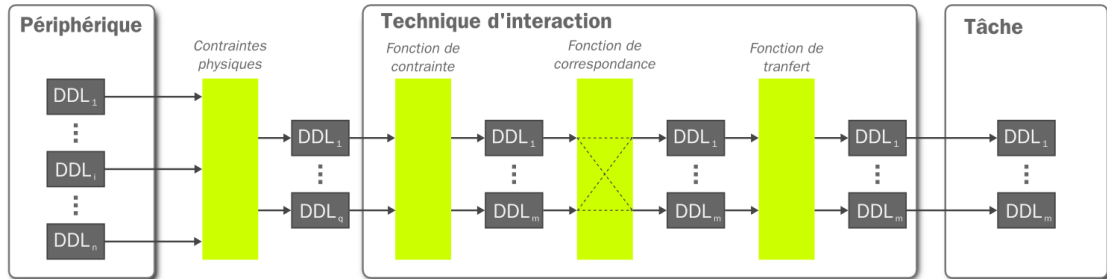


FIGURE 2.6 – Le périphérique d'interaction et la tâche à effectuer sont considérés comme deux ensembles de DDL. La technique d'interaction va définir la relation entre les DDL du périphérique et les DDL de la tâche.

Dans le cadre de cette thèse, nous allons essentiellement nous focaliser sur les contraintes physiques ainsi que sur la fonction de contrainte virtuelle. Dans un souci de complétude, nous discutons également de la fonction de correspondance et de la fonction de transfert mais elles ne font pas, ici, l'objet d'une étude approfondie.

Fonction de correspondance

La fonction de correspondance permet de définir la relation qu'il existe entre les différents DDL que nous retrouvons au niveau du périphérique d'interaction et de la tâche. Plus particulièrement, pour un DDL_t , elle définit quel est le DDL_p qui va permettre de le contrôler. Plaçons nous dans le cas de l'orientation d'un objet dans un espace 3-D à l'aide d'un Wand, périphérique à six DDL. L'orientation 3-D nécessite la manipulation de trois DDL en rotation. La fonction de correspondance va permettre deux choses :

1. Définir quels sont les DDL du Wand qui vont être effectivement utilisés afin d'effectuer la tâche. Le Wand possédant plus de DDL que la tâche, certains d'entre eux vont être ignorés ;
2. Définir parmi les DDL restants, lesquels vont permettre de modifier les trois paramètres d'orientation de l'objet.

La fonction de correspondance la plus simple consiste à faire correspondre au DDL de rotation en x (respectivement y et z) du périphérique d'interaction, le DDL de rotation x (respectivement y et z) de la tâche de rotation. Généralement, des DDL en rotation (respectivement translation) du périphérique d'interaction sont associés à des DDL en rotation (respectivement translation) de la tâche à effectuer. Toutefois, ce n'est

pas toujours le cas. Sur une station de travail, lorsque l'utilisateur oriente un objet dans un espace 3-D à l'aide d'une souris, les mouvements en translation du périphérique permettent de modifier des DDL en rotation de la tâche.

Les questions soulevées par la fonction de correspondance rejoignent l'ensemble des travaux en lien avec la SRC [Fitts 53]. Dans ces travaux, les auteurs s'intéressent aux possibles liens entre les performances des utilisateurs et la fonction de correspondance. Prenons l'exemple d'une tâche d'orientation d'un objet autour de l'axe x . Plusieurs correspondances sont possibles. Parmi ces correspondances, nous considérons les deux fonctions de correspondance suivantes :

1. Les mouvements avant - arrière de l'objet permettent de tourner l'objet autour de l'axe x ;
2. Les mouvements gauche - droite de l'objet permettent de tourner l'objet autour de l'axe x .

Intuitivement, nous pouvons penser que la première correspondance est plus adaptée et peut conduire à de meilleures performances que la seconde. L'objectif des travaux sur la SRC [Schmandt 83] [John 85] [Worringham 98] consiste à comprendre les mécanismes intervenant lors de la mise en correspondance des différents DDL, afin d'identifier des moyens permettant de définir la correspondance la plus adaptée entre un périphérique et une tâche d'interaction.

Fonction de transfert

La fonction de transfert modifie de manière linéaire ou non, la relation entre les DDL_p et les DDL_t . L'utilisation de fonctions de transfert est largement répandue et étudiée dans le domaine de l'IHM, en général via l'utilisation du CDG.

Les premières techniques utilisées dans les environnements de RV sont basées sur des interactions inspirées du monde réel. C'est le cas par exemple de la main virtuelle qui permet de manipuler directement les objets dans l'environnement virtuel. Toutefois, cette technique ne permet à l'utilisateur de manipuler que des objets se trouvant à portée de main. Afin de pouvoir manipuler des objets virtuels se trouvant hors de portée des utilisateurs, Poupyrev *et al.* [Poupyrev 96] ont mis en place la technique du Go-Go. Cette technique consiste à appliquer une fonction non-linéaire à la position du périphérique d'interaction afin de pouvoir désigner et manipuler des objets se trouvant à de grandes distances de l'utilisateur. Une illustration est donnée en Figure 2.7.

Le problème s'est également posé pour la tâche de rotation. En utilisant une technique de manipulation directe, l'utilisateur saisit l'objet et l'oriente directement avec la main. Les nombreuses contraintes anatomiques auxquelles nous sommes soumis ne permettent pas à l'utilisateur d'effectuer, en un seul geste, une rotation de 360 degrés

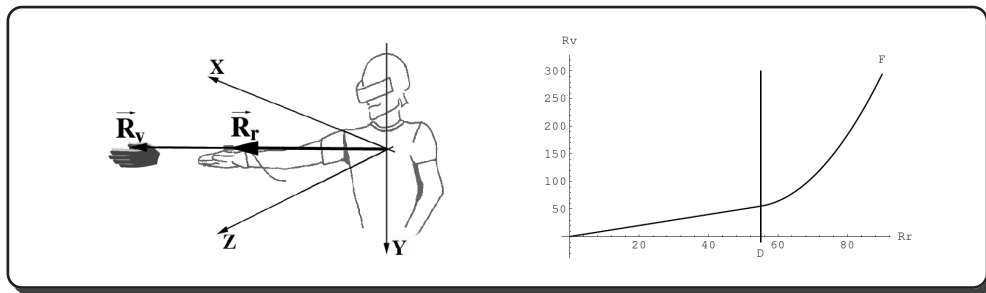


FIGURE 2.7 – La technique du Go-Go est une extension de la main virtuelle. La position de la main virtuelle n'est plus définie comme la position du périphérique d'interaction, mais comme une fonction non-linéaire de la position du périphérique. Sur la droite de l'image, nous retrouvons un exemple de fonction non-linéaire. Image extraite de [Poupyrev 96].

autour d'un axe quelconque. Afin de pallier à cette difficulté, une technique similaire à la technique du Go-Go, appelée **rotations non-isomorphiques**, a été proposée par Poupyrev *et al.* [Poupyrev 99] [Poupyrev 00]. Cette technique consiste à amplifier les rotations appliquées à l'objet manipulé afin que l'utilisateur puisse tourner l'objet d'un angle supérieur à 360 degrés en un seul geste.

Il existe d'autres fonctions de transfert, introduisant un gain constant entre les DDL_p et les DDL_t ou introduisant des fonctions plus complexes comme la fonction de transfert utilisée dans la technique PRISM [Frees 05] où le gain est un facteur dépendant de la vitesse du périphérique.

L'utilisation de fonctions de transfert peut se faire dans des contextes très variés et peut permettre de combler des lacunes importantes comme le manque de précision. Son influence a été étudiée dans de nombreux travaux, notamment dans le cadre de la tâche de sélection 2-D [Casiez 08]. Toutefois, il semblerait qu'elles ne permettent pas une augmentation très significative de la précision des utilisateurs. Dans une étude très complète de Casiez *et al.* [Casiez 08], nous pouvons voir qu'une mauvaise fonction de transfert peut avoir des conséquences négatives sur les performances en réduisant la précision des utilisateurs. C'est le cas lorsque le CDG est trop important, rendant ainsi le contrôle trop imprécis. Au contraire, lorsque le CDG est trop faible, il augmente de manière considérable le nombre de mouvements nécessaires à la réalisation de la tâche.

Contraintes physiques et virtuelles

Au travers du périphérique d'interaction, le nombre de DDL manipulés simultanément peut avoir une influence importante sur les performances de l'utilisateur. Comme nous avons pu le voir dans le chapitre précédent, de nombreux travaux en RV, que se

soit dans le cadre d'applications généralistes ou spécialisées dans la modélisation géométrique, s'intéressent à l'introduction de contraintes physiques ou virtuelles au cours de l'interaction.

Nous illustrons l'utilisation de contraintes virtuelles (introduit par la **fonction de contrainte**) et physiques dans une tâche de positionnement 3-D :

- Positionnement sans contraintes : l'utilisateur saisit et déplace l'objet dans l'environnement 3-D directement. Avec cette technique, l'utilisateur peut manipuler l'ensemble des trois DDL de la tâche simultanément ;
- Positionnement avec contraintes virtuelles : l'utilisateur peut déplacer dans un plan l'objet sélectionné (manipulation dans un espace 2-D). La manipulation de la troisième dimension peut se faire à part, en se mettant dans un mode particulier de positionnement ;
- Positionnement avec contraintes physiques : l'utilisateur peut déplacer dans un plan l'objet sélectionné (manipulation dans un espace 2-D) en touchant l'écran du Workbench se trouvant à portée avec la main dominante. La manipulation de la troisième dimension peut se faire en touchant l'écran avec la main non dominante.

Plus le nombre de DDL à manipuler simultanément est important plus la complexité de la tâche croît. Il semblerait qu'il en découle une augmentation de la charge cognitive ainsi qu'une baisse des performances notamment en matière de précision. Toutefois, la diminution de la dimensionnalité de l'espace de la technique d'interaction, que se soit à l'aide de contraintes physiques ou de contraintes virtuelles, peut nécessiter de décomposer la tâche. En effet, effectuer une tâche à m DDL nécessite que l'utilisateur puisse manipuler individuellement ou simultanément ces m DDL. À l'aide d'un périphérique à m DDL, l'utilisateur peut manipuler les m DDL_{*i*} simultanément et ainsi effectuer la tâche en un seul geste. En introduisant des contraintes, l'utilisateur peut alors manipuler au plus $m - 1$ DDL et la tâche doit être décomposée en au moins deux sous-tâches : une tâche à $m - 1$ DDL et une tâche à 1 DDL. L'autre extrême consisterait à décomposer une tâche à m DDL en m sous-tâches, chacune de dimensionnalité un.

Cette décomposition, au-delà des modifications en matière de contrôle et de réalisation de la tâche, peut avoir une influence sur la perception et la conceptualisation de la tâche.

2.2.2 Caractérisation de la tâche

Tous les éléments que nous avons présenté jusqu'à présent ne tiennent compte que de phénomènes intervenant au niveau moteur (*i.e.* dans la partie **action** de la **boucle perception - cognition - action** [Fuchs 06]). Les caractéristiques de la technique d'interaction peut avoir une influence sur la perception ainsi que sur la stratégie mise en place lors de la réalisation de la tâche.

Intégration conceptuelle des DDL

Les techniques d'interaction suggérant une décomposition de la tâche peuvent altérer la façon dont l'utilisateur va conceptualiser la tâche. Par exemple, la technique *Balloon Selection* [Benko 07] consiste à déplacer un curseur dans un plan horizontal. L'utilisateur peut également modifier la position du plan en hauteur. Cette technique suggère une décomposition du positionnement du point de sélection : positionnement dans le plan horizontal et positionnement du plan en hauteur. Bien que la tâche de positionnement du point de sélection soit une tâche 3-D potentiellement intégrable, la technique d'interaction peut altérer la représentation mentale de la tâche. Avec la technique *Balloon Selection*, la tâche de positionnement peut être perçue comme une tâche à structure séparable constituée de deux sous-tâches.

Il semble donc indispensable que notre modèle puisse être capable de mettre en avant la capacité que peut avoir la technique d'interaction à modifier la structure perceptuelle de la tâche. Cette propriété se matérialise par une **fonction d'intégration conceptuelle des DDL**. La notion d'**intégration conceptuelle** est une notion théorique qui s'illustre initialement dans le cadre des interactions bi-manuelles [Guiard 87]. Lorsque l'utilisateur utilise les deux mains dans le but d'effectuer une tâche, le rôle des deux mains durant l'interaction peut-être identique, on parle alors d'**interaction symétrique**, ou différent, on parle dans ce cas d'**interaction asymétrique**. Quel que soit le type d'interaction, effectuer une tâche T en utilisant les deux mains conduit à exécuter deux sous-tâches, T_1 et T_2 , une pour chaque main. Ces tâches peuvent être exécutées de manière séquentielles ou parallèles. On dit qu'il y a intégration conceptuelle de T_1 et T_2 , lorsqu'au cours de l'interaction, l'utilisateur n'a pas conscience des deux sous-tâches. Il ne perçoit que l'exécution d'une seule tâche, la tâche T . Dans ce cas, l'utilisateur est capable de réaliser les deux sous-tâches simultanément. Au contraire, lorsqu'il n'y a pas intégration conceptuelle, les deux sous-tâches sont réalisées séquentiellement.

Cette notion permet de caractériser la perception que peut avoir l'utilisateur de la tâche en cours de réalisation. La fonction d'intégration conceptuelle des DDL suppose qu'il existe un niveau d'intégration conceptuelle que l'on pourrait considérer de plus bas niveau. Avant d'intégrer conceptuellement des sous-tâches, il est possible qu'il existe une intégration conceptuelle des DDL pour chacune des tâches. La coordination des DDL d'une tâche d'interaction, comme le positionnement, nécessite alors d'intégrer conceptuellement les DDL de la tâche, en l'occurrence x , y et z .

Cette fonction, intervenant à un niveau cognitif, définit la structuration mentale de la tâche. Elle se matérialise par la fonction suivante :

$$FIC : E \rightarrow (S_1, \dots, S_n) \quad (2.1)$$

$E, S_1, \dots, S_n$ sont des multiensembles⁵. E est un multiensemble de DDL (par exemple pour une tâche de positionnement $E = (\{x, y, z\}, m)$), $\forall i \in [1, n], S_i \in (P(E), m) \setminus (\emptyset, m)$ où $P(E)$ désigne l'ensemble de tous les sous-ensembles possibles de E , également appelée ensemble des parties de E (par exemple $\{\{x\}, \{y\}, \{z\}, \{x, y\}, \{x, z\}, \{y, z\}, \{x, y, z\}\}$). Il faut que $\forall e \in E, \sum_{i=1}^n m_i(e) = 1$ où m_i est la multiplicité du multiensemble S_i . n définit le nombre de groupe de DDL. Cette contrainte permet simplement de s'assurer que l'ensemble des éléments de E (i.e. l'ensemble des DDL) apparaissent une seule fois dans l'ensemble des S_i .

Illustrons la définition de la fonction FIC sur la technique *Balloon Selection*. La fonction FIC peut être définie de la manière suivante :

$$FIC : (\{x, y, z\}, m) \rightarrow \{(\{x, z\}, m), (\{y\}, m)\} \quad (2.2)$$

Par la suite, hors du cadre des études menées dans ces travaux, nous souhaitons comprendre les mécanismes intervenant dans la définition de la fonction FIC. Ceci permettrait de mettre en place plus facilement des techniques d'interaction adaptées aux besoins et aux difficultés des utilisateurs. Ces mécanismes sont similaires à ceux intervenant dans la théorie de Jacob *et al.* [Jacob 94] sur la structure perceptuelle. Toutefois, dans ces travaux, il n'est fait mention d'aucune piste concernant les facteurs intervenant dans la construction des structures perceptuelles. Nous pensons que parmi eux, se trouvent des facteurs liés à la sémantique des DDL ainsi que des facteurs liés à la difficulté d'appréhension des différents DDL. Par exemple, dans le cas du positionnement 3-D, les trois DDL ont un lien sémantique fort. Toutefois, il est possible que la dimension z , plus difficile à appréhender que les autres, soit isolée de x et y . En effet, de nombreuses études font mention de difficultés spécifiques liées à cette dimension [Fiorentino 03].

Contrôle associé à la tâche

Dans notre modèle, nous proposons également de caractériser plus finement la tâche que simplement par sa dimensionnalité. Il est possible que le niveau de difficulté associé à une tâche soit proportionnel au nombre de DDL qu'elle requiert de manipuler. Toutefois, il est possible qu'à dimensionnalité égale, deux tâches soient de difficultés différentes. Considérons la tâche de positionnement ainsi que la tâche de suivi de trajectoires dans un espace 3-D. Ces deux tâches nécessitent de manipuler dans un espace 3-D, elles sont donc de dimensionnalité 3. Néanmoins, de part leur nature, ces deux

5. Un multiensemble est un couple (E, m) . E est un ensemble appelé le **support**. $m : E \rightarrow \mathbb{N}$ est une fonction appelée **multiplicité**. Un multiensemble peut être vu comme un ensemble des éléments de E dans lequel un même élément peut apparaître plusieurs fois. Dans le multiensemble (E, m) , l'élément e apparaît $m(e)$ fois.

tâches sont différentes. Dans le cas du positionnement, la trajectoire suivie par l'objet entre sa position initiale et sa position finale importe peu à l'utilisateur. Au contraire, au cours d'une tâche de suivi de trajectoires, à chaque instant de la tâche, la position de l'objet manipulé est importante et doit être contrôlée précisément.

Que se soit en 2-D ou en 3-D, les lois permettant de décrire l'une et l'autre des tâches sont différentes. Pour le positionnement, il existe la **loi de Fitts** [Fitts 54] ainsi qu'une extension proposée pour la 3-D [Grossman 04]. Dans le cas du suivi de trajectoires, la loi utilisée intègre la largeur du chemin sur l'ensemble de la trajectoire [Accot 97] [Accot 99].

$$T_C = a + b \times \int_C \frac{ds}{W(s)}$$

a et b sont deux constantes, C désigne la trajectoire considérée, ds le chemin parcouru le long de la courbe et $W(s)$ la largeur du chemin à l'abscisse s de la courbe C . L'utilisation d'une intégrale dans la définition de la loi traduit le caractère continu de la tâche de suivi de trajectoires. Ceci n'est pas le cas pour la tâche de positionnement 3-D.

Partant de ce constat, nous proposons de distinguer les tâches en fonction du **type de contrôle** qu'elles nécessitent. Nous proposons deux types de tâches :

1. Les tâches à **contrôle final** : désignent l'ensemble des tâches (comme le positionnement ou l'orientation) où le critère de précision intervient seulement en fin de tâche. La succession des positions ou des orientations durant l'intégralité de la tâche n'a, dans ce cas, pas de réelle importance pour l'utilisateur ;
2. Les tâches à **contrôle continu** : désignent l'ensemble des tâches (comme le suivi de trajectoires) où le critère de précision intervient tout au long de la tâche. Dans ce cas, la succession des positions du périphérique d'interaction, qui va définir la trajectoire finale, intervient dans l'évaluation de la précision. C'est également le cas pour des tâches de navigation où l'utilisateur va devoir contrôler de manière continue non seulement la position de la caméra, mais également son orientation.

Ces deux types de tâches nécessitent un type de contrôle différent. Il est possible que celles-ci puissent bénéficier d'assistances spécifiques afin de permettre à l'utilisateur d'avoir un meilleur contrôle de son geste d'interaction.

2.2.3 Conclusion

Dans le modèle que nous proposons, nous caractérisons la technique d'interaction à l'aide de plusieurs caractéristiques :

1. Les contraintes physiques ;

2. La fonction de contrainte ;
3. La fonction de correspondance ;
4. La fonction de transfert ;
5. La fonction d'intégration conceptuelle des DDL ;
6. Le type de contrôle lié à la tâche.

Certaines de ces caractéristiques interviennent à un niveau moteur (par exemple les contraintes physiques ou encore la fonction de contrainte) tandis que d'autres tentent de représenter certains processus intervenant à un niveau cognitif (par exemple la fonction d'intégration conceptuelle des DDL). Notre modèle est synthétisé dans la Figure 2.8.

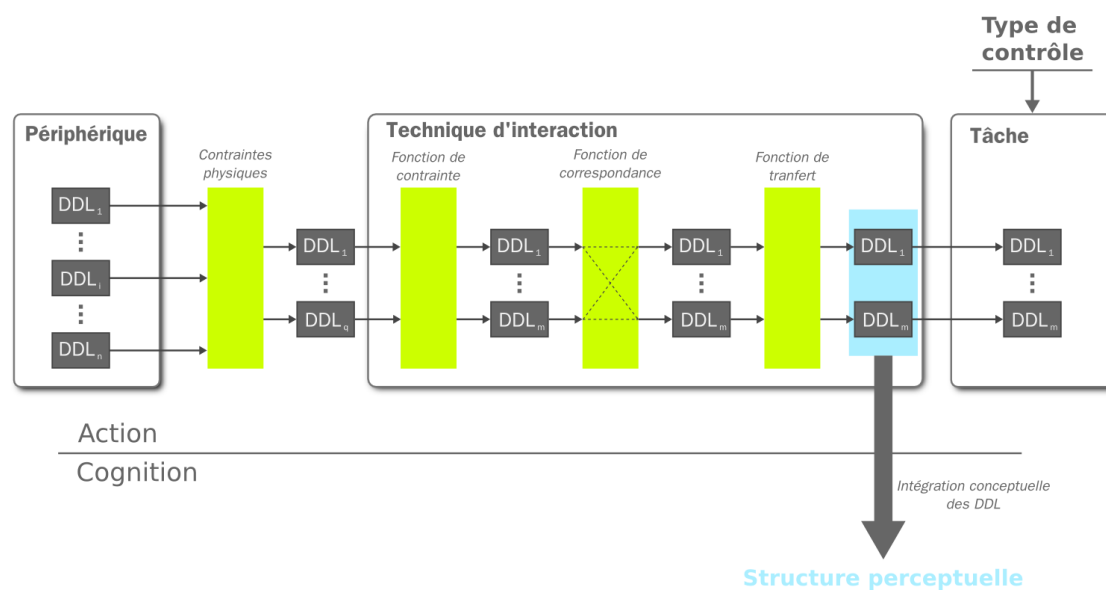


FIGURE 2.8 – Synthèse de notre modèle. Nous retrouvons l'ensemble des éléments présentés tout au long de ce chapitre.

Naturellement, de nombreuses questions se posent quant à l'influence relative de ces paramètres sur les performances de l'utilisateur. Des travaux précédemment menés semblent répondre à certaines de ces questions. C'est le cas notamment concernant l'influence de la fonction de transfert [Casiez 08] ou encore concernant la fonction de correspondance [Keijser 07]. Dans le cadre de notre travail, nous répondrons à quelques unes des questions qui se posent lorsque nous exploitons des contraintes physiques ou encore quand nous appliquons une fonction de contrainte. Il est alors nécessaire de décomposer la tâche. Nous nous intéressons notamment à l'influence de la décomposition sur les performances des utilisateurs.

Avant de proposer un ensemble de solutions basé sur l'utilisation de contraintes physiques et virtuelles, nous effectuons trois études :

- La première étude a pour objectif de comprendre le comportement des utilisateurs dans une tâche à contrôle final à trois DDL. La tâche considérée est une tâche de rotation 3-D ;
- La seconde étude a pour objectif de comprendre le comportement des utilisateurs dans une seconde tâche à contrôle final à trois DDL. La tâche considérée est une tâche de positionnement 3-D ;
- La troisième étude a pour objectif de mesurer l'apport de contraintes physiques dans une tâche de suivi de trajectoires, *i.e.* une tâche à contrôle continu à trois DDL.

Afin de mettre en oeuvre de telles études, nous avons besoin d'une métrique permettant de quantifier la coordination des DDL de la tâche. Dans le chapitre suivant, nous proposons une nouvelle métrique de coordination. Cette métrique est validée et comparée aux métriques existantes.

Une nouvelle métrique de coordination

Sommaire

3.1 Mesures existantes	50
3.1.1 Mesure d'inefficacité	51
3.1.2 $\mathcal{M}$ – metric	53
3.2 NDC : Nombre de Degrés de liberté Combinés	54
3.2.1 Exemple : cas de l'orientation 3-D	55
3.2.2 Formulation générale pour n DDL	58
3.2.3 Implémentation	60
3.2.4 Comparaison	60
3.2.5 Validation	61
3.3 Conclusion	63

Dans le domaine de l'IHM, de nombreuses mesures sont habituellement utilisées afin d'évaluer et de différencier les techniques d'interaction. Les plus couramment utilisées sont la précision d'exécution de la tâche, le temps d'exécution ou encore le taux d'erreur. Toutefois, ces mesures peuvent parfois s'avérer insuffisantes. En effet, ces dernières permettent de mettre en avant les différences de performances qui peuvent exister entre différentes techniques, mais elles ne peuvent généralement pas permettre de comprendre l'origine de ces différences.

Le développement de nouvelles mesures est alors nécessaire afin de mieux comprendre les spécificités d'un dispositif ou encore de caractériser le comportement de l'utilisateur. Dans le but de comprendre les différences de performances induites par l'utilisation de différents types de périphériques durant la tâche de désignation, MacKenzie *et al.* ont mis en place de nouvelles mesures permettant de quantifier la qualité d'un geste [MacKenzie 01]. Parmi ces mesures, nous pouvons retenir *Target Re-Entry* (dont une illustration est donnée en Figure 3.1) et *Movement Offset*. La première mesure

indique le nombre de ré-entrées du curseur dans la cible. La seconde est définie comme la distance moyenne à la trajectoire optimale. Ces mesures, dont les définitions restent très simples, ont été utilisées conjointement à des mesures de performances classiques (par exemple le taux d'erreur). Elles permettent de quantifier la qualité du contrôle du geste, tout au long de la tâche, pour la mesure *Movement Offset*, et plus particulièrement en fin de tâche, pour la mesure *Target Re-Entry*.

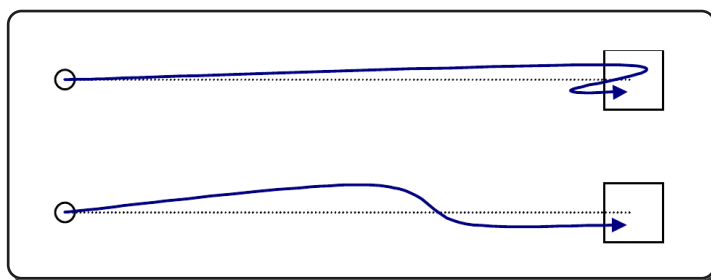


FIGURE 3.1 – Sur la première trajectoire, une occurrence de la mesure *Target Re-entry*. Sur la deuxième trajectoire, aucune occurrence de la mesure. Image extraite de [MacKenzie 01].

L'utilisation de ces nouvelles mesures a permis de mettre en avant des différences en matière de contrôle, entre des périphériques d'interaction offrant un même niveau de performance. Ces mesures ont également permis de comprendre quelles sont les caractéristiques ayant une influence sur les performances.

Dans le cadre de notre travail, nous avons dû mettre en place une nouvelle mesure afin de comprendre le comportement de l'utilisateur vis-à-vis de la manipulation des DDL. Nous nous sommes intéressés plus particulièrement à la quantification de la coordination des DDL, un facteur nous paraissant essentiel dans la caractérisation du comportement utilisateur. Nous présentons à présent les mesures existantes.

3.1 Mesures existantes

Des travaux précédents se sont intéressés à la quantification de la coordination. Le champs d'application d'une telle mesure ne se limite pas à l'étude du comportement de l'utilisateur dans un contexte d'interaction 3-D. Elle permet également de s'intéresser à des problèmes d'évaluation neurologique (par exemple pour suivre l'évolution de la maladie de Parkinson). Avec l'avènement des technologies des sciences de l'information, certains travaux proposent d'effectuer ces tâches en utilisant des technologies de *tracking* 3-D afin de collecter des données quantitatives et d'évaluer de manière objective les dommages neurologiques [Watson 97].

3.1.1 Mesure d'inefficacité

Dans le cadre de leurs travaux, Zhai *et al.* ont proposé une mesure de coordination appelée *inefficiency* ou inefficacité [Zhai 98]. Zhai *et al.* considèrent que dans une tâche impliquant la manipulation de n DDL, la trajectoire qui a la longueur la plus petite dans cet espace à n dimensions est la trajectoire correspondant à la meilleure coordination. Ils comparent différentes trajectoires dans le cas d'un positionnement 2-D illustré en Figure 3.2. Dans cette figure, la trajectoire la plus coordonnée est celle qui suit au plus près la trajectoire optimale reliant A à B.

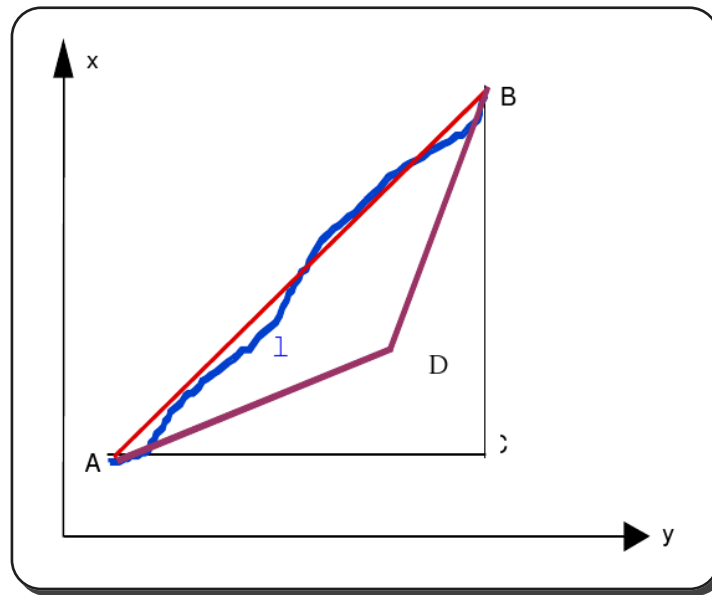


FIGURE 3.2 – Différentes trajectoires sont illustrées dans le cas d'un positionnement 2-D. Ces trajectoires correspondent à différents niveaux de coordination. Dans cette illustration, la coordination du mouvement ACB est inférieure à celle du mouvement ADB.

Partant de ce constat, toute déviation du chemin optimal est alors considérée comme le résultat d'un mouvement non coordonné et inefficace. Ils proposent alors de mesurer la coordination pour une tâche de positionnement de la manière suivante :

$$I = \frac{\text{Longueur du chemin effectué} - \text{Longueur du chemin optimal}}{\text{Longueur du chemin optimal}} \quad (3.1)$$

En utilisant cette formulation, la mesure d'inefficacité a le comportement suivant :

- Elle donne une valeur élevée pour une tâche où le geste est très peu coordonné. La valeur donnée correspond à une forte inefficacité ;

- Elle donne une valeur faible pour une tâche où le geste est coordonné. Au minimum, elle vaut zéro lorsque la longueur du chemin effectué est égale à la longueur du chemin optimal. La valeur donnée correspond à une faible inefficacité.

Toutefois, nous pouvons noter que dans certains cas, la mesure d'inefficacité peut avoir une valeur faible alors que le mouvement effectué correspond à une faible coordination. C'est le cas dans l'exemple donné en Figure 3.3. La coordination **par rapport à la tâche** est élevée, mais **dans l'absolu**, la coordination atteinte par l'utilisateur est faible puisqu'il ne manipule qu'un seul DDL.

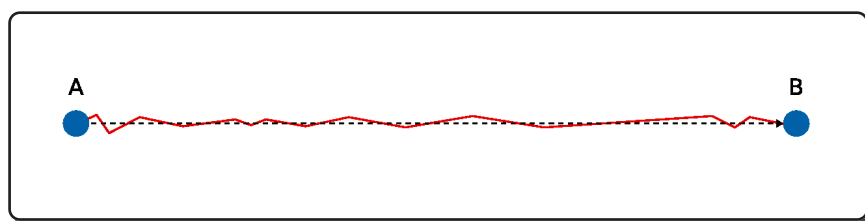


FIGURE 3.3 – Durant la réalisation de la tâche, l'utilisateur ne manipule qu'un DDL. La valeur de la mesure d'inefficacité correspondant à la trajectoire de l'utilisateur est pourtant très faible.

Zhai *et al.* [Zhai 98] définissent également la mesure d'inefficacité dans le contexte de la tâche d'orientation. Le principe est similaire. Pour une tâche d'orientation, le chemin optimal est défini par un vecteur de rotation, *i.e.* un axe de rotation et un angle de rotation autour de cet axe. Le chemin optimal correspond au vecteur de rotation dont l'angle de rotation est le plus petit possible. La mesure est alors définie de la manière suivante :

$$I = \frac{\text{Total des angles de rotations appliquées} - \text{Angle de rotation associé au chemin optimal}}{\text{Total des angles de rotations appliquées}} \quad (3.2)$$

Les remarques que nous pouvons faire sont les suivantes. Comme pour une tâche de positionnement, dans le cas d'une tâche d'orientation impliquant un ou deux DDL, la mesure d'inefficacité peut avoir une valeur faible alors que, dans l'absolu, l'utilisateur ne manipule que un ou deux DDL. De plus, dans le cas de la coordination des DDL en rotation, la dépendance de la mesure au chemin optimal peut être problématique. En effet, Parsons *et al.* ont conduit une étude dans laquelle ils montrent l'importance de la difficulté pour un utilisateur d'identifier la rotation minimale entre deux orientations d'un même objet [Parsons 95]. Il est donc quasiment impossible pour un utilisateur d'identifier le chemin optimal. Pour une tâche d'orientation, une mesure de coordination ne devrait donc pas dépendre du chemin optimal.

3.1.2 $\mathcal{M}$ – metric

Masliah *et al.* [Masliah 00] ont proposé une métrique de coordination appelée $\mathcal{M}$ – metric. L’approche proposée est différente de celle utilisée pour la définition de l’inefficacité. Pour définir cette métrique, ils distinguent deux choses :

1. La mesure de simultanéité (*simultaneity of control*) : elle vise à quantifier le niveau de parallélisme dans la manipulation des DDL ;
2. La mesure d’efficacité (*efficiency of control*) : elle vise à quantifier l’efficacité d’un geste par rapport au chemin optimal.

Masliah et Milgram définissent alors la coordination (qu’ils appellent également **allocation du contrôle**) comme étant le produit de la valeur de **simultanéité** et de la valeur d’**efficacité**. Ici, nous nous intéresserons plus en détail à la mesure de simultanéité car c’est celle qui se rapproche le plus de notre définition de la coordination. De plus, la mesure d’efficacité, dépendant du chemin optimal, souffre des mêmes lacunes que la mesure d’inefficacité.

La simultanéité est calculée en utilisant l’erreur à la cible pour chaque DDL. Cette erreur est définie par la différence entre la position de la cible et la position courante. À l’aide de cette erreur, ils définissent, pour chaque DDL, la **fonction de réduction d’erreur** (notée E_i) qui est la valeur instantanée de la réduction de l’erreur entre la position courante et la cible. Lorsque la valeur de réduction de l’erreur est positive – *i.e.* que l’utilisateur s’éloigne de la cible – la valeur de la fonction de réduction d’erreur est égale à 0. Ils normalisent cette fonction pour définir la **fonction normalisée de réduction d’erreur**, notée $NERF_i$.

$$NERF_i(t) = \begin{cases} \frac{-dE_i(t)}{dt} \times \frac{1}{ACT_i} & , \text{ si } E_i > 0 \\ 0 & , \text{ si } E_i < 0 \end{cases} \quad (3.3)$$

ACT_i est le total de la réduction d’erreur non normalisée du i^{eme} DDL pour la tâche effectuée. ACT_i ne peut être calculé qu’une fois la tâche terminée. La fonction $NERF_i(t)$ est définie pour chaque DDL. La simultanéité est alors calculée de la manière suivante :

$$SIMULTANEITE = \int_0^T \text{Min}(NERF_1(t), NERF_2(t), \dots, NERF_n(t)) dt \quad (3.4)$$

T représente le temps d’exécution de la tâche. Nous pouvons faire plusieurs remarques quant à la définition de cette mesure. Les auteurs ignorent les phases où l’utilisateur s’éloigne de la cible, ce qui implique que cette métrique ne permet de mesurer la coordination des DDL que de manière directionnelle (*i.e.* orientée vers la résolution de

la tâche). Toutefois, dans des tâches comme les tâches d'orientation, l'utilisateur peut s'éloigner de la cible, mais ne pas percevoir cet éloignement. De plus, l'utilisation d'une fonction *Min* dans la définition de la *SIMULTANEITE* peut être problématique. Considérons la manipulation de trois DDL. Si l'utilisateur manipule deux DDL à la fois avec une même vitesse et ne manipule que très peu le troisième, la valeur de simultanété sera très faible (proche de 0 si la vitesse du troisième DDL est proche de 0). Or, la manipulation de deux DDL peut être considérée comme ayant une valeur de simultanété intermédiaire. Cette mesure n'est pas à même de tenir compte de ce phénomène et ne permet donc pas de quantifier des niveaux de coordination intermédiaires.

Nous proposons donc une nouvelle mesure de la coordination.

3.2 NDC : Nombre de Degrés de liberté Combinés

Comme nous l'avons vu, les définitions précédentes considèrent la coordination comme dépendante de la tâche à effectuer. Toutefois, les caractéristiques d'une tâche peuvent exiger un niveau de coordination particulier (une illustration est donnée en Figure 3.4). C'est pourquoi, avant de présenter la métrique que nous proposons, il nous paraît important de préciser ce que nous entendons par coordination en en proposant une définition.

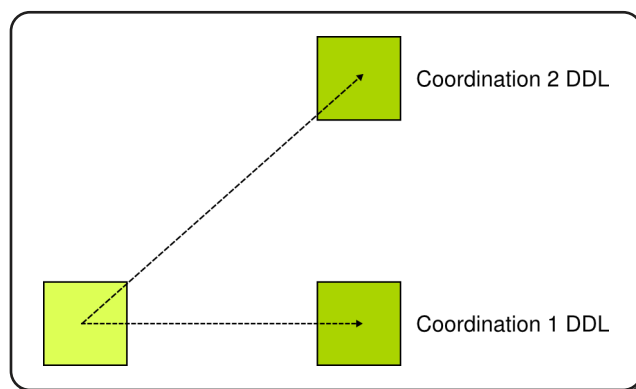


FIGURE 3.4 – Dans le cas d'un positionnement 2-D, la position de l'objet cible va définir la coordination nécessaire afin d'effectuer la tâche d'une manière optimale.

Le terme coordonné sous-entend la notion de **simultanété** et d'**harmonie**. Dans le contexte de l'interaction 3-D, nous nous intéressons à la coordination de la manipulation des DDL de la tâche à effectuer. Dans ce contexte, nous définissons un geste coordonné comme **un geste au travers duquel l'utilisateur manipule simultanément et dans l'exacte même proportion les DDL qui composent la tâche**.

Cette définition de la coordination nous semble plus appropriée au cadre de l'interac-

tion 3-D que les définitions précédemment proposées. Par rapport à cette définition, les mesures existantes souffrent de certaines lacunes. Elles nous paraissent donc peu adaptées à la compréhension du comportement utilisateur dans le contexte de l'interaction 3-D. Nous proposons donc une nouvelle mesure, appelée **Nombre de DDL Combinés** (notée NDC), permettant de quantifier la coordination dans un espace de manipulation à n DDL. Cette mesure, à la différence des mesures précédentes, ne dépend pas du chemin optimal de la tâche. De plus, cette mesure est capable de distinguer différents niveaux de coordination. Avant de présenter la forme générale de la mesure, nous illustrons son comportement et sa formulation dans le cas d'une tâche d'orientation 3-D.

3.2.1 Exemple : cas de l'orientation 3-D

Une tâche donnée implique la manipulation d'un nombre fixe de DDL. Pour effectuer une tâche, l'utilisateur peut manipuler les différents DDL associés à la tâche de différentes manières. Dans le cas de l'orientation d'un objet dans un espace 3-D, l'utilisateur peut effectuer la tâche en manipulant l'ensemble des DDL simultanément tout au long de la tâche. Il peut également effectuer la tâche en ne manipulant qu'un seul DDL à la fois, décomposant la tâche à trois DDL en trois sous-tâches à 1 DDL. Il peut encore mettre en place une stratégie hybride, en combinant les DDL dans un premier temps, pour enfin séparer la manipulation des DDL en fin de tâche. Deux de ces stratégies sont illustrées en Figure 3.5. Le choix de ces stratégies peut découler d'une stratégie de gestion de la charge cognitive associée à la tâche.

Tout d'abord, nous proposons de quantifier la séparation des DDL en définissant la mesure *Magnitude of Degrees of Freedom's Separation* (notée MDS). L'objectif de cette mesure est de donner une valeur quantitative permettant de savoir, à chaque instant de la tâche, quel est le niveau de séparation des DDL. Ainsi, lorsque l'utilisateur manipule un DDL, *i.e.* tourne l'objet autour d'un axe de rotation particulier, nous voulons que la mesure ait une valeur élevée. Au contraire, lorsque l'utilisateur effectue une rotation complexe en tournant l'objet autour des trois axes de rotation simultanément et avec la même amplitude, nous voulons que la mesure ait une valeur faible.

Afin de calculer cette mesure, nous utilisons les courbes de vitesses angulaires (notées v_x , v_y et v_z). Ceci est illustré en Figure 3.6. Ces courbes représentent l'amplitude de la variation des différents DDL au cours de la tâche. Pour quantifier l'amplitude de la manipulation d'un DDL au cours d'une période de temps, nous utilisons l'intégrale de la courbe de vitesse associée au DDL correspondant sur cette même période de temps.

$$V_i = \int_T^{T+1} v_i(t) dt, \forall i \in \{x, y, z\}$$

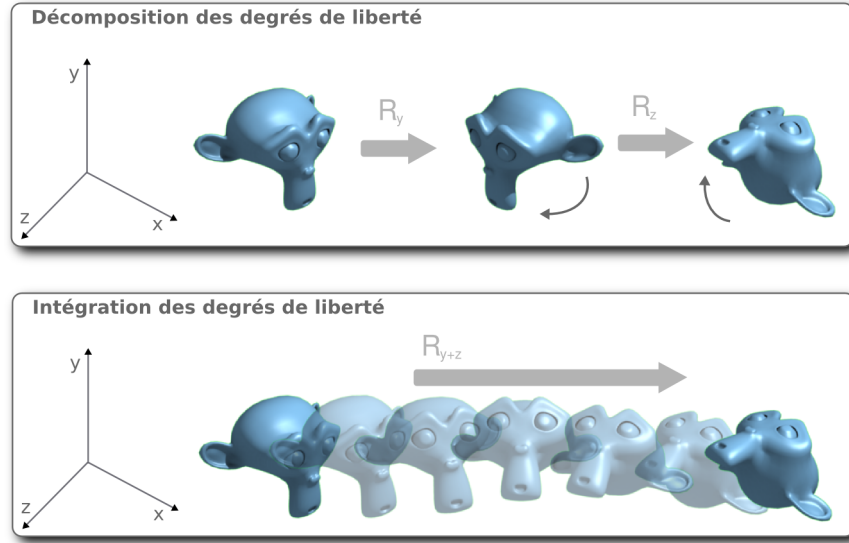


FIGURE 3.5 – Illustration de l'utilisation de deux stratégies possibles pour effectuer une tâche d'orientation. Sur le haut de l'image l'utilisateur décompose la tâche en manipulant un DDL à la fois. Au contraire, sur le bas de la figure, l'utilisateur combine deux DDL à la fois.

Afin de quantifier la séparation des DDL, nous comparons deux à deux chacun des DDL en calculant la valeur absolue de la différence des intégrales associées. Plus la somme de ces différences est importante plus l'utilisateur sépare les DDL. Plus la somme de ces différences est faible, plus le nombre de DDL manipulés simultanément est important. La définition mathématique de la mesure est la suivante :

$$MDS_T = \frac{|V_x - V_y| + |V_x - V_z| + |V_y - V_z|}{2 \times V_{tot}}$$

$$V_{tot} = V_x + V_y + V_z$$

La somme des différences est normalisée par la vitesse totale afin d'obtenir un résultat indépendant de la vitesse totale. Elle est également divisée par 2 afin d'obtenir un résultat contenu dans l'intervalle $[0, 1]$. La valeur de MDS a une signification particulière du point de vue de la manipulation des DDL. Les valeurs clés ainsi que leurs significations sont détaillées en Figure 3.6.

La mesure NDC est une simple transformation mathématique permettant une lecture plus facile de la coordination des DDL. NDC donne explicitement le nombre de DDL manipulés simultanément par l'utilisateur au cours d'une période de temps. En l'occurrence, pour une tâche d'orientation, la valeur de NDC est une valeur réelle comprise

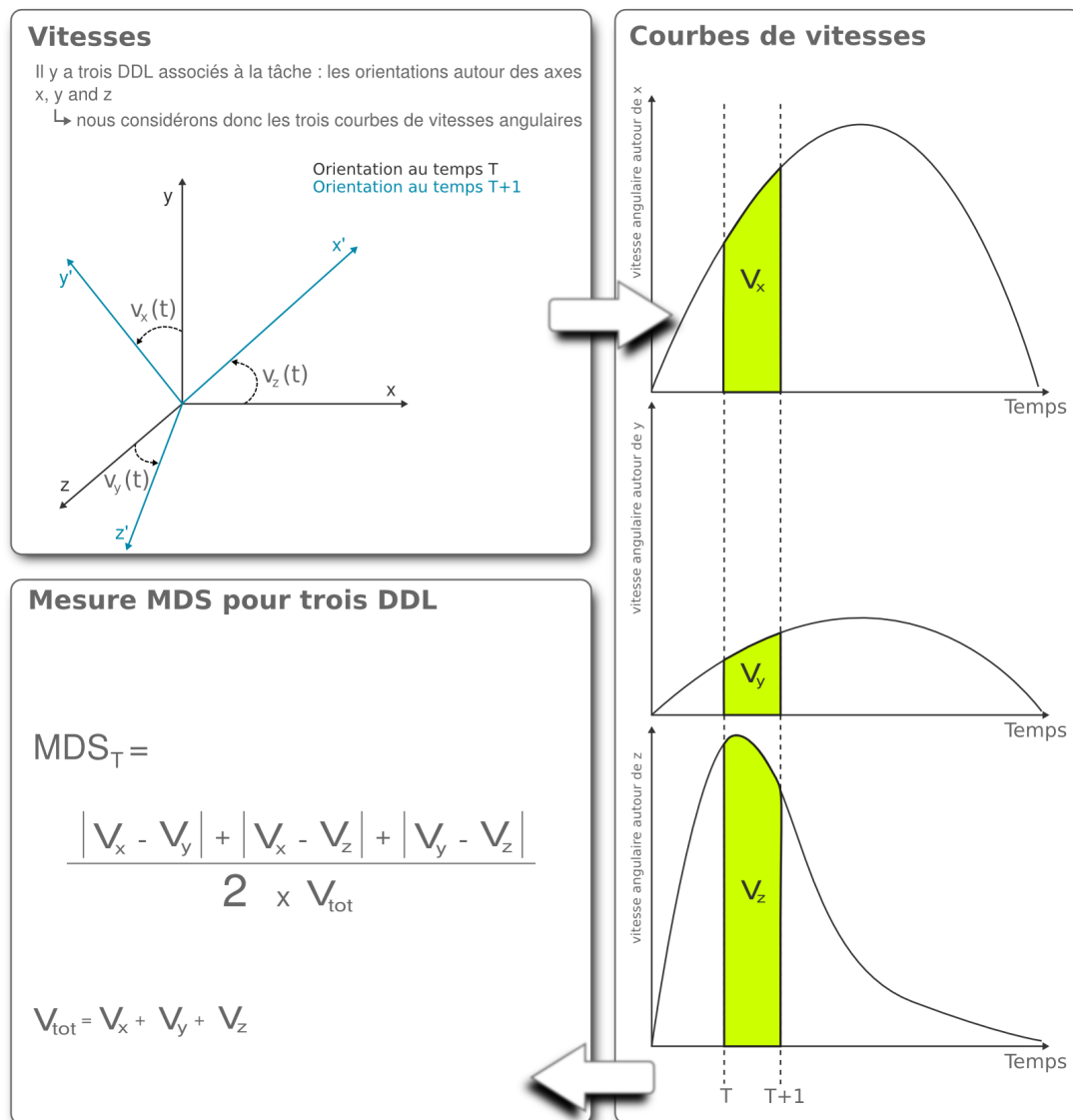


FIGURE 3.6 – Illustration du calcul de la mesure MDS pour une tâche de rotation. Les valeurs clés sont : 0 (manipulation des trois DDL), 0,5 (manipulation de deux DDL) et 1 (manipulation d'un DDL).

entre 1 et 3 si au moins une des vitesses est non nulle. Si l'orientation de l'objet n'a pas été modifiée au cours de la période de temps, la valeur de NDC est nulle.

$$NDC_T = \begin{cases} 1 + 2 \times (1 - MDS_T) & , \text{ si } V_x \neq 0 \text{ ou } V_y \neq 0 \text{ ou } V_z \neq 0 \\ 0 & , \text{ sinon} \end{cases} \quad (3.5)$$

La mesure peut être facilement adaptée pour quantifier la coordination dans le cadre d'une tâche de positionnement 3-D. Pour cela, il suffit simplement d'appliquer la mesure NDC aux trois vitesses de déplacement – selon l'axe x , y et z – de l'objet manipulé. Nous pouvons remarquer que cette mesure peut être utilisée pour quantifier la coordination d'un ensemble quelconque de DDL. La seule restriction est la suivante : les DDL comparés doivent être **de même nature**. En effet, on ne peut comparer des DDL associés à la position de l'objet à des DDL associés à l'orientation du même objet. Les quantités mesurées par les vitesses sont exprimées dans des unités différentes, en l'occurrence des mètres par seconde et des radians ou degrés par seconde. Il n'est donc pas possible de quantifier la coordination dans une tâche de *docking* à six DDL, impliquant à la fois des DDL en translation et en rotation. Après avoir présenté la formulation générale permettant de mesurer la coordination de n DDL, nous présenterons une solution à cette restriction.

3.2.2 Formulation générale pour n DDL

Considérons une tâche nécessitant la manipulation de n DDL. Ces DDL sont notés $DDL_1, DDL_2, \dots, DDL_n$. Nous allons généraliser la mesure NDC à la manipulation de ces n DDL. Comme précédemment, la définition de la mesure NDC repose sur la définition de la mesure MDS. Nous généralisons donc cette dernière dans un premier temps.

Pour une période de temps donnée, nous voulons que la mesure MDS ait une valeur élevée lorsque l'utilisateur ne manipule qu'un DDL. Au contraire, lorsque l'utilisateur combine les n DDL sur une période de temps, la valeur de la mesure doit être faible. Pour chaque DDL_i , nous calculons la fonction représentant la vitesse de variation du DDL au cours du temps notée $V_i(t)$. Comme dans le cas de l'orientation, pour un pas de temps donné, nous calculons l'intégrale de la vitesse de variation du DDL. Pour $i \in [1, N]$, nous calculons donc :

$$I_i = \int_T^{T+1} V_i(t) dt$$

Nous comparons chaque paire de DDL en calculant la valeur absolue de la différence des intégrales. La somme des différences nous permet de connaître le niveau de séparation des DDL sur la période de temps considérée. Pour n DDL, la formulation est alors la suivante :

$$MDS_T = \frac{\sum_{i=1}^n \sum_{j=i+1}^n |I_i - I_j|}{I_{tot} \times (n - 1)}$$

$$I_{tot} = \sum_{i=1}^n I_i$$

De manière similaire à la formulation proposée pour la tâche d'orientation 3-D, la somme des différences est normalisée par la vitesse totale afin d'obtenir un résultat indépendant de la vitesse totale. Elle est également divisée par $n - 1$ afin d'obtenir un résultat dans l'intervalle $[0, 1]$.

À présent, nous pouvons définir la mesure NDC. Cette mesure a une valeur réelle comprise entre 1 et n , indiquant le nombre de DDL combinés au cours d'une période de temps. Elle a une valeur nulle lorsqu'aucun des DDL n'est manipulés au cours de cette période de temps. La formulation est la suivante :

$$NDC_T = \begin{cases} 1 + (n - 1) \times (1 - MDS_T) & , \text{ si } \exists i \text{ tel que } V_i \neq 0 \\ 0 & , \text{ sinon} \end{cases} \quad (3.6)$$

La mesure NDC permet de mesurer la coordination lors de la manipulation de n DDL. Comme nous l'avons indiqué précédemment, cette mesure ne permet de mesurer la coordination de n DDL que si ceux-ci sont de même nature, *i.e.* dont les unités de mesures sont identiques. Pour une tâche de *docking*, où l'utilisateur manipule à la fois des DDL associés à la position de l'objet mais aussi des DDL associés à l'orientation de l'objet, nous ne pouvons donc pas mesurer la coordination en utilisant les six courbes de vitesses associées aux différents DDL.

La solution que nous proposons est la suivante. Le calcul de la coordination dans une tâche de *docking* se fait en deux temps. Tout d'abord, il faut calculer séparément la coordination des trois DDL associés à la position de l'objet (NDC_t) et la coordination associée aux trois DDL associés à l'orientation de l'objet (NDC_r). Nous proposons ensuite de sommer les valeurs de coordination afin d'obtenir une valeur de coordination globale. Dans ce cas $NDC = NDC_t + NDC_r$.

Plus généralement, lorsqu'une tâche implique la manipulation de m groupes de DDL de types différents ($G_1, \dots, G_m$), le calcul de la coordination est défini de la manière suivante :

$$NDC = \sum_{i=1}^m NDC_i, NDC_i \text{ représentant la coordination associée au groupe } G_i \quad (3.7)$$

Nombre de DDL	Simultanéité	Inefficacité	NDC
1	0	0	1
2	0	0	2
3	1	0	3

TABLE 3.1 – Pour les trajectoires nécessitant la manipulation d'un, deux et trois DDL, les valeurs calculées par chacune des mesures.

3.2.3 Implémentation

NDC a été implémentée et intégrée dans la librairie VRHPG. Cette librairie permet de faciliter l'implémentation de protocoles expérimentaux en fournissant un ensemble d'outils logiciels de base. Dans cette librairie, nous retrouvons l'implémentation de la mesure NDC pour des tâches à 3 et n DDL.

Nous avons utilisé cette implémentation afin d'évaluer et de comparer notre métrique aux autres métriques de coordination. Nous avons enfin évalué le comportement de notre mesure dans un cas plus réaliste, le cas d'une tâche de suivi de trajectoires.

3.2.4 Comparaison

Nous avons comparé la mesure NDC à la simultanéité et à la mesure d'inefficacité dans des cas simples afin de mettre en avant leurs différences de comportement. Pour cela, nous avons mis en place un programme de test simple permettant de simuler des tâches de suivi de trajectoires impliquant un nombre de DDL fixé et connu.

Nous avons utilisé vingt six trajectoires impliquant un nombre fixe d'un, deux et trois DDL. Les différentes trajectoires générées sont illustrées en Figure 3.7. Nous simulons dans l'application le déplacement d'un objet le long de cette trajectoire. Pour se faire, nous supposons que la durée de la tâche est fixée à un temps T . Nous calculons la position de l'objet au temps t par simple interpolation entre la position initiale de l'objet et la position finale – *i.e.* d'une extrémité à l'autre de la trajectoire courante.

$$\text{Position courante} = \left(1 - \frac{t}{T}\right) \times \text{Position initiale} + \frac{t}{T} \times \text{Position finale} \quad (3.8)$$

Ceci nous permet de créer un jeu de données sur lesquelles nous allons pouvoir calculer les différentes mesures de coordination. Les résultats numériques que nous obtenons sont donnés dans le Tableau 3.1.

Les résultats obtenus montrent les faiblesses déjà évoquées précédemment. Quel que

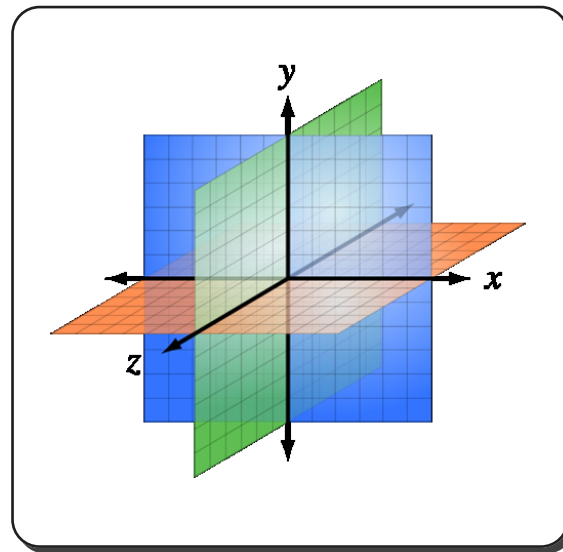


FIGURE 3.7 – Cette figure permet d'illustrer les différentes directions utilisées dans l'application. Sur la figure, sont présentées les trois directions élémentaires à un DDL. S'ajoutent à ces trois directions, les six diagonales des carrés représentés et les quatre diagonales du cube contenant les trois carrés.

soit le nombre de DDL à manipuler au cours de la tâche, la mesure d'inefficacité est identique. Ceci est dû au fait que la mesure d'inefficacité ne se base que sur le chemin optimal pour le calcul de la coordination.

En ce qui concerne la mesure de simultanéité, nous pouvons remarquer que celle-ci n'est pas capable de différencier la coordination d'un et de deux DDL. Comme son calcul se base sur le minimum de la fonction de réduction d'erreur normalisée, l'absence de manipulation d'un DDL se traduit par une mesure de simultanéité nulle.

Nous pouvons voir que NDC permet de mettre en avant les différents niveaux de coordination. Elle répond bien aux critères que nous nous sommes fixés lors de la définition de la coordination. Toutefois, nous avons voulu valider notre mesure dans un cas plus réaliste.

3.2.5 Validation

Afin de valider le comportement de la mesure NDC dans une situation plus réelle, nous avons mis en place un protocole simple. Ce protocole a été effectué dans l'environnement semi-immersif dont nous disposons. Cet environnement est décrit plus précisément au Chapitre 4. Le périphérique d'interaction utilisé est le Flystick, un périphérique

à six DDL disposant de boutons. Nous allons maintenant décrire le protocole ainsi que les résultats que nous avons obtenus.

Protocole

La méthodologie utilisée pour le protocole est similaire à celle utilisée pour la mise en place de l'application de test décrite en Section 3.2.4. La tâche à effectuer consiste à déplacer une sphère d'un bout à l'autre d'une trajectoire linéaire. Pour se faire, l'utilisateur clique sur la gâchette du Flystick et maintient la gâchette enfoncée pour déplacer la sphère. L'objectif fixé est de suivre la trajectoire au plus proche tout au long de la tâche. Une fois la sphère positionnée à l'extrémité de la trajectoire, la tâche est considérée comme terminée. Pour commencer une nouvelle tâche de suivi de trajectoires, l'utilisateur clique simplement sur un des boutons se trouvant sur le Flystick.

Nous distinguons les trois conditions suivantes : suivi de trajectoire à un DDL, suivi de trajectoire à deux DDL et suivi de trajectoire à trois DDL. L'utilisateur effectue vingt et une tâches pour chacune des conditions. Dans le cas de suivi de trajectoires à un DDL, il y a sept tâches pour chacune des trois dimensions de l'espace 3-D. Dans le cas de suivi de trajectoires à deux DDL, il y a 7 tâches pour chacune des trois combinaisons de deux dimensions de l'espace 3-D.

Nous recueillons les positions de la sphère tout au long de la tâche pour pouvoir calculer la mesure NDC a posteriori. L'objectif est ici de vérifier que, dans un cas réaliste de manipulation par un utilisateur, la mesure fournit des résultats pertinents. Ceci permet notamment de vérifier le comportement de la mesure sur des gestes imparfaits, affectés par des tremblements.

Résultats

Nous avons récoltés 252 suivis de trajectoires. Voici la moyenne de la mesure NDC pour les différentes conditions testées :

- Trajectoires à un DDL : moyenne de NDC = 1,44 ;
- Trajectoires à deux DDL : moyenne de NDC = 2,09 ;
- Trajectoires à trois DDL : moyenne de NDC = 2,79.

Les résultats que nous obtenons sont cohérents et montrent bien que la mesure de coordination est viable dans le contexte de l'interaction 3-D. Elle permet de quantifier la coordination de manière assez précise. Nous pouvons toutefois noter une certaine sensibilité aux tremblements. Les petits tremblements semblent affecter la mesure mais ne semblent pas fausser les résultats. Nous avons pu constater lors des phases de tests, qu'il est presque impossible, dans le contexte de la manipulation 3-D, de déplacer un objet selon une seule dimension.

3.3 Conclusion

Nous proposons une nouvelle métrique de coordination. Cette métrique permet de quantifier explicitement la coordination des gestes effectués par les utilisateurs. Nous avons évalué cette métrique lors de deux tests. Le premier consiste à comparer la métrique aux métriques existantes. Cette comparaison a été effectuée dans une application où nous simulons des mouvements linéaires à un, deux et trois DDL. Nous avons alors pu mettre en avant les différences entre les métriques existantes et la métrique NDC.

Dans un second temps, nous avons validé notre métrique dans un contexte plus réaliste. Nous avons récolté les données issues des capteurs optiques lors de tâches de suivi de trajectoires à un, deux et trois DDL. Les résultats semblent indiquer que la mesure permet de quantifier la coordination des gestes des utilisateurs dans une situation réaliste. Nous utiliserons donc cette métrique dans les études que nous mènerons afin de quantifier la coordination des gestes des utilisateurs. Particulièrement, nous calculerons à de nombreuses reprises le temps que les utilisateurs passent à manipuler un, deux et trois DDL. Pour se faire, nous devons tenir compte de la sensibilité de notre métrique aux tremblements des utilisateurs au cours de la manipulation. C'est pourquoi, nous utiliserons les seuils suivants afin de déterminer la dimensionnalité d'un geste :

- Un mouvement est considéré comme un mouvement à un DDL si et seulement si la valeur NDC associée est dans l'intervalle $[1; 1,66[$;
- Un mouvement est considéré comme un mouvement à deux DDL si et seulement si la valeur NDC associée est dans l'intervalle $[1,66; 2,33[$;
- Un mouvement est considéré comme un mouvement à trois DDL si et seulement si la valeur NDC associée est dans l'intervalle $[2,33; 3]$.

Deuxième partie

Expérimentations et résultats

Introduction

Dans le cadre de notre travail, nous nous intéressons principalement aux problématiques d'interaction liées à la modélisation géométrique en environnement de RV. Dans un tel contexte, l'utilisateur doit pouvoir manipuler avec précision l'ensemble des entités géométriques disponibles (sommets, arêtes et faces d'un maillage, points de contrôle d'une courbe, position et trajectoire d'une contrainte de déformation, *etc.*). Notre objectif est de proposer et de mettre en place des techniques d'interaction adaptées aux besoins des utilisateurs.

Toutefois, les tâches rencontrées en modélisation sont de natures très diverses (positionnement, orientation, suivi de trajectoires, navigation, *etc.*). Dans ce contexte, il est difficile de mettre en place et d'évaluer des solutions spécifiques à chacune des tâches. Par ailleurs, l'évaluation de solutions spécifiques ne permet généralement pas d'extraire des guides de conception. Ces guides sont importants pour aiguiller les concepteurs dans les choix des techniques d'interaction à mettre en place.

Par rapport au cadre théorique proposé, nous nous intéressons à l'influence des contraintes physiques ainsi que de la fonction de contrainte sur les performances des utilisateurs. Nous allons étudier l'influence de ces deux facteurs sur les deux types de tâches que nous différencions, les tâches à contrôle final et les tâches à contrôle continu. Notamment, afin d'améliorer le contrôle de l'utilisateur dans des applications de modélisation géométrique, nous envisageons de mettre en place des techniques basées sur la simulation d'une interface tactile pour l'interaction 3-D. Par rapport au cadre théorique et par rapport à des techniques de manipulation directe, l'utilisation d'une telle solution introduit deux modifications majeures dans la situation de manipulation :

1. **L'introduction d'une contrainte physique** introduite par le support physique utilisé pour la simulation d'une interface tactile ;
2. La **décomposition des tâches** de dimensionnalité supérieure à 2. L'utilisation d'une surface de manipulation induit une réduction de la dimension de l'espace

moteur. En manipulant sur une surface tactile ne permettant de suivre qu'un seul point de contact, l'utilisateur ne peut manipuler que deux DDL à la fois.

Nous étudions l'influence des contraintes physiques et de la décomposition induite par l'utilisation de ces contraintes sur différentes classes de tâches : les tâches à trois DDL à contrôle final et à contrôle continu. Pour se faire, nous avons mis en place trois expérimentations qui nous permettent de quantifier l'influence respective du support physique et de la décomposition. Nous exploitons enfin ces résultats afin de proposer des solutions spécifiques à notre cadre applicatif. Dans l'approche que nous proposons, le contexte de la modélisation géométrique n'est qu'un contexte applicatif permettant de mettre en oeuvre, dans un cas concret, les solutions proposées.

Décomposition et orientation 3-D

Sommaire

4.1	Métaphore du potier	70
4.1.1	Mise en oeuvre	70
4.1.2	Fonctionnement	71
4.2	Protocole expérimental	72
4.2.1	Matériel	72
4.2.2	Techniques comparées	74
4.2.3	Procédure expérimentale	76
4.2.4	Mesures	78
4.2.5	Hypothèses	78
4.3	Résultats	79
4.3.1	Performances	79
4.3.2	Manipulation des degrés de liberté	80
4.4	Analyse et discussions	81
4.4.1	Performances	81
4.4.2	Coordination des degrés de liberté	83
4.5	Conclusion	84

L'intégration des DDL dans un même geste utilisateur est souvent présenté comme un avantage majeur pour les techniques d'interaction 3-D. Toutefois, l'analyse de l'état de l'art indique que l'introduction de contraintes virtuelles lors de la manipulation peut être nécessaire. Les contraintes virtuelles sont utilisées notamment dans des phases où l'utilisateur a besoin de précision. Les contraintes nécessitent de décomposer partiellement ou totalement les tâches. L'introduction d'une décomposition soulève un certain nombre de questions : comment fonctionne cette décomposition du point de vue de l'impact sur

la stratégie de réalisation de la tâche ? Quels sont les impacts sur les performances ? Quelles recommandations en tirer ?

Dans une première expérimentation, nous souhaitons étudier le comportement des utilisateurs lors d'une tâche à contrôle final, l'orientation dans un espace 3-D. Plus particulièrement, l'objectif de cette première expérimentation est d'étudier les stratégies de réalisation basées sur la coordination des DDL. Pour se faire, nous comparons deux techniques de rotations. La première, inspirée de la manipulation directe, permet de manipuler aisément plusieurs DDL à la fois dans un même geste de contrôle. La seconde est une technique basée sur la simulation d'une interface tactile. Cette technique permet de manipuler facilement un DDL à la fois. Une de nos hypothèses est que les utilisateurs décomposent partiellement cette tâche, même en utilisant la première technique.

Avant de présenter le protocole que nous avons mis en place, nous décrivons la technique que nous proposons ainsi que sa mise en oeuvre dans l'environnement matériel dont nous disposons.

4.1 Métaphore du potier

Nous proposons une technique permettant d'orienter rapidement un objet dans un espace 3-D. Dans un contexte de modélisation, l'orientation rapide d'un objet permet de donner un point de vue idéal sur la zone que l'utilisateur est entrain d'éditer. Ce type de manipulation intervenant à de nombreuses reprises au cours de la modélisation d'un objet 3-D, nous souhaitons utiliser la simulation d'une interface tactile afin de donner à l'utilisateur un meilleur confort. Cela pourrait permettre de minimiser la fatigue introduite au cours de longues phases de modélisation. La solution que nous proposons est appelée la **métaphore du potier** (notée MP).

Avant de présenter la technique nous décrivons la mise en oeuvre proposée pour la simulation d'une interface tactile.

4.1.1 Mise en oeuvre

Cette mise en oeuvre est spécifique au contexte matériel dont nous disposons. Une des spécificités de notre dispositif de RV est que l'écran horizontal du Workbench se trouve à portée de mains de l'utilisateur. En utilisant les capteurs de pression se trouvant aux extrémités des doigts sur les gants de données, nous sommes capables de savoir si l'utilisateur est actuellement en train de toucher l'écran ou non. Ceci est illustré en Figure 4.1. En se basant sur le système de tracking optique, il nous est possible de calculer les déplacements de la main sur la surface de l'écran. Cette solution nous permet de simuler l'utilisation d'un écran tactile permettant de détecter deux points de contact

(un point de contact pour chaque main).

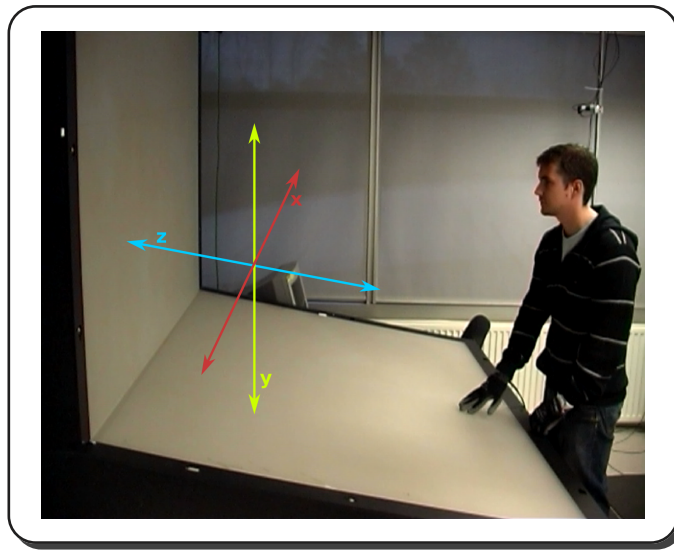


FIGURE 4.1 – L'état des capteurs de pressions et la position de la main dans l'environnement nous permet de savoir si l'utilisateur touche l'écran se trouvant à portée de main. Les mouvements de la main sur l'écran sont effectués dans le plan X-Z.

4.1.2 Fonctionnement

Avec cette technique, l'utilisateur touche l'écran se trouvant à portée de main afin de pouvoir orienter l'objet dans l'espace 3-D. L'espace moteur étant réduit à un espace à deux dimensions, les tâches de dimensions supérieures à deux doivent être décomposées. C'est le cas pour la tâche d'orientation 3-D. La décomposition que nous avons choisie a été définie de manière empirique. Bien que plusieurs décompositions soient possibles, une décomposition semble plus naturelle que les autres et c'est celle que nous avons utilisé pour cette technique :

- Une rotation autour de l'axe X (*i.e.* le roulement de l'objet autour de l'axe X) se fait par un déplacement avant – arrière de la main dominante ou non dominante sur l'écran ;
- Une rotation autour de l'axe Y (*i.e.* le roulement de l'objet autour de l'axe Y) se fait par un déplacement gauche – droite de la main dominante sur l'écran. C'est de cette association, axe de rotation - mouvement de translation, que provient le nom de métaphore du potier. Comme lorsqu'on utilise un tour de potier, l'objet tourne autour de son axe vertical grâce à un mouvement latéral de la main ;
- Une rotation autour de l'axe Z (*i.e.* le roulement de l'objet autour de l'axe Z) se fait par un déplacement gauche – droite de la main non dominante sur l'écran.

Chacune des mains permet à l'utilisateur de tourner l'objet autour de deux axes de rotation. Le détail des relations entre les différents déplacements de la main et les rotations appliquées à l'objet sont illustrées sur la Figure 4.2. L'utilisateur peut effectuer des mouvements en diagonales sur l'écran afin de combiner deux axes de rotations simultanément. Il peut également utiliser les deux mains en même temps afin de combiner l'ensemble des axes de rotation. La combinaison des rotations se fait par un simple cumul des rotations autour des axes x , y et z . Cette technique permet donc de combiner ou de séparer les différents DDL de la tâche de rotation.

Avec cette technique, l'utilisateur peut donc tourner l'objet autour d'un axe de rotation en particulier. Ceci a pour avantage de permettre à l'utilisateur d'effectuer des révolutions complètes autour d'un axe de rotation afin de lui permettre de visualiser un objet dans sa globalité. L'utilisateur peut également orienter l'objet rapidement pour lui permettre d'atteindre des zones auxquelles l'utilisateur souhaite pouvoir accéder. La modélisation se faisant généralement sur l'intégralité du maillage, l'ensemble des parties du modèle 3-D doivent donc être facilement accessibles.

De plus, cette technique permet de décomposer la tâche d'orientation. Comme nous l'avons déjà indiqué précédemment dans ce document, la tâche d'orientation est d'une grande complexité. L'impossibilité des utilisateurs à identifier le chemin optimal entre deux orientations [Parsons 95] peut les conduire à mettre en place des stratégies spécifiques lors de la réalisation de tâches d'orientation 3-D. Ces stratégies peuvent notamment être basées sur une décomposition de la tâche en sous-tâches plus simples – *i.e.* de dimensionnalités réduites.

Il est donc possible que l'utilisateur puisse tirer partie des spécificités de la technique d'interaction que nous proposons, non seulement pour orienter grossièrement et rapidement un objet, mais également lors de tâche d'orientation de précision. Nous proposons donc d'évaluer cette technique d'interaction par rapport à une technique de manipulation directe. L'objectif est ici de voir si la décomposition d'une tâche à contrôle temporel final a une influence sur les performances des utilisateurs.

4.2 Protocole expérimental

Afin d'évaluer les apports de la décomposition de la tâche, nous comparons la technique MP avec une technique inspirée de la manipulation directe.

4.2.1 Matériel

Le contexte matériel que nous présentons ici constitue le contexte matériel pour l'ensemble des études que nous conduirons par la suite.

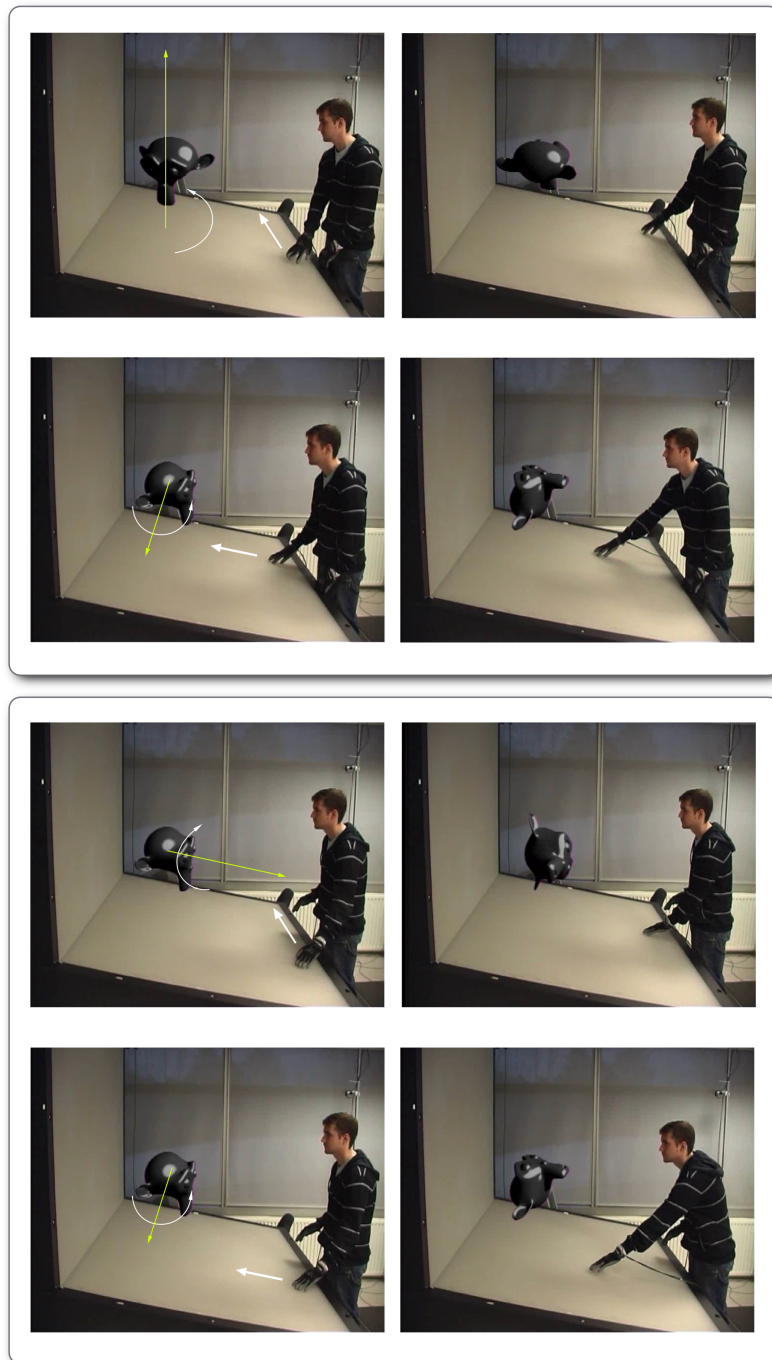


FIGURE 4.2 – Les différentes relations entre les déplacements des mains et les rotations appliquées à l'objet.

L'environnement de RV dont nous disposons au LSIIT est un environnement semi-immersif. Celui-ci est illustré en Figure 4.3. Il est constitué de deux écrans permettant un affichage stéréoscopique à une résolution de 1400 x 1050. Le taux de rafraîchissement monoscopique est de 120 Hz alors que le taux de rafraîchissement stéréoscopique est de 60 Hz. Pour permettre une visualisation 3-D des objets virtuels, l'affichage est synchronisé avec des lunettes à obturation de type CrystalEyes CE-2. Les images sont générées en utilisant deux stations de travail (une pour chacun des écrans) disposant d'un Intel Quad Core cadencé à 3 GHz et une NVIDIA Quadro FX 5600.

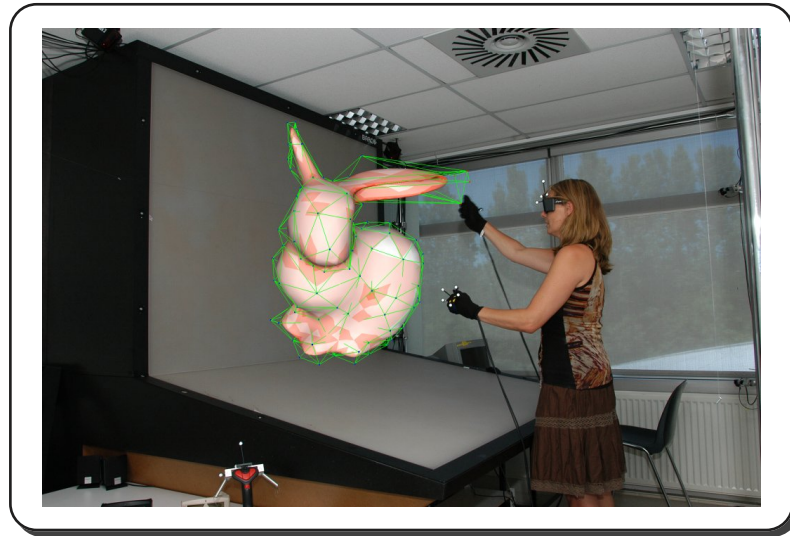


FIGURE 4.3 – Utilisation d'une application de modélisation dans le Workbench, L'utilisateur manipule l'objet virtuel à l'aide des gants de données.

Concernant les périphériques d'interaction, nous disposons d'un système de tracking ART ainsi que d'un ensemble de capteurs optiques. Ce dispositif permet de suivre la position de la tête de l'utilisateur ainsi que la position des différents périphériques. Les utilisateurs disposent de deux gants de données XIST (cf. Figure 4.4) pour interagir avec l'environnement virtuel. Ces gants sont composés d'un ensemble de 10 capteurs : 5 capteurs de flexion et 5 capteurs de pression. Les capteurs de pression sont placés à l'extrémité de chaque doigt.

4.2.2 Techniques comparées

Nous avons demandé aux sujets ayant participé à cette étude d'effectuer une succession de tâches d'orientation en utilisant une technique d'interaction similaire à la manipulation directe (appelée **Rotation avec Intermédiaire** et notée RI) et en utilisant



FIGURE 4.4 – Les gants de données.

la technique MP.

Avec une technique de manipulation directe, l'utilisateur saisit et oriente directement l'objet avec la main. Cette technique offre une manière simple et intuitive d'orienter un objet dans un espace 3-D. Toutefois, ce type de technique ne permet de manipuler que des objets se trouvant à portée de main. Dans notre environnement, pour pouvoir manipuler un objet se trouvant à une position appropriée pour l'évaluation de l'orientation d'un objet, l'utilisateur doit se pencher pour manipuler l'objet. Cette contrainte peut avoir une influence sur la manière dont l'utilisateur effectuera la tâche.

Afin d'éviter ce problème, nous proposons une technique où l'utilisateur saisit et manipule un objet se trouvant à portée de main (dans notre cas, il s'agit d'un cube). L'ensemble des rotations appliquées au cube sont alors appliquées à l'objet. Ceci est illustré sur la Figure 4.5.

L'objectif de cette technique n'est pas d'offrir une interaction directe¹. Cette technique offre une situation dans laquelle l'utilisateur peut facilement évaluer la tâche en cours de réalisation tout en permettant de manipuler simultanément l'ensemble des DDL de la tâche d'orientation 3-D.

Durant le protocole, le fonctionnement de la MP est identique à celui décrit dans la Section 4.1.2. Néanmoins, au moment où nous avons mis en place ce protocole, nous ne disposions pas des gants XIST. Les gants de données utilisés alors étaient des 5DT, des gants sans capteurs de pressions. Pour détecter alors le touché de l'écran se trouvant à portée de main, nous utilisons la position du capteur optique dans l'espace 3-D. Lorsque les coordonnées en Y étaient sous un certain seuil, nous considérons que l'utilisateur

1. Interaction directe au sens de Schneiderman [Schneiderman 87], c'est-à-dire sans media intermédiaires.

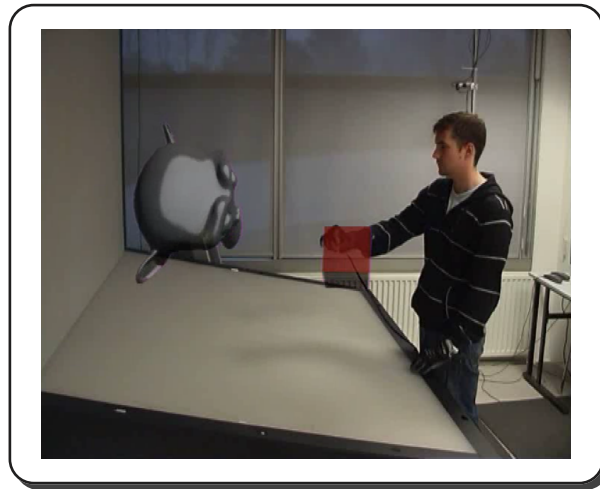


FIGURE 4.5 – L'utilisateur saisit et tourne le cube se trouvant à portée de main afin d'orienter l'objet manipulé.

touchait la surface de l'écran.

4.2.3 Procédure expérimentale

Treize sujets, avec différents niveaux d'expériences de manipulation en environnements virtuels, ont participé à cette étude. Dans cette étude, nous utilisons la même tâche que celle utilisée lors d'études précédemment menées [Chen 88]. Elle est illustrée sur la Figure 4.6. Cette tâche consiste à orienter un objet (une maison) pour que son orientation corresponde à une orientation cible. L'orientation de la cible ne change pas au cours de l'étude et peut être représentée par le quaternion unitaire.

La session commence par une présentation de l'environnement de RV et des périphériques utilisés au cours de l'expérimentation. Après cette présentation, les utilisateurs peuvent manipuler dans l'environnement aussi longtemps qu'ils le désirent avant le début des tests. La session de tests consiste en deux blocs de vingt quatre orientations, un bloc pour chacune des techniques. Nous utilisons ici un plan expérimental intra-sujet (*i.e.* chaque sujet effectue la tâche avec l'ensemble des techniques d'interaction). Afin d'éviter tout effet d'apprentissage, nous contre balançons l'ordre de présentation des techniques d'interaction.

La consigne est d'orienter la maison manipulée le plus précisément possible tout en effectuant la tâche dans un temps raisonnable. Les sujets stoppent la tâche lorsqu'ils pensent avoir atteint une bonne précision. Afin de valider la tâche courante, les sujets doivent mettre la main dominante dans un composant graphique de validation pendant

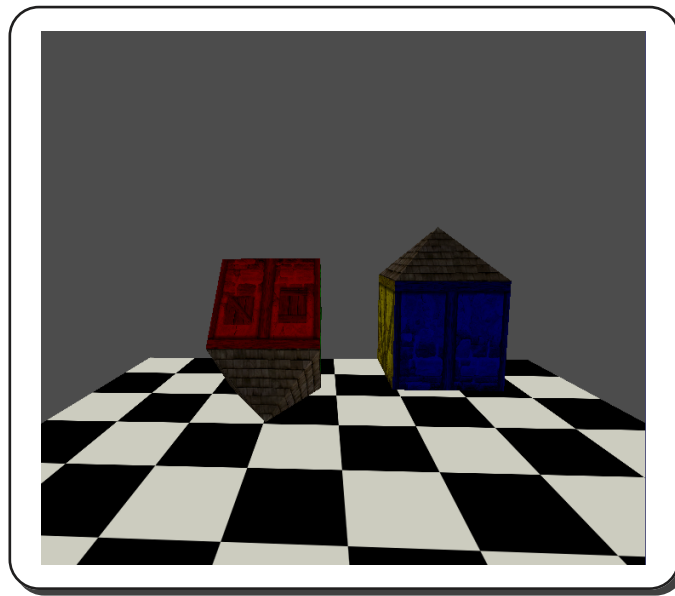


FIGURE 4.6 – La tâche consiste à orienter la maison se trouvant sur la gauche de la scène de sorte que son orientation corresponde à la maison se trouvant sur la droite de la scène.

une seconde. Ce composant graphique est représenté par un cube rouge dans le coin inférieur droit de l'environnement. Une fois la validation effectuée, les sujets rentrent dans une phase de transition où aucune mesure n'est prise. C'est durant cette phase que les utilisateurs sont invités à faire des commentaires ou à se reposer. Pour commencer une nouvelle tâche, les sujets exécutent la même séquence de validation. Le temps nécessaire à la validation n'est pas pris en compte dans les mesures.

Dans cette étude, nous distinguons deux types d'orientations :

- Orientations Simples (OSI) : pour ce type d'orientation, la tâche peut être effectuée en manipulant un DDL uniquement ;
- Orientations Complexes (OC) : pour ce type d'orientation, pour effectuer la tâche, le sujet doit manipuler au moins deux DDL.

Pour chacune des techniques, les sujets doivent effectuer douze orientations simples et douze orientations complexes. Nous distinguons également les orientations en fonction de l'amplitude minimale de l'orientation à effectuer pour accomplir la tâche :

- Petites Orientations (PO) : la maison peut être tournée de moins de 60 degrés pour atteindre l'orientation cible ;
- Grandes Orientations (LO) : la rotation minimale doit être de plus de 60 degrés.

4.2.4 Mesures

Au cours de cette étude, nous enregistrons différentes données quantitatives : la différence d'orientation entre l'objet manipulé et la cible ainsi que la position des périphériques d'interaction. En utilisant ces données, nous calculons les mesures suivantes :

- La Distance Angulaire (DA, en degrés) : la distance angulaire finale entre l'objet manipulé et la cible ;
- Le Temps de Réalisation (TRE, en secondes) : le temps écoulé entre le premier et le dernier mouvement de l'objet ;
- Le Temps Grossier (TG, en secondes) : le temps auquel la distance angulaire est, pour la première fois, en dessous d'un seuil de 18 degrés ;
- Le Temps Fin (TF, en secondes) : le temps auquel la distance angulaire est, pour la première fois, en dessous d'un seuil de 9 degrés ;
- La mesure de coordination (NDC) en utilisant les vitesses angulaires de l'objet manipulé ;
- Pourcentage de temps passé à manipuler 1, 2 et 3 DDL : ce calcul est basé sur la métrique NDC.

Nous utilisons les données ci-dessus afin d'étudier la manière dont l'utilisateur combine les DDL. Nous présentons à présent les hypothèses concernant cette première étude.

4.2.5 Hypothèses

H1 : Puisque la technique RI permet d'effectuer des petits ajustements sur l'orientation de l'objet, elle est plus précise que la technique MP.

H2 : Pour des orientations petites et complexes, la technique RI est plus efficace que la technique MP. Plus particulièrement, puisque la technique RI permet de combiner plusieurs DDL en même temps, elle permet d'effectuer la tâche plus rapidement.

H3 : Pour des orientations larges et simples, la technique MP est plus efficace que la technique RI. Plus particulièrement, puisque la technique MP permet de manipuler un DDL spécifique et d'effectuer des rotations de grandes amplitudes, elle permet d'effectuer la tâche plus rapidement.

H4 : L'utilisateur effectue la tâche d'orientation en manipulant un DDL à la fois en utilisant la technique MP même si elle permet de combiner plusieurs DDL simultanément.

4.3 Résultats

Nous nous trouvons dans le cas d'un plan expérimental intra-sujet. Nous avons donc analysé les données au moyen d'une ANOVA à mesures répétées, à trois facteurs : la technique d'interaction, le type et l'amplitude de la rotation. Lorsque l'un des facteurs semble avoir une influence sur la variable dépendante, nous utilisons une comparaison à l'aide d'un test de Student paire à paire en appliquant une correction de Benjamini et Hochberg [Benjamini 95]. Cette correction permet de réduire le risque alpha, *i.e.* le risque que nous avons de conclure à une différence qui n'existe pas.

Dans l'ensemble des études que nous menons, nous utilisons les seuils de significativité classiques :

- $p - \text{value} > 0,1$: résultat non significatif ;
- $p - \text{value} \in [0,1,0,05]$: résultat indiquant une tendance ;
- $p - \text{value} \in [0,05,0,001]$: résultat significatif ;
- $p - \text{value} < 0,001$: résultat très significatif.

4.3.1 Performances

IT	DA					TRE				
	Amplitude		Complexité			Amplitude		Complexité		
	PO	LO	OSI	OC	Moy.	PO	LO	OSI	OC	Moy.
RI	3,64	4,1	3,87	4,20	4,04	16,28	25,19	20,86	25,86	23,37
MP	3,48	3,83	3,46	4,09	3,78	19,59	25,10	20,71	27,92	24,31
$p - \text{value}$	0,772	0,228	0,215	0,737	0,264	0,175	0,934	0,919	0,144	0,346

TABLE 4.1 – La moyenne selon chacun des critères de performances, la précision et le temps de réalisation.

Le Tableau 4.1 montre la précision finale moyenne obtenue pour chacune des techniques d'interaction comparées. La différence observée entre les techniques n'est pas significative ($F(1,6) = 1,13$). Nous observons également aucune interaction entre la technique utilisée et le type d'orientation ($F(1,6) = 0,41$, $p - \text{value} = 0,52$) ou encore entre la technique et l'amplitude de l'orientation ($F(1,6) = 0,22$, $p - \text{value} = 0,64$). Ces résultats semblent indiquer que les deux techniques offrent le même niveau de précision quelle que soit la caractéristique de la tâche d'orientation.

Toujours dans le même tableau, nous pouvons observer que les temps de réalisation sont également similaires pour l'ensemble des techniques. En effet, nous n'observons aucune influence statistiquement significative de la technique d'interaction ($F(1,6) = 0,85$). Les temps de réalisation sont similaires quelles que soient les caractéristiques

de la tâche. Nous n'observons aucune interaction entre la technique et la complexité de l'orientation ($F(1,6) = 1,46$) ou entre la technique et l'amplitude de l'orientation ($F(1,6) = 3,21$). Ce résultat semble suggérer qu'en général, utiliser une technique de manipulation directe ne semble pas réduire le temps de réalisation.

IT	TG					TF				
	Amplitude		Complexité			Amplitude		Complexité		
	PO	LO	OSI	OC	Moy.	PO	LO	OSI	OC	Moy.
RI	5,73	11,23	8,83	12,03	10,54	7,17	12,86	9,88	13,61	11,82
MP	6,51	9,36	7,21	10,80	9,03	3,48	11,14	8,03	13,20	10,59
$p - value$	0,622	0,002	0,043	0,109	0,006	0,489	0,010	0,034	0,638	0,045

TABLE 4.2 – Le temps moyen nécessaire pour atteindre les deux seuils de précision, grossier et fin.

Bien que nous n'observons aucune différence statistiquement significative au niveau des temps de réalisation, nous observons une différence significative pour les mesures TG et TF. La technique d'interaction semble avoir une influence significative sur le temps nécessaire pour atteindre le seuil de précision grossier ($F(1,6) = 8,52$) ainsi que le seuil de précision fin ($F(1,6) = 4,62$). En moyenne, la mesure TG est diminuée de 16,72 % pour la technique MP. Nous pouvons également noter que la différence est significative dans le cas d'orientations simples et d'orientations de larges amplitudes. Toutefois, la différence n'est pas significative pour les orientations complexes et de faibles amplitudes.

Nous observons également une influence significative de la technique sur la mesure TF. En utilisant la technique MP, les utilisateurs sont en moyenne 11,61 % plus rapides. Ce résultat semble suggérer, qu'en utilisant la technique MP, les utilisateurs sont capables de s'approcher de la cible plus rapidement qu'en utilisant la technique RI. De manière analogue aux résultats obtenus pour la mesure TG, nous observons une influence significative de la technique d'interaction dans le cas d'orientations simples et de larges amplitudes sur la mesure TF. Cette différence n'est pas significative dans le cas d'orientations complexes et de faibles amplitudes.

4.3.2 Manipulation des degrés de liberté

Le Tableau 4.3 montre le pourcentage moyen de temps durant lequel les utilisateurs manipulent un, deux et trois DDL. La technique d'interaction a une influence significative sur ces mesures :

- Un DDL : $F(1,6) = 3366,61$;

- Deux DDL : $F(1,6) = 912,96$;
- Trois DDL : $F(1,6) = 1860,15$.

En utilisant la technique MP, les utilisateurs manipulent un DDL à la fois pendant presque 90 % du temps. Les utilisateurs combinent deux DDL pendant plus de 10 % du temps mais ne réussissent quasiment jamais à combiner trois DDL. En utilisant la technique RI, les utilisateurs manipulent deux DDL à la fois pendant la moitié du temps de manipulation. Ils manipulent trois DDL simultanément pendant 30 % du temps seulement. Ces résultats semblent suggérer, qu’avec la technique RI, il n’y ait pas intégration complète des DDL de la tâche. De plus, les utilisateurs effectuent la tâche d’orientation en manipulant un DDL à la fois avec la technique MP.

IT	1 DOF	2 DOF	3 DOF
RI	15,71 %	49,28 %	28,64 %
MP	88,31 %	11,38 %	0,26 %
$p - value$	< 0,001	< 0,001	< 0,001

TABLE 4.3 – La proportion de temps de manipulation passé à manipuler un, deux et trois DDL.

4.4 Analyse et discussions

Les résultats présentés dans la section précédente semblent indiquer que la technique d’interaction a une influence significative sur la manière d’effectuer la tâche et sur les performances.

4.4.1 Performances

Les résultats que nous obtenons suggèrent que l’intégration des DDL ne conduit pas forcément aux meilleures performances. En effet, bien qu’en utilisant la technique MP, les utilisateurs décomposent la tâche, ils atteignent le même niveau de précision qu’avec la technique RI – ce qui est en opposition avec notre première hypothèse. Les résultats que nous obtenons sont meilleurs que ceux indiqués dans les études conduites par Poupyrev *et al.*, (6,8 degrés) [Poupyrev 00], Hinckley *et al.*, (7,07 degrés) [Hinckley 97] ou encore LaViola (3,9 degrés) [LaViola 08]. Nos résultats sont cohérents avec les résultats obtenus dans le cadre d’études en environnement immersif avec du *head tracking*. Ils confirment la suggestion faite par Hinckley, indiquant que, la précision de la rotation dépend plus des capacités des utilisateurs à percevoir et évaluer l’erreur en rotation, que du contrôle du geste [Hinckley 97]. Dans ce cas, nous ne pouvons indiquer clairement les effets de la décomposition de la tâche sur la précision des utilisateurs.

De manière similaire, les résultats obtenus concernant le temps moyen de réalisation suggèrent que la décomposition de la tâche est aussi efficace que l'intégration. En effet, le Tableau 4.1 montre que les performances sont similaires pour les deux techniques d'interaction. Ceci est un exemple dans lequel la décomposition n'introduit pas d'augmentation significative du temps de réalisation. C'est pourquoi, de manière similaire à l'étude effectuée par Poupyrev *et al.* [Poupyrev 00], nous avons décidé de regarder les temps nécessaires pour atteindre différents niveaux de précision. Dans cette étude, nous utilisons deux seuils différents. Le premier est un seuil de précision grossier – fixé à 18 degrés. Le second, fixé à 9 degrés, est considéré comme un seuil de précision fin.

Comme nous pouvons le voir dans le Tableau 4.2, les utilisateurs sont capables d'atteindre ces différents seuils plus rapidement avec la technique MP qu'avec la technique RI. Plus particulièrement, les utilisateurs atteignent le seuil de précision grossier 14,39 % plus rapidement avec la technique MP qu'avec la technique RI. En utilisant la technique MP, ils atteignent le seuil de précision fin 10,38 % plus rapidement qu'avec la technique RI. Ces résultats semblent indiquer que les utilisateurs sont capables d'effectuer la tâche plus rapidement avec la technique MP. Nous pensons que l'absence de différences significatives au niveau du temps de réalisation est due à l'utilisation d'une mauvaise méthode de filtrage pour la technique MP. Afin d'éviter des petites rotations parasites avec cette technique, nous avons utilisé un filtrage basé sur un seuil minimum de vitesse. L'utilisation de ce seuil, bien qu'évitant les rotations parasites, rend difficile la possibilité de faire de petits ajustements en fin de tâche. L'utilisateur passe alors un temps considérable à tenter d'ajuster finement l'orientation de l'objet.

Dans le cas d'orientations complexes et de faibles amplitudes, les utilisateurs sont capables d'atteindre les seuils de précision fin et grossier dans des temps similaires, ce qui indique que nous avons tort à propos de notre seconde hypothèse. Nous pensons qu'en utilisant la technique RI, les utilisateurs tireraient parti de l'intégration des DDL pour effectuer la tâche plus efficacement. Ceci peut être dû au fait que les utilisateurs n'intègrent que partiellement la tâche avec la technique RI.

Enfin, en considérant les orientations simples et de grandes amplitudes, les résultats montrent que les utilisateurs atteignent les deux seuils de précision plus rapidement avec MP qu'avec la technique RI. Dans le cas des orientations simples et de grandes amplitudes, les utilisateurs tirent profit de la décomposition – ce qui confirme notre troisième hypothèse. Ces résultats corroborent ceux obtenus dans l'étude menée par Chen *et al.* [Chen 88]. En effet, dans cette étude, nous pouvons voir que les utilisateurs effectuent les orientations simples plus rapidement avec les sliders (moins de 10 s) qu'avec la *Virtual Sphere* (presque 15 s).

La technique MP étant une technique similaire à la manipulation de sliders, nous proposons de reconsidérer les raisons expliquant pourquoi l'utilisation de sliders n'est pas appropriée à la tâche d'orientation [Chen 88]. La manipulation de sliders est souvent

décrite comme une technique lourde et inappropriée pour orienter un objet. L'hypothèse avancée ici, est qu'elle nécessite la décomposition d'une tâche à trois DDL en une succession de trois tâches à un DDL. Toutefois, elle nécessite également de devoir sélectionner le slider à manipuler avant de pouvoir orienter l'objet. L'ajout de sélections successives peut accroître significativement le temps de manipulation. C'est peut être plus particulièrement le cas, en fin de tâche, lorsque l'utilisateur veut effectuer de petites corrections. L'ajout des temps de sélection ainsi que la nécessité d'alterner entre une tâche de rotation et une tâche de sélection peut expliquer pourquoi leur utilisation n'est pas appropriée.

Pour résumer, les résultats que nous avons obtenus semblent indiquer que l'intégration des DDL n'offre pas toujours le meilleur niveau de performances. De plus, la décomposition de la tâche d'orientation peut être une stratégie efficace qui peut introduire de meilleurs temps de réalisation.

4.4.2 Coordination des degrés de liberté

Dans le Tableau 4.3, nous pouvons voir qu'en utilisant la technique RI, les utilisateurs manipulent les trois DDL simultanément pendant seulement 28 % de la tâche. Pendant environs 50 % du temps de manipulation, les utilisateurs combinent seulement deux DDL à la fois – ce qui confirme notre quatrième hypothèse. Ce résultat indique que pendant la tâche d'orientation, les utilisateurs ne sont capables d'intégrer que partiellement la manipulation des DDL. Une explication possible réside dans la complexité de la tâche et dans l'incapacité des utilisateurs à identifier le chemin optimal de l'orientation initiale à l'orientation cible. Les utilisateurs essayent peut être de décomposer la tâche en une succession d'orientations plus simples.

En utilisant la technique MP, les utilisateurs manipulent un DDL à la fois durant presque 90 % du temps de manipulation, ce qui confirme notre cinquième hypothèse. La technique facilite la décomposition de la tâche et les utilisateurs tirent profit de cette caractéristique pour décomposer la tâche à trois DDL en une succession de trois tâches à un DDL. Comme les utilisateurs ne sont pas capables d'identifier le chemin optimal entre l'orientation initiale de l'objet et l'orientation cible, ils essayent peut être de décomposer la tâche. Cette stratégie peut permettre d'effectuer la tâche en passant par des orientations spécifiques (comme des alignements). Cela peut expliquer les meilleurs résultats obtenus par la technique MP au niveau de TG et TF.

4.5 Conclusion

Dans cette étude, l'objectif était de comparer les performances offertes par une technique intégrant (RI) et une technique décomposant les DDL (MP) dans le cadre d'une tâche d'orientation dans un espace 3-D. Afin d'étudier formellement l'impact de la décomposition, nous utilisons la mesure NDC afin de quantifier le niveau d'intégration durant l'exécution de la tâche.

En utilisant cette mesure, nous pouvons voir que les utilisateurs décomposent la tâche d'orientation dans un espace 3-D en trois sous-tâches d'orientations à un DDL avec la technique MP. Cette technique facilite en effet la manipulation d'un DDL à la fois. Que se soit avec la technique RI ou la technique MP, les utilisateurs atteignent le même niveau de précision. Toutefois, avec la technique MP, les utilisateurs sont capables d'effectuer la tâche plus rapidement. Ces résultats sont en opposition avec les résultats précédents, indiquant que la technique d'interaction doit intégrer la manipulation des DDL afin d'obtenir les meilleures performances [Chen 88] [Jacob 94].

De plus, la mesure NDC nous a permis de mettre en avant le fait que les utilisateurs ne sont capables d'intégrer que partiellement la tâche d'orientation 3-D, même avec une technique permettant la manipulation simultanée de l'ensemble des DDL. Ceci peut être expliqué par la complexité de la tâche et l'impossibilité qu'éprouve l'utilisateur à identifier la rotation minimale à appliquer à l'objet manipulé pour qu'il corresponde à l'orientation de la cible. Cette impossibilité peut conduire l'utilisateur à utiliser des stratégies consistant à diviser une tâche complexe en sous-tâches plus simples.

Toutefois, la technique MP introduit deux facteurs pouvant influencer les performances : la décomposition de la tâche et l'introduction d'un support physique. Comme nous l'avons indiqué, la décomposition de la tâche dans la cas de la tâche d'orientation peut permettre de décomposer l'orientation 3-D en trois orientations 1-D. L'efficacité de cette décomposition peut être due à la complexité inhérente à la tâche. Qu'apporte la décomposition de la tâche dans le contexte d'une tâche moins complexe où l'utilisateur est capable d'identifier le chemin optimal ?

La technique MP introduit également un support physique au cours de la manipulation. Introduire un support physique peut être intéressant dans la mesure où celui-ci peut réduire la fatigue liée à la manipulation. En environnement de RV, les techniques proposées sont souvent basées sur une manipulation libre. Dans un contexte d'applications de modélisation géométrique, où les sessions de travail sont longues, cette fatigue peut être d'autant plus importante. Toutefois, l'introduction d'un support physique nécessite de décomposer une tâche de manipulation 3-D. Il est d'autant plus important d'évaluer l'influence de la décomposition dans des tâches de manipulations simples afin d'évaluer la pertinence de l'utilisation de techniques introduisant un support physique (et donc une décomposition). Ceci permettrait de rendre possible la mise en place de longues

sessions de manipulation dans notre environnement de travail.

C'est pourquoi nous proposons dans le chapitre suivant, de s'intéresser à l'influence de la décomposition et du support physique sur les performances et le confort de manipulation. Nous nous intéresserons plus particulièrement à la tâche de positionnement 3-D. Cette tâche, moins complexe que la tâche d'orientation, nous permettra de quantifier les apports respectifs de ces deux caractéristiques.

Décomposition et positionnement 3-D

Sommaire

5.1	Introduction	88
5.2	Protocole expérimental	89
5.2.1	Techniques comparées	89
5.2.2	Tâches	92
5.2.3	Procédure expérimentale	93
5.2.4	Mesures	94
5.2.5	Hypothèses	94
5.3	Résultats	95
5.3.1	Performances	95
5.3.2	Analyse comportementale	97
5.3.3	Évaluation subjective	100
5.4	Discussions	102
5.4.1	Décomposition en fin de tâche	102
5.4.2	Évaluation subjective	103
5.4.3	Précision	104
5.5	Conclusion	108

Nous proposons, dans cette seconde étude, de nous intéresser à l'influence de la décomposition de la tâche et du support physique dans une tâche de positionnement 3-D. Nous avons vu que dans le cadre de la tâche d'orientation 3-D, la décomposition peut permettre d'améliorer les performances des utilisateurs. Les gains observés peuvent être dus à la complexité inhérente à la tâche d'orientation. Nous proposons donc d'aller plus loin, en étudiant l'influence respective de la décomposition et de l'introduction d'une contrainte physique dans le cadre d'une tâche à priori de complexité moindre, le positionnement 3-D.

5.1 Introduction

Nous utilisons la simulation d'une interface tactile afin de décomposer la tâche de positionnement. En considérant l'ensemble des décompositions possibles, trois décompositions sont envisageables :

1. Positionnement dans le plan X-Y et positionnement selon la dimension Z (profondeur) ;
2. Positionnement dans le plan X-Z et positionnement selon la dimension Y (hauteur) ;
3. Positionnement dans le plan Y-Z et positionnement selon la dimension X (largeur).

La décomposition séparant la manipulation du paramètre X semble être la moins intéressante dans la mesure où cette dimension semble la plus simple à appréhender par les utilisateurs [Fiorentino 03]. Les deux solutions restantes semblent offrir une décomposition appropriée à la tâche de positionnement. Néanmoins, les performances obtenues par ces deux décompositions peuvent être différentes.

Nous pensons qu'en séparant la manipulation d'une des dimensions, la technique d'interaction va modifier la perception que va avoir l'utilisateur de la tâche de positionnement. Effectivement, la séparation d'une dimension peut avoir comme effet d'accroître la distance sémantique entre elle et les autres dimensions. En modifiant la distance sémantique entre les différentes dimensions, la technique modifie la structure perceptuelle de la tâche à effectuer. La séparation de la profondeur et la séparation de la hauteur sont deux décompositions pouvant induire des structures perceptuelles différentes. La décomposition, en modifiant la structure perceptuelle, peut donc avoir une influence sur les performances. Elle peut également avoir une influence sur la charge cognitive. La séparation du paramètre de profondeur, en proposant une décomposition appropriée aux difficultés des utilisateurs, peut permettre de réduire la charge cognitive par rapport à la séparation de la hauteur.

De plus, par rapport à une technique de manipulation directe, la simulation d'une interface tactile introduit un support physique au cours de la manipulation. En permettant un meilleur contrôle du geste, cette contrainte physique peut avoir une influence propre sur les performances des utilisateurs.

Dans l'étude que nous proposons, nous nous intéressons aux questions suivantes.

1. La décomposition a-t-elle une influence sur les performances des utilisateurs ? La séparation de la dimension de profondeur peut-elle conduire à une amélioration des performances plus importantes que celle de la hauteur ?
2. Nous savons que la tâche de positionnement est décomposée en deux phases : une

balistique et une de contrôle¹. Quels sont les apports de la décomposition dans la phase balistique et dans la phase de contrôle du positionnement ?

3. Le support physique a-t-il une influence sur les performances ? Des gains sont-ils envisageables si le support physique permet d'accroître le contrôle du geste de l'utilisateur ?

Nous décrivons maintenant le protocole expérimental que nous avons mis en place.

5.2 Protocole expérimental

Dans l'expérience mise en place, nous avons deux facteurs : la présence du support physique et le type de décomposition. Chacun de ces facteurs possède deux niveaux : avec / sans support physique, décomposition avec séparation de la hauteur / décomposition avec séparation de la profondeur. Nous utilisons un plan factoriel complet (*i.e.* nous utilisons l'ensemble des combinaisons possibles des facteurs) avec comme condition de référence la manipulation directe. Nous considérons les conditions suivantes :

- Manipulation directe (notée MD) : (Pas de support physique, Pas de décomposition) ;
- Manipulation directe avec séparation de la profondeur (notée MDIP) : (Pas de support physique, Décomposition X-Y + Z) ;
- Manipulation directe avec séparation de la hauteur (notée MDIH) : (Pas de support physique, Décomposition X-Z + Y) ;
- Tactile avec séparation de la profondeur (notée TIP) : (Support physique, Décomposition type X-Y + Z) ;
- Tactile avec séparation de la hauteur (notée TIH) : (Support physique, Décomposition type X-Z + Y).

Avec ces conditions, nous sommes à même de quantifier les apports respectifs du support physique (en comparant MDIP à TIP et MDIH à TIH) et de la décomposition (en comparant MD à MDIP et MDIH).

5.2.1 Techniques comparées

Pour chacune des conditions présentées précédemment, nous avons une technique d'interaction associée. Nous allons présenter ces différentes techniques. Dans l'ensemble des techniques présentées aucune technique de filtrage n'est appliquée et le CDG (présenté dans le Chapitre 2) est de 1. De plus, aucune des techniques utilisées au cours de

1. Nous rappelons que la phase balistique est la phase durant laquelle l'utilisateur se rapproche rapidement de la cible. La phase de contrôle est la phase durant laquelle l'utilisateur effectue des corrections afin de placer l'objet précisément.

ce protocole n'introduit une manipulation colocalisée².

Manipulation directe

Dans cette condition, l'utilisateur doit pincer l'index et le pouce pour définir la position de référence. En déplaçant la main, tout en maintenant les deux doigts pincés, l'utilisateur déplace l'objet.

Manipulation directe avec séparation de la profondeur

En pinçant l'index et le pouce, l'utilisateur peut déplacer l'objet dans le plan X-Y. Pour déplacer l'objet dans la profondeur, l'utilisateur doit pincer l'index et le majeur. Si l'utilisateur essaye de pincer l'ensemble des trois doigts, aucun des mouvements effectués n'est appliqué à l'objet. Nous obligeons donc l'utilisateur à décomposer la tâche.

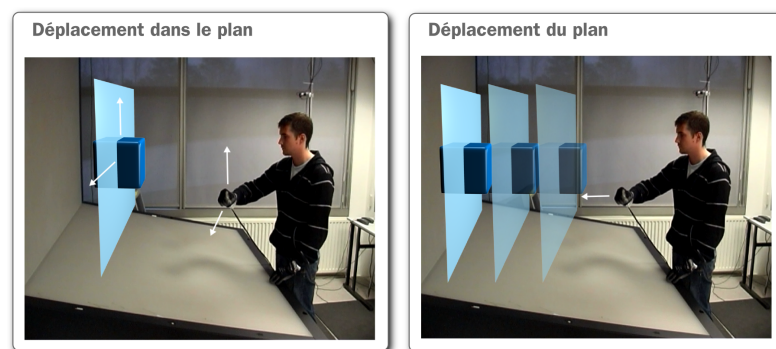


FIGURE 5.1 – Nous illustrons ci-dessus la relation entre les déplacements de la main et ceux appliqués à l'objet.

Cette condition impose une décomposition mais sans introduire de support physique puisque l'utilisateur doit effectuer les mouvements à main levée.

Manipulation directe avec séparation de la hauteur

En pinçant l'index et le pouce, l'utilisateur peut déplacer l'objet dans le plan X-Z. Pour déplacer l'objet selon la hauteur, l'utilisateur doit pincer l'index et le majeur. Si

2. Une manipulation est dite colocalisée lorsque le périphérique d'interaction est confondu avec l'objet virtuel manipulé. Lors d'un positionnement 3-D, une manipulation colocalisée implique que les positions de l'objet virtuel et du périphérique d'interaction soient confondues.

l'utilisateur essaye de pincer l'ensemble des trois doigts, aucun des mouvements effectués n'est appliqué à l'objet.

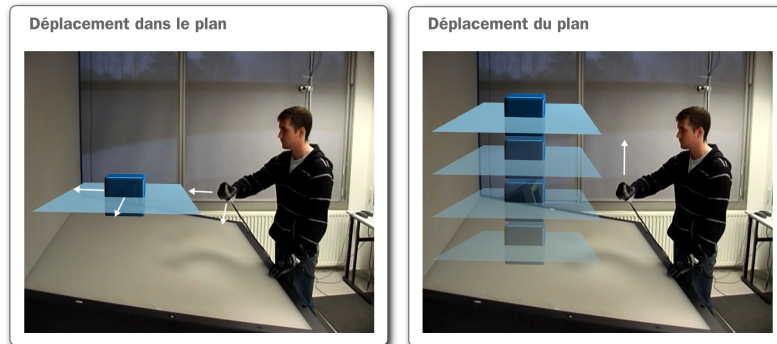


FIGURE 5.2 – Nous illustrons ci-dessus la relation entre les déplacements de la main et ceux appliqués à l'objet.

Comme pour la condition précédente, nous n'introduisons ici aucun support physique puisque l'utilisateur doit effectuer les mouvements à main levée.

Tactile avec séparation de la profondeur

En demandant à l'utilisateur d'aller toucher l'écran se trouvant à portée de main, nous introduisons un support physique en plus de la décomposition. Pour déplacer l'objet dans le plan X-Y, l'utilisateur doit glisser sa main tout en touchant l'écran avec l'index. En touchant l'écran avec le majeur, l'utilisateur peut déplacer l'objet selon la profondeur. La technique est illustrée en Figure 5.3.

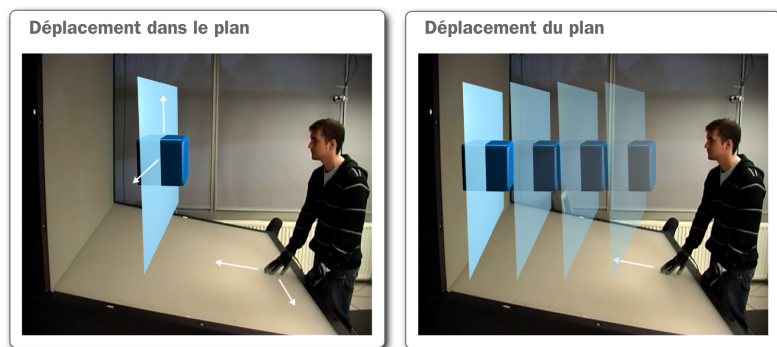


FIGURE 5.3 – Nous illustrons ci-dessus la relation entre les déplacements de la main et ceux appliqués à l'objet.

Tactile avec séparation de la hauteur

Comme pour la technique précédente, en demandant à l'utilisateur d'aller toucher l'écran se trouvant à portée de main, nous introduisons un support physique au cours de la manipulation. Pour déplacer l'objet dans le plan X-Z, l'utilisateur doit glisser sa main tout en touchant l'écran avec l'index. En touchant l'écran avec le majeur, l'utilisateur peut déplacer l'objet selon la hauteur. La technique est illustrée en Figure 5.4.

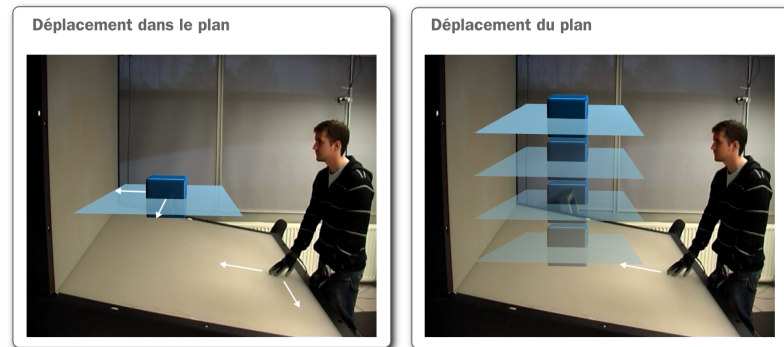


FIGURE 5.4 – Nous illustrons ci-dessus la relation entre les déplacements de la main et ceux appliqués à l'objet.

5.2.2 Tâches

La tâche est une tâche de **positionnement de précision**. Elle consiste à positionner un objet, représenté ici par une croix, dont les axes sont alignés sur les axes du repère du monde virtuel, dans une cible représentée par une sphère de rayon et de position fixe. La longueur des axes est égale au double du rayon de la sphère afin de permettre à l'utilisateur d'apprécier la précision selon chacune des dimensions. Une illustration est donnée en Figure 5.5.

La position initiale de la croix est définie de telle sorte que :

1. Elle se trouve toujours derrière la cible ;
2. Le segment joignant la position initiale et la cible soit une diagonale parfaite (*i.e.* la distance initiale selon chacune des dimensions soit la même).

Chaque sujet effectue un ensemble de six positionnements pour chacune des quatre diagonales. Parmi ces six positionnements, la moitié sont des positionnements de faibles amplitudes (pour chaque DDL, la distance initiale entre l'objet et la cible est de 30 cm). L'autre moitié sont des positionnements de grandes amplitudes (pour chaque DDL, la distance initiale entre l'objet et la cible est de 90 cm).

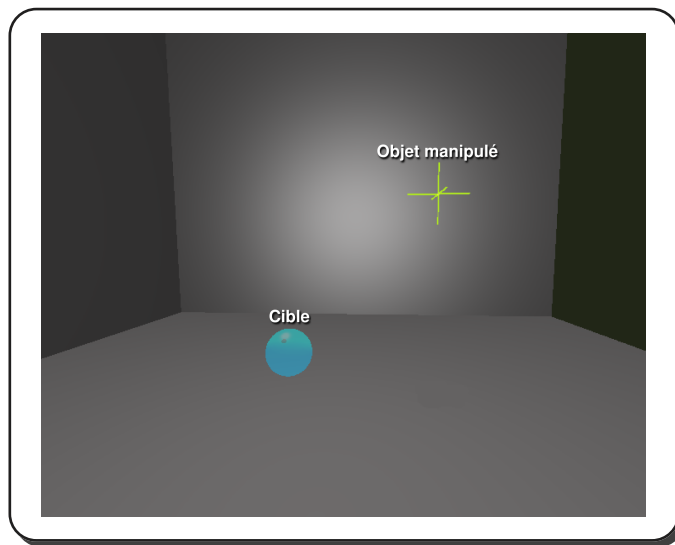


FIGURE 5.5 – Illustration du retour visuel associé à la tâche. La croix symbolise l'objet à manipuler. Le centre de la sphère constitue la cible à atteindre.

5.2.3 Procédure expérimentale

Quinze sujets volontaires ont participé à cette étude. Ces derniers ont différents niveaux d'expérience de manipulation en environnement de RV. Chaque sujet effectue l'ensemble des vingt quatre positionnements avec l'ensemble des cinq techniques proposées.

Dans un premier temps, le dispositif est présenté au sujet. Dans un second temps débute la phase de prise en main. La prise en main consiste en cinq blocs de cinq positionnements, un bloc par technique. Avant chaque bloc, la technique d'interaction est présentée oralement au sujet. Durant la phase d'apprentissage, aucune limite de temps n'est imposée. Il peut manipuler aussi longtemps qu'il le désire pour se familiariser à la fois avec l'environnement et les différentes techniques utilisées. Enfin, la phase de test débute. La consigne donnée aux utilisateurs est d'essayer d'atteindre une précision maximale dans un temps raisonnable.

Dans le cadre de ce protocole, nous avons utilisé un plan expérimental intra-sujet. Chaque sujet utilise donc les cinq techniques que nous souhaitons comparer. Chaque sujet effectue vingt quatre positionnements. Afin d'éviter un effet d'ordre, nous contre balançons cet effet en utilisant une répartition de l'ordre des conditions basée sur un carré latin³.

3. Un carré latin est un tableau carré de n lignes et n colonnes. Il contient n éléments distincts. Ce tableau est rempli de telle manière à ce que chaque ligne et chaque colonne ne contienne qu'un seul

5.2.4 Mesures

Dans le cadre de cette expérimentation, nous prenons les mesures suivantes :

- Temps de réalisation (T, en seconde) : temps écoulé entre le premier et le dernier mouvement de la croix ;
- Erreur euclidienne (E, en mm) : distance euclidienne à la fin de la tâche ;
- Erreur selon les DDL X, Y et Z (E_x , E_y , E_z , en mm) : erreur selon chacune de ces dimensions à la fin de la tâche ;
- Coordination (NDC) : calcul de la mesure NDC à partir de la vitesse de déplacement de la croix ;
- Pourcentage de temps passé à manipuler 1, 2 et 3 DDL : ce calcul est basé sur la métrique NDC ;
- Pourcentage de temps passé à manipuler 1, 2 et 3 DDL avant et après un seuil de précision : le seuil de précision utilisé est défini comme $5 \times E$. Ceci permet d'avoir un seuil prenant en compte la précision des différents utilisateurs.

À la fin de chaque expérimentation, l'utilisateur est invité à remplir un questionnaire NASA-TLX [Hart 88]. Ce questionnaire nous permet entre autres choses d'évaluer les charges mentale, physique ainsi que le niveau de frustration. De plus, à la fin de la session de tests, nous demandons aux utilisateurs d'effectuer une évaluation de leur préférence vis-à-vis de chacune des techniques sur une échelle de Likert [Likert 32] à 7 niveaux. Plus la note est élevée, plus la technique a été appréciée par les testeurs.

5.2.5 Hypothèses

Nous formulons les différentes hypothèses évoquées dans la Section 5.1.

Influence du support physique

H1 : Le support physique accroît la précision et réduit les temps de réalisation. En effet, le support physique peut permettre de supprimer des mouvements parasites et de fournir à l'utilisateur une information haptique passive (*i.e.* que l'information haptique n'est pas contrôlée et ne dépend que des propriétés physiques du dispositif d'interaction) par le biais du support physique.

H2 : Le support physique accroît le confort d'utilisation en réduisant la fatigue par rapport à une situation de manipulation sans contraintes physiques.

exemplaire de chaque élément. Nous utilisons un carré latin, dans lequel les éléments sont les conditions, afin de réduire un potentiel effet d'ordre de présentation des conditions. Pour le sujet i , $i \in [0, \text{nombre de sujet}]$, l'ordre de présentation sera celui de la $i \bmod n$ ligne du carré latin.

Intégration vs. séparation des DDL

H3 : La décomposition accroît la précision.

H4 : L'intégration diminue la précision mais réduit les temps de réalisation. En combinant l'ensemble des DDL, l'utilisateur est capable de tirer profit du parallélisme induit.

H5 : L'intégration réduit la durée de la phase balistique mais accroît la durée de la phase de contrôle par rapport à la décomposition avec ou sans support physique.

H6 : Le choix des correspondances entre les DDL manipulés et les DDL de l'objet a une influence sur les performances.

H7 : La séparation de la manipulation de la profondeur permet de mieux appréhender cette dimension et d'être plus précise que les autres techniques selon cette dimension.

H8 : La séparation de la hauteur permet de mieux appréhender cette dimension et d'être plus précise que les autres techniques selon cette dimension.

5.3 Résultats

Nous nous trouvons dans le cas d'un plan expérimental intra-sujet. Nous avons donc analysé les données au moyen d'une ANOVA à mesures répétées, à un facteur, la technique d'interaction. Lorsque la technique d'interaction semble avoir une influence sur la variable dépendante, nous utilisons une comparaison à l'aide d'un test de Student paire à paire en appliquant une correction de Benjamini et Hochberg.

5.3.1 Performances

L'évaluation des performances des utilisateurs porte sur les données de précision et le temps de réalisation.

Précision

Le Tableau 5.1 indique l'erreur moyenne pour chacune des techniques ainsi que l'erreur moyenne selon chacune des trois dimensions de l'espace 3-D. La technique d'interaction semble avoir un effet sur l'erreur euclidienne moyenne observée. Il semble y avoir une différence significative entre MD et les techniques séparant la profondeur (MD vs. MDIP : $p\text{-value} < 0,001$; MD. vs. TIP : $p\text{-value} < 0,001$). MDIP et TIP sont respectivement 14,26 % et 14,55 % plus précise que MD. Il n'y a toutefois aucune différence entre les techniques séparant la hauteur et MD (MD vs. MDIH : $p\text{-value} = 0,94$; MD. vs. TIH : $p\text{-value} = 0,71$). Il semble également qu'il y ait une différence significative

TI	E	E_x	E_y	E_z
MD	6,94 mm	1,64 mm	2,52 mm	5,29 mm
MDIH	6,92 mm	1,63 mm	2,64 mm	5,14 mm
TIH	6,77 mm	1,41 mm	2,40 mm	5,37 mm
MDIP	5,95 mm	1,48 mm	2,27 mm	4,35 mm
TIP	5,93 mm	1,43 mm	2,19 mm	4,45 mm
F	$F(4,51) = 4,63$	$F(4,51) = 2,22$	$F(4,51) = 2,94$	$F(4,51) = 3,68$
$\rho - value$	0,003	0,079	0,029	0,048

TABLE 5.1 – L'erreur moyenne pour chacune des techniques comparées.

entre les techniques séparant la profondeur et celles séparant la hauteur :

- MDIH vs. MDIP : $\rho - value < 0,001$;
- MDIH vs. TIP : $\rho - value < 0,001$;
- TIH vs. MDIH : $\rho - value = 0,0017$;
- TIH vs. TIP : $\rho - value = 0,0016$.

Les techniques MDIP et TIP sont en moyenne 13,22 % plus précises que les techniques MDIH et TIH.

Toutefois, nous n'observons aucune différence entre les techniques MDIH et TIH ainsi qu'entre les techniques MDIP et TIP (MDIH vs. TIH : $\rho - value = 0,71$; MDIP vs. TIP : $\rho - value = 0,94$). Au vu de ces résultats, il semblerait que seul l'introduction d'une décomposition de la tâche ait une influence sur la précision de l'utilisateur. Nous pouvons remarquer que pour les techniques séparant la profondeur, le gain en matière de précision est significatif en ce qui concerne la dimension isolée (MDIP vs. MD : $\rho - value = 0,003$; MDIP vs. MDIH : $\rho - value = 0,008$; MDIP vs. TIH : $\rho - value = 0,002$; TIP vs. MD : $\rho - value = 0,005$; TIP vs. MDIH : $\rho - value = 0,02$; TIP vs. TIH : $\rho - value = 0,003$). Ceci n'est pas le cas pour les techniques séparant la hauteur.

Temps de réalisation

L'analyse montre une influence statistiquement significative de la technique d'interaction sur T. Comme le montre le Tableau 5.2, l'introduction d'une décomposition de la tâche semble accroître T. Nous observons une augmentation de près de 50 % pour les techniques MDIH et TIH et une augmentation de 35,9 % pour les techniques MDIP et TIP. L'ensemble des différences sont statistiquement significatives ($\rho - value < 0,001$) exceptées les différences entre MDIH et TIH ($\rho - value = 0,97$) et entre MDIP et TIP ($\rho - value = 0,76$). Le type de décomposition semble avoir une influence sur les temps de réalisation.

TI	T
MD	11,24 s
MDIH	16,88 s
TIH	16,89 s
MDIP	15,19 s
TIP	15,36 s
<i>F</i>	$F(4,51) = 24,76$
$\rho - value$	$< 0,001$

TABLE 5.2 – Pour chacune des techniques utilisées les temps moyens d'exécution.

5.3.2 Analyse comportementale

Les mesures de coordination nous permettent de faire l'analyse des stratégies de positionnement employées par les sujets. Dans un premier temps, nous analysons la coordination moyenne des DDL tout au long de la tâche.

Coordination

TI	NDC
MD	2,15
MDIH	1,35
TIH	1,33
MDIP	1,37
TIP	1,36
<i>F</i>	$F(4,51) = 1617,29$
$\rho - value$	$< 0,001$

TABLE 5.3 – Coordination moyenne pour chaque technique.

Les données du Tableau 5.3 indiquent que le niveau de coordination atteint avec les différentes techniques est différent. Le Tableau 5.4 indique, plus précisément, le niveau de coordination pour chacune des techniques en donnant le pourcentage de temps où l'utilisateur manipule un, deux et trois DDL simultanément. La technique d'interaction a une influence sur l'ensemble de ces mesures. Excepté pour les différences entre MDIH et TIH ainsi qu'entre MDIP et TIP, l'ensemble des différences sont significatives pour les pourcentages de temps passés à manipuler un et deux DDL (les $p - value$ sont inférieures à 0,001). Particulièrement, nous pouvons noter qu'en ce qui concerne le pourcentage de temps passé à manipuler un et deux DDL, les différences entre les tech-

niques séparant la profondeur et les techniques séparant la hauteur sont significatives ($p - value < 0,001$ pour l'ensemble des comparaisons). En ce qui concerne le pourcentage de temps passé à manipuler trois DDL, dans la mesure où le pourcentage de temps est nul pour les quatre techniques décomposant la tâche, seules les différences entre MD et l'ensemble des autres techniques sont significatives (avec des $p - value < 0,001$).

TI	1 DDL	2 DDL	3 DDL
MD	12,39 %	51,89 %	34,39 %
MDIH	76,82 %	22,94 %	0,0 %
TIH	77,88 %	22,20 %	0,0 %
MDIP	69,12 %	30,72 %	0,0 %
TIP	69,68 %	30,35 %	0,0 %
F	$F(4,51) = 675,84$	$F(4,51) = 100,34$	$F(4,51) = 346,47$
$p - value$	$< 0,001$	$< 0,001$	$< 0,001$

TABLE 5.4 – Moyenne des proportions de temps de manipulation de un, deux et trois DDL par rapport au temps de manipulation total.

Décomposition en fin de tâche

TI	1 DDL		2 DDL		3 DDL	
	Avant	Après	Avant	Après	Avant	Après
MD	11,57 %	15,68 %	47,30 %	56,89 %	41,12 %	27,43 %
MDIH	75,80 %	78,69 %	23,99 %	21,22 %	0,0 %	0,0 %
TIH	76,22 %	80,12 %	23,73 %	19,85 %	0,0 %	0,0 %
MDIP	65,22 %	81,64 %	34,61 %	18,30 %	0,0 %	0,0 %
TIP	65,95 %	81,97 %	34,01 %	18,01 %	0,0 %	0,0 %

TABLE 5.5 – Moyenne des proportions de temps de manipulation de un, deux et trois DDL en début / fin de tâche par rapport au temps passé en début / fin de tâche.

La distinction entre la manipulation à proximité et à distance de l'objet est basée sur la précision finale atteinte. Le Tableau 5.5 indique le pourcentage de temps moyen passé à manipuler un, deux et trois DDL à proximité de la cible. Pour chacune des techniques, nous avons comparé les pourcentages de temps avant et après le seuil de précision à l'aide d'un test de Student afin de voir si la différence observée est significative. Pour l'ensemble des comparaisons effectuées – exceptée pour les comparaisons des pourcentages de temps passé à manipuler trois DDL pour les techniques décomposant la tâche de positionnement – les $p - value$ sont inférieures à 0,001.

Nous constatons qu’une fois passé le seuil de précision, l’utilisateur essaie davantage de décomposer le positionnement avec la manipulation directe. De plus, nous pouvons noter la forte proportion de temps passé à manipuler un DDL en fin de tâche. Pour les techniques MDIH, TIH, MDIP et TIP, il semblerait que l’utilisateur décompose le positionnement dans le plan de manipulation (*i.e.* le plan horizontal pour les techniques MDIP et TIP, le plan vertical pour les techniques MDIH et TIH). Ce résultat semble indiquer qu’en fin de tâche l’utilisateur décompose le positionnement 3-D en trois positionnements à un DDL.

TI	1 DDL		2 DDL	
	Avant	Après	Avant	Après
<i>MDIH</i>	45,74 %	52,26 %	54,26 %	47,73 %
<i>TIH</i>	47,81 %	55,33 %	52,18 %	44,67 %
<i>MDIP</i>	26,88 %	41,99 %	73,12 %	58,01 %
<i>TIP</i>	28,32 %	42,36 %	71,68 %	57,63 %
<i>F</i>	$F(3,38) = 14,48$	$F(3,38) = 7,24$	$F(3,38) = 14,48$	$F(3,38) = 7,24$
<i>p – value</i>	< 0,001	< 0,001	< 0,001	< 0,001

TABLE 5.6 – Proportion de temps passée à manipuler un et deux DDL dans le plan de manipulation avant et après le seuil de précision.

La technique d’interaction semble avoir une influence sur la décomposition dans le plan de manipulation. Les données du Tableau 5.6 montrent que l’intégration dans le plan de manipulation est plus importante pour les techniques séparant la profondeur que pour les techniques séparant la hauteur. Nous pensons que cela provient du fait que les sujets tentent de manipuler la profondeur de façon indépendante. Pour cela, les sujets isolent la manipulation de la profondeur dans le plan de manipulation pour les techniques MDIH et TIH.

Il semblerait également que la décomposition dans le plan de manipulation soit plus importante en fin de tâche. Nous constatons, en effet, que les comparaisons des pourcentages de temps avant et après le seuil de précision sont statistiquement significatives. Pour l’ensemble des comparaisons effectuées les *p – value* sont inférieures à 0,001. Ce résultat suggère que même avec les techniques décomposant la tâche, l’utilisateur décompose dans le plan de manipulation en fin de tâche.

Nombre de *clutch*

Le *clutch* est une phase durant laquelle l’utilisateur repositionne le périphérique d’interaction, sans repositionner l’objet manipulé, afin d’effectuer un nouveau geste d’interaction. La technique d’interaction a une influence sur le nombre de *clutch*. Comme

TI	(Nombre de <i>clutch</i> (NC))
<i>MD</i>	5,67
<i>MDIH</i>	10,80
<i>TIH</i>	11,60
<i>MDIP</i>	9,08
<i>TIP</i>	10,58
<i>F</i>	$F(4,51) = 90,51$
$\rho - value$	$< 0,001$

TABLE 5.7 – Le nombre moyen de *clutch* par tâche.

nous pouvons le noter sur les valeurs indiquées dans le Tableau 5.7, le nombre moyen de *clutch* par tâche semble nettement réduit pour la technique MD comparé aux autres techniques d'interaction. Il semblerait également que le nombre de *clutch* soit réduit pour les techniques sans support physique (*MDIH* vs. *TIH* : $\rho - value < 0,001$; *MDIP* vs. *TIP* : $\rho - value = 0,002$).

5.3.3 Évaluation subjective

À partir de notre questionnaire, nous avons extrait l'évaluation pour chacun des critères intervenant dans le questionnaire NASA-TLX ainsi que l'évaluation subjective de la préférence des utilisateurs.

Préférence

TI	PSU
<i>MD</i>	4,80
<i>MDIH</i>	3,73
<i>TIH</i>	3,67
<i>MDIP</i>	5,67
<i>TIP</i>	5,80
<i>F</i>	$F(4,56) = 13,68$
$\rho - value$	$< 0,001$

TABLE 5.8 – La note moyenne, notée PSU (Préférence Subjective de l'Utilisateur), représentant la préférence des utilisateurs.

Le Tableau 5.8 indique la moyenne des évaluations subjectives de la préférence pour

chacune des techniques. Seules les différences entre MDIP et TIP ainsi qu'entre MDIH et TIH ne sont pas significatives. Les autres différences sont significatives avec une $p - value < 0,05$. Nous pouvons constater que l'évaluation de MD est meilleure que celles de MDIH et TIH. À l'opposé, les techniques MDIP et TIP ont une meilleure évaluation que la technique MD (MD vs. MDIP : $p - value = 0,023$; MD vs. TIP : $p - value = 0,01$). Ces résultats semblent indiquer que la séparation de la profondeur des autres dimensions offre une manipulation appréciée des utilisateurs. Nous pensons que celle-ci offre un moyen de mieux appréhender les difficultés liées à la gestion de ce paramètre.

Charge cognitive

Le questionnaire NASA-TLX nous permet d'avoir une évaluation selon différents critères. En combinant l'ensemble de ces facteurs, il est possible d'estimer la charge cognitive (notée CC). Les résultats pour les différents critères sont donnés dans le Tableau 5.9. Les résultats en ce qui concerne CC sont donnés dans le Tableau 5.10.

TI	MD	PD	TD	P	E	F
MD	2,40	5,13	3,53	4,27	3,67	3,40
MDIH	4,67	4,67	3,73	4,20	4,80	5,27
TIH	4,33	2,80	3,00	3,40	4,00	4,93
MDIP	2,00	3,80	2,73	2,67	2,73	2,20
TIP	2,33	2,67	2,27	2,07	2,53	2,13
<i>F</i>	$F(4,56) = 9,8$	$F(4,56) = 5,8$	$F(4,56) = 2,7$	$F(4,56) = 5,2$	$F(4,56) = 5,2$	$F(4,56) = 9,5$
<i>p - value</i>	$< 0,001$	$< 0,001$	0,03	$< 0,001$	$< 0,001$	$< 0,001$

TABLE 5.9 – *Mental Demand (MD), Physical Demand (PD), Temporal Demand (TD), Performance (P), Effort (E) et Frustration Level (FL).*

La présence de support physique semble réduire significativement la demande physique (pour PD, MDIH vs. TIH : $p - value = 0,02$; MDIP vs. TIP : $p - value = 0,18$). Le support physique semble donc pouvoir accroître le confort de l'utilisateur au cours de la manipulation. Cela montre que l'utilisation de techniques basées sur la manipulation libre peut induire une fatigue importante pouvant être significativement réduite par l'introduction d'une contrainte physique.

La décomposition semble avoir une influence sur la charge cognitive. La décomposition de la tâche en un positionnement X-Z ainsi qu'un positionnement selon l'axe Y complexifie la tâche. Ce n'est pas le cas pour la décomposition X-Y plus Z, qui elle, ne semble pas accroître la complexité de la tâche. Les différences entre les techniques séparant la profondeur et les autres techniques sont significatives avec une $p - value$

inférieure à 0,05 (exceptée la différence entre MDIP et MD : $p\text{-value} = 0,078$). De plus, la décomposition semble avoir également une influence significative sur le niveau de frustration.

TI	CC
<i>MD</i>	3,77
<i>MDIH</i>	4,83
<i>TIH</i>	4,02
<i>MDIP</i>	2,67
<i>TIP</i>	2,31
<i>F</i>	$F(4, 56) = 7,5$
$p\text{-value}$	$< 0,001$

TABLE 5.10 – L'estimation de la charge cognitive.

5.4 Discussions

L'étude statistique que nous avons effectuée montre que le choix de la technique a une influence sur la précision, les temps de réalisation et l'évaluation subjective. Nous allons à présent discuter ces résultats de façon plus détaillée.

5.4.1 Décomposition en fin de tâche

Les utilisateurs, ont, à de nombreuses reprises, fait la remarque que la décomposition séparant le paramètre de profondeur est plus naturelle que la décomposition séparant la hauteur. Cette tendance se traduit, au niveau de l'évaluation, par une nette préférence des utilisateurs pour les techniques MDIP, TIP et MD. Ce constat apparaît très nettement dans le Tableau 5.8. Ce résultat semble confirmer l'hypothèse que, les techniques séparant la profondeur ont une structure perceptuelle plus proche de la représentation mentale que se font les utilisateurs de la tâche.

Un utilisateur nous a également fait la remarque suivante : « *J'ai l'impression que la décomposition m'aide en fin de tâche mais j'aimerais pouvoir manipuler un DDL à la fois* ». La proportion de temps passé à manipuler un DDL à la fois semble confirmer cela. Nous pouvons remarquer que pour les techniques décomposant la tâche, l'utilisateur manipule un DDL à la fois pendant environs 72,8 % du temps. L'utilisateur, passant une grande partie de son temps en phase de contrôle, nous pensons qu'en phase finale du positionnement, l'utilisateur essaie de manipuler chacun des DDL individuellement. Les mesures semblent confirmer cette hypothèse. Pour l'ensemble des techniques, la

décomposition est plus importante à proximité de la cible. Concernant plus particulièrement la technique de manipulation directe, nous constatons que :

1. Une fois à proximité de l'objet, la proportion du temps passé à manipuler un DDL augmente de 4,11 points ;
2. Une fois à proximité de l'objet, la proportion du temps passé à manipuler trois DDL diminue de 13,69 points.

Même avec la manipulation directe, l'utilisateur essaie de décomposer le positionnement 3-D en fin de tâche.

La coordination des DDL dans le plan pour les techniques MDIP, TIP, MDIH et TIH varie en fonction de la phase de positionnement.

- Proportion moyenne de temps passé à manipuler un DDL pour MDIH et TIH avant et après le seuil de précision : 46,8 % / 53,8 % ;
- Proportion moyenne de temps passé à manipuler deux DDL pour MDIH et TIH avant le seuil de précision : 53,2 % / 46,2 % ;
- Proportion moyenne de temps passé à manipuler un DDL pour MDIP et TIP avant le seuil de précision : 27,6 % / 42,2 % ;
- Proportion moyenne de temps passé à manipuler deux DDL pour MDIP et TIP avant le seuil de précision : 72,4 % / 57,8 %.

Nous pouvons remarquer qu'une fois à proximité de la cible, le pourcentage de temps passé à manipuler un DDL à la fois augmente significativement. Cela semble indiquer que l'utilisateur essaie de décomposer la manipulation dans le plan. En fin de tâche, l'utilisateur n'effectue pas une tâche de type 2 + 1 DDL mais une tâche du type 3×1 DDL.

De plus, nous pouvons remarquer que dans le plan de manipulation, en fin de tâche, la décomposition est en moyenne plus importante pour les techniques séparant la hauteur (environ 55 % du temps passé à manipuler un DDL à la fois) que pour les techniques séparant la profondeur (environ 42 % du temps passé à manipuler un DDL à la fois). Encore une fois, il semblerait que la séparation du paramètre de profondeur soit un besoin pour les utilisateurs.

Nous proposons à présent d'étudier l'évaluation subjective des utilisateurs.

5.4.2 Évaluation subjective

L'analyse de l'évaluation subjective de la préférence des utilisateurs montre une nette préférence pour les techniques TIP et MDIP. Les techniques séparant la hauteur se sont vues attribuer des notes inférieures. La décomposition de la tâche a été perçue par les utilisateurs comme un handicap dans la phase d'approche, en revanche lors de la phase

de positionnement de précision, la décomposition a été perçue comme un avantage. Toutefois, de nombreuses remarques ont été faites sur le fait que la séparation du paramètre de profondeur semble bien plus naturelle que la séparation du paramètre de hauteur. C'est pour cela que les techniques TIH et MDIH ont été sanctionnées au moment de l'évaluation.

Par rapport à MD, les techniques MDIP et TIP ont été mieux notées. Cela semble confirmer l'adéquation de la séparation de la profondeur par rapport à la tâche de positionnement 3-D. Bien que perçue comme plus adaptée en phase balistique, l'utilisation de MD en phase de contrôle a souvent été décrite comme frustrante et peu précise.

Le questionnaire de la NASA permet d'évaluer la charge de travail, que nous utilisons ici comme un estimateur de la charge cognitive. Initialement, nous pensions que l'introduction d'une décomposition augmenterait la charge cognitive, notamment, en introduisant une complexification de la phase de planification.

Au contraire, nous constatons que cela dépend de la décomposition utilisée. Encore une fois, la séparation de la profondeur ne semble pas introduire d'augmentation de la charge cognitive par rapport à la manipulation directe. Ce résultat peut traduire une adéquation de la structure perceptuelle de la technique à la tâche effectuée. Les techniques séparant la hauteur semblent accroître de manière significative la charge de travail. Nous pensons que ce résultat est dû au manque d'intuitivité de la séparation de la hauteur et à son inadéquation aux difficultés de la tâche. Cela se traduit par un niveau de frustration très élevé, qui est en moyenne de 5,1. La profondeur étant la dimension la plus complexe à appréhender dans un environnement 3-D, les utilisateurs semblent tirer profit de sa séparation pour la résolution de la tâche et cela se traduit par un gain au niveau des performances.

L'évaluation subjective semble indiquer une adéquation de la séparation de la profondeur durant la tâche de positionnement 3-D. Nous analysons maintenant les performances obtenues par les différentes techniques.

5.4.3 Précision

Les résultats obtenus semblent indiquer que le type de décomposition utilisé a une influence significative sur la précision des utilisateurs. La manipulation de la profondeur indépendamment des autres dimensions accroît la précision. Ceci n'est pas le cas lorsque la hauteur peut être manipulée indépendamment des autres dimensions.

Décomposition

Nos résultats montrent que la décomposition séparant la hauteur (techniques MDIH et TIH) n'est pas adaptée à la tâche de positionnement. En effet, cette décomposition n'accroît pas la précision des utilisateurs. En revanche, les techniques MDIP et TIP montrent un gain significatif. Ce résultat est en concordance avec des travaux menés précédemment par Fiorentino *et al.* [Fiorentino 03], montrant notamment que, parmi les trois dimensions de l'espace, la hauteur n'est pas la dimension la plus difficile à appréhender. Cette même étude montre que la profondeur est la dimension la plus difficile à manipuler. En décomposant la tâche de positionnement 3-D en un positionnement 1-D plus 2-D, l'utilisateur peut se concentrer sur la manipulation de la profondeur. La séparation de ce paramètre peut alors expliquer les gains observés en précision pour les techniques MDIP et TIP. Ceci est confirmé par l'analyse de la précision par dimension. L'ensemble des précisions est donné dans le Tableau 5.1. Nous pouvons noter le gain en précision de placement selon la profondeur (représentée par la précision selon l'axe Z) pour les techniques séparant la profondeur.

Ce résultat semble indiquer que l'intégration des DDL de la tâche n'est pas forcément la solution offrant les meilleures performances en matière de précision. Nous obtenons de meilleures performances avec des techniques ayant une structure perceptuelle séparable (MDIP et TIP) qu'avec une technique ayant une structure perceptuelle intégrable (MD) dans le cas du positionnement 3-D (tâche ayant, a priori, une structure intégrable). Nos résultats sont en opposition avec les conclusions données à l'issue de l'étude menée par Jacob *et al.* [Jacob 94]. En effet, Jacob *et al.* concluent que l'utilisation de techniques intégrant les DDL est la solution offrant les meilleures performances en termes de précision et temps de réalisation sur des tâches intégrales.

Notre hypothèse, pour unifier ces résultats, est que la structure perceptuelle de la tâche de positionnement varie en fonction de la phase de réalisation. Dans une tâche à contrôle final, comme la tâche de positionnement 3-D, nous pensons que la structure perceptuelle de la tâche est spécifique à la phase dans laquelle se trouve l'utilisateur. Dans un premier temps, lors de la phase balistique, la structure perceptuelle est intégrable. La manipulation, des trois DDL à la fois, est appropriée à cette phase de la tâche qui consiste à se rapprocher grossièrement de la cible. Toutefois, durant la phase de contrôle, l'utilisateur a besoin de précision, la manipulation simultanée des trois DDL ne semble plus appropriée. L'utilisateur peut vouloir faire des ajustements selon chacune des dimensions séparément (et particulièrement selon la profondeur). La structure perceptuelle n'est, dans ce cas, plus intégrable, mais séparable. L'utilisateur ne conçoit plus la tâche de positionnement 3-D comme un tout, mais comme des sous-tâches distinctes et indépendantes. Le fort pourcentage de temps passé à manipuler un DDL à la fois en fin de tâche pour les techniques MDIH, MDIP, TIH et TIP semble indiquer que l'utilisateur décompose la tâche 3-D en trois sous-tâches à un DDL. De nombreux utili-

sateurs nous ont indiqué que les techniques séparant la profondeur sont plus adaptées en fin de tâche alors que la manipulation directe, est elle, plus adaptée pour le rapprochement grossier. L'analyse de la coordination semble confirmer cette hypothèse. En effet, la séparation de la profondeur se traduit (pour les techniques MDIH et TIH) par une décomposition plus importante dans le plan de manipulation.

Nous pouvons également noter que la décomposition séparant la profondeur induit des gains de performances, ce qui n'est pas le cas lorsque la hauteur est séparée des autres dimensions. Ce résultat est intéressant puisqu'il semble indiquer que le gain en matière de performance ne provient pas d'un meilleur contrôle de la dimension isolée. En effet, nous observons une précision accrue selon l'axe de profondeur pour les techniques séparant la profondeur. Toutefois, nous n'observons aucune amélioration pour les techniques séparant la hauteur selon l'axe Y. Dans ce cas, les gains observés ne proviennent pas d'un meilleur contrôle moteur. Nous pensons que la séparation de la profondeur offre une meilleure adéquation à la tâche que la séparation de la hauteur. En d'autres termes, la structure perceptuelle des techniques MDIP et TIP est plus proche de la structure perceptuelle de la tâche que les techniques MDIH et TIH. L'évaluation subjective que nous avons présentée semble confirmer ce résultat.

Support physique

Parmi les hypothèses de ce travail, nous pensions que l'introduction d'un support physique pourrait aider les utilisateurs à mieux contrôler leurs gestes, notamment en supprimant des mouvements parasites dus à d'éventuels tremblements de la main. Toutefois, un utilisateur nous a fait la remarque suivante : « *Même avec la présence du support physique, j'ai l'impression qu'il y a toujours des petits mouvements parasites qui me gênent en fin de tâche* ». C'est ce que semblent indiquer les résultats. Nous n'observons aucune différence entre les techniques introduisant un support physique et celles qui nécessitent une manipulation libre.

De plus, le support physique permet d'introduire une information haptique passive au cours de la manipulation. Nous pensions que cette information, en enrichissant l'information visuelle, aurait pu aider à corriger la position de l'objet durant la phase de contrôle. Toutefois, il semblerait que ce ne soit pas le cas. Ce résultat semble confirmer celui de Meyer *et al.* [Meyer 88]. Les résultats de leur étude indiquent qu'en phase de contrôle, l'information visuelle est primordiale pour l'évaluation de la précision.

Toutefois, nous pouvons noter que le support physique permet de réduire significativement la fatigue. Des techniques basées sur la simulation d'une interface tactile seraient donc adaptées, notamment lors de longues phases de manipulation. La réduction de la demande physique sera d'autant plus significative que la durée de manipulation est importante. Les gains observés ici se font sur une session de travail d'au plus une

heure, ce qui reste relativement peu comparé à la longueur des phases de manipulation que nous pouvons rencontrer dans le cadre d'applications de modélisation géométrique.

Temps

Au niveau des temps de réalisation, les techniques introduisant une décomposition induisent des temps de réalisation plus importants que MD. La technique MD permet de manipuler les trois dimensions de l'espace simultanément, ce qui permet de suivre une trajectoire directe sur la cible. Ces résultats sont d'autant moins étonnants que nous nous sommes volontairement placés dans le cas de positionnements à trois DDL, qui constitue le pire des cas pour les techniques décomposant la tâche par rapport à une technique de manipulation directe. Les écarts de temps observés sont donc dus au fait qu'une séparation des DDL impose une distance à parcourir plus importante. De plus, comme nous l'indique le Tableau 5.7, le nombre de *clutch* est bien plus important pour les techniques MDIP, TIP, MDIH et TIH. L'utilisateur passe donc beaucoup de temps à repositionner sa main pour effectuer un nouveau déplacement de l'objet. En facilitant le parcours de grandes distances, grâce à une fonction de transfert non-linéaire comme PRISM, ce phénomène aurait pu être amoindri.

Un deuxième facteur permet d'expliquer cette différence. Le temps d'exécution est défini comme le temps entre le premier et le dernier déplacement de l'objet. Toutefois, nous avons pu constater que l'utilisateur passe plus de temps à analyser la tâche avec les techniques MDIP, TIP, MDIH et TIH qu'avec la technique MD. Nous avons pu remarquer que ce phénomène intervient essentiellement en fin de tâche, lorsque l'utilisateur entre dans la phase de contrôle. Il passe alors du temps à identifier les dimensions à corriger pour atteindre le maximum de précision. Il semblerait que ce phénomène intervienne dans une moindre mesure avec la technique MD.

Nous pouvons également observer une différence entre les techniques séparant la profondeur ainsi que les techniques séparant la hauteur. Avec les techniques séparant la profondeur, les utilisateurs sont capables d'accomplir la tâche en moyenne 2,2 s plus rapidement qu'avec les techniques séparant la hauteur. En effet, pour les techniques MDIH et TIH, la manipulation dans le plan horizontal ne permet pas de séparer la profondeur. Un déplacement en profondeur implique généralement un déplacement latéral et inversement. Pour ces techniques, les ajustements en profondeur sont plus difficiles à réaliser. Le résultat que nous observons alors peut provenir du fait que l'utilisateur a plus de difficultés à positionner l'objet en profondeur.

L'introduction d'une décomposition accroît significativement les temps d'exécution. Bien que ce résultat soit exagéré par le fait que la tâche effectuée soit à trois DDL, il semblerait qu'introduire une décomposition durant l'ensemble de la tâche ne soit pas adapté au positionnement. La possibilité de pouvoir se déplacer dans les trois dimensions de

l'espace en même temps semble être un atout de poids, surtout en phase d'approche de l'objet.

5.5 Conclusion

Cette étude semble montrer plusieurs choses :

- Dans une tâche à contrôle final, la décomposition peut améliorer la précision des utilisateurs par rapport à une technique de manipulation directe. Cette décomposition doit être adaptée aux besoins des utilisateurs. Notamment, dans le cas du positionnement, il semblerait que seule la décomposition séparant la profondeur permette d'accroître la précision des utilisateurs. N'observant aucun gain en matière de précision de positionnement selon le paramètre Y pour les techniques séparant la hauteur, le gain observé n'est peut être pas dû uniquement à un meilleur contrôle moteur du geste mais plutôt à une meilleure adéquation de la technique à la tâche. Cela suggérerait que la structure perceptuelle de la tâche n'est pas forcément intégrable, spécialement en fin de tâche ;
- La décomposition n'est pas adaptée dans la phase d'approche de l'objet où l'utilisateur souhaite pouvoir manipuler l'ensemble des DDL de la tâche pour se rapprocher rapidement de la cible. Toutefois, elle semble adaptée en fin de tâche lorsque l'utilisateur souhaite faire des corrections selon des dimensions spécifiques ;
- Dans une tâche à contrôle final, l'introduction d'un support physique ne semble pas accroître les performances des utilisateurs.

Grâce à cette deuxième étude, nous pouvons mieux appréhender les difficultés de manipulation liées aux tâches à contrôle final. Il semblerait que, dans ce type de tâche, la décomposition puisse être une solution intéressante permettant d'améliorer le contrôle de l'utilisateur. Le support physique, quant à lui, ne semble pas être un facteur essentiel.

Nous souhaitons à présent nous intéresser à l'utilisabilité de la simulation d'une interface tactile dans le cadre de tâches à contrôle continu. Dans ce type de tâche, il n'est pas possible d'introduire une quelconque décomposition de la tâche. Toutefois, le support physique peut ici apporter un meilleur contrôle du geste. Nous nous intéresserons, dans le chapitre suivant, à l'influence du support physique mais également à l'utilisation d'une fonction de transfert basée sur la vitesse, dans une tâche à contrôle continu, la tâche de suivi de trajectoires.

Fonction de transfert, contrainte physique et tâche à contrôle continu

Sommaire

6.1	Introduction	110
6.2	Protocole expérimental	111
6.2.1	Techniques comparées	111
6.2.2	Tâche	112
6.2.3	Procédure expérimentale	113
6.2.4	Mesures	114
6.2.5	Hypothèses	114
6.3	Résultats	115
6.3.1	Erreurs	115
6.3.2	Temps de réalisation	116
6.3.3	Évaluation subjective	116
6.4	Analyse	117
6.4.1	Performances	117
6.4.2	Confort d'utilisation	119
6.5	Conclusion	120

Dans les études que nous avons menées précédemment, nous nous sommes intéressés exclusivement aux tâches à contrôle final. Les résultats que nous avons obtenus semblent montrer que, même en utilisant une technique de manipulation directe, les utilisateurs décomposent partiellement les DDL. Dans ce contexte, nous avons pu mettre en avant les bénéfices que peuvent apporter la décomposition ainsi que le support physique, aussi bien en matière de performances qu'en matière de confort d'utilisation.

Néanmoins, dans le cadre théorique que nous proposons, nous distinguons les tâches à contrôle final des tâches à contrôle continu. Pour les tâches à contrôle final, seuls le début et la fin de la tâche sont importants. Pour les tâches à contrôle continu, l'ensemble de la trajectoire effectuée est importante. Ainsi, les bénéfices apportés par le support physique dans des tâches à contrôle final, peuvent être différents de ceux que nous pourrions observer dans des tâches à contrôle continu. Nous nous proposons donc d'étudier l'influence du support physique introduit par la simulation d'une interface tactile dans ce type de tâche. Nous nous intéressons également à l'utilisation d'une fonction de transfert basée sur la vitesse de déplacement pour l'amélioration du contrôle dans une tâche à contrôle continu.

6.1 Introduction

Le suivi de trajectoires constitue l'exemple le plus commun illustrant la notion de tâches à contrôle continu. Dans un espace 3-D, les tâches à contrôle continu ne peuvent être décomposées. Une trajectoire diagonale nécessite un mouvement diagonal du périphérique d'interaction, quelle que soit la technique d'interaction. Une étude de l'influence de la décomposition n'a donc ici aucun sens.

De plus, habituellement, les courbes 3-D ne sont pas définies directement dans l'espace 3-D. Elles sont généralement créées en utilisant plusieurs courbes 2-D. Cohen *et al.* [Cohen 99] définissent une courbe 3-D en manipulant une courbe dans l'espace écran ainsi qu'une courbe dite, **courbe d'ombre**. Cette courbe d'ombre est définie comme la projection de la courbe sur un plan considéré comme étant le « sol ». En manipulant la courbe ainsi que sa projection, l'utilisateur peut créer des courbes tri-dimensionnelles.

Partant de ces considérations, nous proposons de mettre en place des techniques simulant l'utilisation d'interfaces tactiles pour des tâches de suivi de trajectoires. La simulation d'une interface tactile introduit un support physique tout au long de la manipulation. Ce support physique peut permettre d'apporter un meilleur contrôle du geste dans ce type de tâche.

Habituellement, dans un environnement immersif ou semi-immersif, les courbes 2-D sont définies à l'aide de contraintes virtuelles. Par exemple, dans le modèleur proposé par Grossman *et al.* [Grossman 02], l'utilisateur définit une courbe dans un plan virtuel 2-D à l'aide d'un périphérique 3-D. La position sur le plan virtuel est définie comme la projection de la position du périphérique d'interaction sur le plan. Toutefois, d'après les auteurs, l'utilisation d'une telle méthode ne semble pas offrir un niveau de contrôle suffisant afin de définir précisément des courbes.

De plus, certaines des difficultés sont spécifiques aux caractéristiques de la trajectoire. Liu *et al.* [Liu 10] proposent de caractériser la difficulté de la tâche de suivi de

trajectoires en tenant compte, à la fois de la largeur du chemin à suivre, ainsi que de la courbure locale de la trajectoire. Localement, plus la courbure est élevée, plus le suivi est difficile à réaliser précisément. Dans ce cas, une adaptation de la fonction de transfert peut avoir une influence sur les performances. C'est pourquoi, en plus de l'influence du support physique, nous proposons d'étudier l'influence d'un filtrage du mouvement basé sur la vitesse de déplacement sur la précision du suivi. Le filtrage est identique à celui utilisé dans la méthode PRISM [Frees 05]. Dans PRISM, la vitesse instantanée sert comme un indicateur des intentions des utilisateurs : vitesse élevée → manipulation grossière, vitesse réduite → manipulation fine. Le principe reste similaire pour le suivi de trajectoires :

- Dans les portions linéaires de la trajectoire, l'utilisateur peut avoir tendance à manipuler rapidement ;
- Dans les portions de la trajectoire où la courbure est élevée, l'utilisateur peut avoir tendance à ralentir son mouvement pour accroître le contrôle de son geste.

La méthode PRISM semble donc appropriée pour des tâches de suivi de trajectoires. Nous évaluerons donc, en plus de l'apport du support physique, l'influence d'un tel filtrage sur les performances des utilisateurs.

6.2 Protocole expérimental

Dans cette étude, nous comparons les contraintes virtuelles et les contraintes physiques dans une tâche à contrôle continu, le suivi de trajectoires. Nous nous intéressons également dans le cadre de cette expérience à l'influence sur les performances d'un filtrage du mouvement basé sur la vitesse de déplacement de l'objet.

6.2.1 Techniques comparées

Dans ce protocole expérimental, nous avons trois conditions :

1. Contrainte virtuelle (CV) : l'utilisateur ne dispose ici d'aucun support physique. Les mouvements sont effectués dans l'espace 3-D. Les mouvements du curseur sont toutefois contraints au plan dans lequel se trouve la trajectoire ;
2. Simulation d'une interface tactile (SIT) : l'utilisateur effectue le suivi de trajectoires en utilisant la surface de l'écran se trouvant à portée de main comme support. Les déplacements 2-D effectués sur la surface de l'écran sont alors appliqués au curseur manipulé ;
3. Simulation d'une interface tactile avec filtrage (SITF) : la condition de manipulation est similaire à celle de SIT. La manipulation du curseur se fait en utilisant l'écran se trouvant à portée de main comme support. Les déplacements

effectués sont filtrés à l'aide d'une fonction de transfert avant d'être appliqués au curseur. La fonction de transfert est définie par : $D_{\text{curseur}} = D_{\text{périphérique}} \times \text{Min}(1, \frac{\text{Vitesse instantanée}}{0,6})$. D_{curseur} définit le déplacement effectif du curseur. $D_{\text{périphérique}}$ constitue le déplacement effectué par le périphérique dans le plan horizontal. La vitesse instantanée est la vitesse de déplacement du périphérique entre son précédent rafraîchissement et le temps courant. La constante 0,6 a été fixée de manière empirique lors d'une étude préliminaire de telle sorte que le contrôle soit maximal.

Une illustration des situations de manipulation est donnée en Figure 6.1.

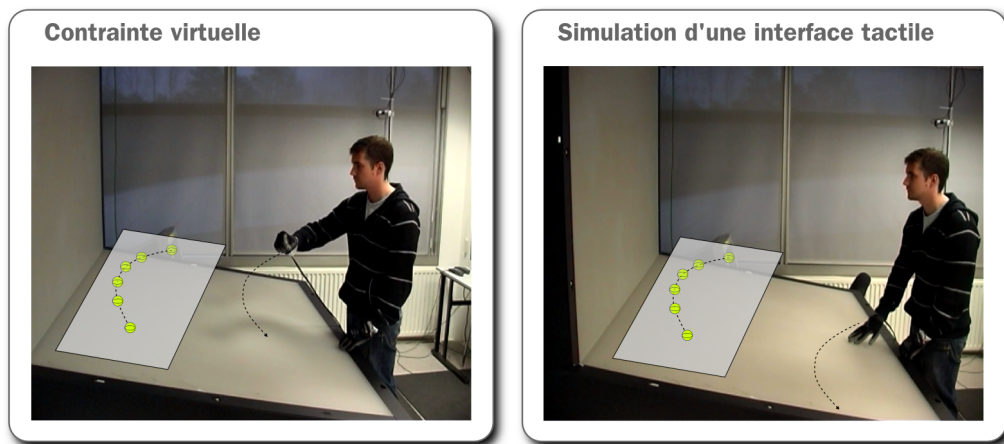


FIGURE 6.1 – Une illustration de deux des trois conditions de manipulation : CV et SIT.

6.2.2 Tâche

Pour chacune des conditions décrites ci-dessus, il est demandé à l'utilisateur de suivre une trajectoire. Cette trajectoire se trouve dans un plan horizontal, à une hauteur fixe dans l'environnement. La tâche consiste à déplacer un curseur – représenté par une sphère semi-transparente – de l'extrémité gauche à l'extrémité droite de la trajectoire. L'objectif est de suivre au plus près la trajectoire tout au long de la tâche.

Trois types de trajectoires sont utilisées. Chacune de ces trajectoires correspond à un niveau de difficulté :

- Simple ;
- Moyenne ;
- Difficile.

Les trajectoires associées à chacun des niveaux de difficulté sont données en Figure

6.2.

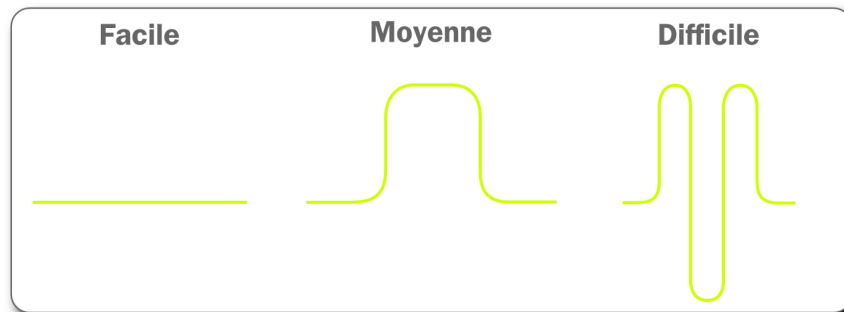


FIGURE 6.2 – Les trajectoires pour chaque niveau de difficulté.

Ces courbes sont présentées à l'utilisateur selon deux niveaux d'éloignement : proche et éloigné. Dans la condition proche, la trajectoire se trouve à proximité de l'utilisateur. Dans la condition éloignée, la courbe est translatée de 1,6 m selon l'axe de profondeur.

6.2.3 Procédure expérimentale

Douze sujets ont participé à cette étude. Comme pour les études précédentes, le protocole débute par une présentation du dispositif ainsi que des périphériques d'interaction. Une fois la présentation terminée, le protocole débute par une phase d'apprentissage. Durant cette phase, les différentes conditions sont présentées à l'utilisateur au travers d'un bloc de trois suivis de trajectoires par technique. Cette phase est essentielle, notamment afin de comprendre le fonctionnement des capteurs de pression.

Une fois la phase d'apprentissage terminée, débute la phase de test à proprement parlé. Nous utilisons ici un plan expérimental intra-sujet. L'ensemble des sujets effectue les tâches de suivi dans les trois conditions. La consigne pour l'ensemble des tâches est de suivre la trajectoire le plus précisément possible. Pour chacune des trois conditions, l'utilisateur effectue un bloc de douze suivis de trajectoires correspondant à deux suivis de trajectoires pour chacune des trois courbes combinées avec chacune des deux distances de manipulation. Les trajectoires apparaissent dans l'ordre suivant de manière répétée : facile, moyenne, difficile. Le facteur distance de manipulation est présenté de manière aléatoire. L'ordre des techniques est contre balancé en utilisant un ordonnancement en carré latin pour éviter un effet d'ordre.

6.2.4 Mesures

Pour chacun des suivis de trajectoires, nous enregistrons les mesures suivantes :

- Temps de réalisation (TRE, en secondes) : temps nécessaire à l'utilisateur pour déplacer le point d'une extrémité à l'autre de la courbe ;
- Erreur moyenne (EM, en mm) : écart moyen entre la position du curseur et la position la plus proche de la courbe ;
- Erreur moyenne en sections linéaires (EML, en mm) : écart moyen entre la position du curseur et la position la plus proche de la courbe dans les sections linéaires de la courbe ;
- Erreur moyenne en sections curvilignes (EMC, en mm) : écart moyen entre la position du curseur et la position la plus proche de la courbe dans les sections curvilignes de la courbe.

À la fin de l'expérimentation, l'utilisateur remplit le questionnaire donné en Annexe B. L'objectif de ce questionnaire est de recueillir les impressions des utilisateurs dans les différentes conditions. Il est constitué de cinq questions visant à évaluer les cinq critères suivants :

1. L'intuitivité de la technique ;
2. Le niveau de contrôle offert par la technique ;
3. La précision que permet d'atteindre la technique ;
4. Le niveau de fatigue induit par la technique ;
5. La préférence des utilisateurs.

6.2.5 Hypothèses

H1 : le support physique permet d'améliorer la précision du suivi de trajectoires. Le support physique permet d'enrichir l'information visuelle par une information haptique passive. Cette information peut être utilisée pour une meilleure correction du geste utilisateur. De plus, le support physique peut permettre d'avoir un meilleur contrôle moteur du geste.

H2 : le support physique permet des gains en matière de précision plus importants lorsque la manipulation se fait à distance. L'introduction d'un support physique permet de bénéficier, au cours de l'interaction, d'une information haptique passive. Cette information peut venir compléter l'information visuelle surtout lorsque cette dernière se trouve altérée lors d'une manipulation distante.

H3 : le support physique n'influe pas sur les temps de réalisation.

H4 : le filtrage permet des gains en matière de précision dans les sections curvilignes de

la trajectoire. Le filtrage permet un contrôle adapté du geste dans les sections curvilignes de la trajectoire.

H5 : en augmentant les déplacements à effectuer dans l'espace moteur, le filtrage augmente significativement les temps de réalisation par rapport aux techniques n'introduisant aucun filtrage.

6.3 Résultats

Le plan d'expérience utilisé dans cette étude est un plan intra-sujet. Nous utilisons donc une ANOVA à mesures répétées à un facteur, la condition de manipulation. Nous utilisons comme test à posteriori une comparaison paire à paire à l'aide d'un test de Student avec une correction de Benjamini et Hochberg.

6.3.1 Erreurs

TI	EM	EML	EMC
CV	9,44 mm	9,42 mm	8,59 mm
SIT	7,72 mm	7,70 mm	7,67 mm
SITF	7,87 mm	7,88 mm	6,67 mm
<i>F</i>	$F(2, 21) = 10,11$	$F(2, 21) = 9,99$	$F(2, 22) = 4,66$
$\rho - value$	$< 0,001$	$< 0,001$	0,02

TABLE 6.1 – Les erreurs moyennes pour chacune des techniques comparées.

Comme l'indiquent les données se trouvant dans le Tableau 6.1, pour chaque mesure d'erreur, la technique d'interaction a une influence significative. Pour EM, seule la différence entre SIT et SITF n'est pas significative. Les différences entre CV et SIT ainsi que CV et SITF sont significatives ($CV vs. SIT$: $\rho - value = 0,004$; $CV vs. SITF$: $\rho - value = 0,005$). Il semblerait donc que le support physique permette une amélioration de la précision des utilisateurs dans la tâche de suivi de trajectoires.

Le constat est identique pour EML : la différence entre SIT et SITF n'est pas significative, les différences entre CV et les deux autres techniques sont significatives. Les $\rho - value$ sont les suivantes : $CV vs. SIT$: $\rho - value = 0,004$ et $CV vs. SITF$: $\rho - value = 0,006$. Toutefois, les résultats ne sont pas les mêmes pour EMC. Seule la différence entre CV et SITF est significative avec une $\rho - value = 0,007$. Les autres différences ne sont pas significatives ($CV vs. SIT$: $\rho - value = 0,14$; $SIT vs. SITF$: $\rho - value = 0,14$).

6.3.2 Temps de réalisation

TI	TRE
CV	26,5 s
SIT	29,6 s
SITF	36,96 s
<i>F</i>	$F(2, 20) = 9,93$
$\rho - value$	0,001

TABLE 6.2 – Pour chacune des techniques utilisées, le temps moyen de réalisation.

La technique d'interaction a une influence significative sur les temps de réalisation, La différence observée entre CV et SIT n'est pas significative ($\rho - value = 0,18$), CV et SIT permettent donc d'effectuer le suivi de trajectoires dans un laps de temps similaire, Toutefois, la différence entre SITF et les autres techniques sont significatives ($SITF vs. CV : \rho - value < 0,001$; $SITF vs. SIT : \rho - value = 0,002$), Il semblerait donc que le filtrage basé sur le mouvement augmente de manière significative les temps de réalisation,

6.3.3 Évaluation subjective

TI	Q1	Q2	Q3	Q4	Q5
CV	6,50	4,00	3,58	5,00	4,58
SIT	6,58	6,25	6,08	2,83	6,50
SITF	4,58	5,33	5,33	3,66	4,58
<i>F</i>	$F(2, 36) = 0,19$	$F(2, 36) = 8,19$	$F(2, 36) = 9,34$	$F(2, 36) = 7,75$	$F(2, 36) = 6,14$
$\rho - value$	< 0,001	0,002	0,001	0,003	0,008

TABLE 6.3 – Évaluation subjective par les utilisateurs.

Pour l'ensemble des questions, la technique d'interaction a une influence significative sur l'évaluation donnée par les utilisateurs. Les résultats sont donnés dans le Tableau 6.3. Les $\rho - value$ pour chacune des comparaisons sont données dans le Tableau 6.4. Pour Q1, la différence entre CV et SIT n'est pas significative. Les différences entre ces deux techniques et SITF sont toutefois significatives. En ce qui concerne Q2, les différences entre CV et les deux autres techniques sont significatives. Toutefois, bien qu'une tendance semble apparaître, la différence entre SIT et SITF n'est pas significative. Les résultats observés pour Q3 et Q4 sont similaires, nous observons une différence significative entre CV et les autres techniques mais pas entre SIT et SITF. Finalement, pour

Q5, les différences entre SIT et les autres techniques sont significatives alors que la différence entre SITF et CV ne l'est pas.

TI	Q1	Q2	Q3	Q4	Q5
<i>CV vs. SIT</i>	0,859	0,002	< 0,001	0,005	0,005
<i>CV vs. SITF</i>	< 0,001	0,049	0,007	0,066	1,000
<i>SIT vs. SITF</i>	< 0,001	0,135	0,205	0,20	0,005

TABLE 6.4 – p – value pour chacune des comparaisons.

6.4 Analyse

Nous allons à présent analyser les résultats que nous avons obtenus dans cette étude.

6.4.1 Performances

Erreurs

Au vu des résultats obtenus, il semblerait que le support physique améliore de manière significative la précision des utilisateurs durant le suivi de trajectoires. En observant les résultats concernant EM, nous constatons une diminution de 22,3 % de l'erreur moyenne entre CV et SIT. Toutefois, entre SIT et SITF, la différence observée n'est pas significative. Ce résultat semble indiquer que le support physique permet d'accroître la précision des utilisateurs. Ce n'est pas le cas pour le filtrage des mouvements basé sur la vitesse de déplacement.

Ces résultats confirment les impressions des utilisateurs. Les questions Q2 et Q3 ont pour objectifs de permettre à l'utilisateur d'évaluer les techniques respectivement, en matière d'adaptation à la tâche et de performances. Pour ces deux questions, nous pouvons remarquer que les différences entre SIT et SITF ne sont pas significatives, alors que, les différences entre SIT et les deux autres techniques sont significatives. Il semblerait donc que les utilisateurs perçoivent le support physique comme le facteur prépondérant influençant leurs performances.

Par rapport au gain que nous pouvons observer, nous envisageons deux hypothèses :

1. Le support physique permet de stabiliser le geste d'interaction. En supprimant des tremblements liés à la manipulation sans contraintes physiques, le geste de l'utilisateur est alors plus stable. Le gain observé proviendrait donc d'un contrôle moteur plus approprié ;

2. Le support physique permet, à travers l'information haptique passive qu'il véhicule, de compléter l'information visuelle et ainsi permet un meilleur contrôle du curseur.

L'hypothèse qui nous paraît ici la plus plausible est la seconde. En effet, dans l'étude précédente, nous avons pu constater que dans une tâche de positionnement de précision, le support physique n'a aucune influence sur la précision. Il semblerait donc que, même en fin de tâche, le support physique ne permette pas un contrôle moteur plus précis. Dans le cadre de cette expérimentation, il est donc possible qu'il en soit de même. Le gain que nous observons peut donc provenir du second facteur. En enrichissant l'information visuelle d'une information haptique passive, le support physique peut permettre une meilleure perception des mouvements du curseur. Dans ce cas, l'utilisateur est alors plus à même de corriger son geste, permettant ainsi un suivi de trajectoires plus précis.

Afin d'analyser les données plus finement, nous proposons de distinguer l'erreur moyenne dans les sections linéaires de la courbe, de l'erreur moyenne dans les sections curvilignes. En analysant les résultats obtenus pour EML et EMC, nous constatons qu'en matière de précision :

- Dans les sections linéaires de la courbe, le support physique semble apporter un gain significatif ;
- Dans les sections linéaires de la courbe, le filtrage basé sur la vitesse ne semble pas apporter de gain significatif ;
- Dans les sections curvilignes de la courbe, le support physique ne semble pas apporter de gain significatif ;
- Dans les sections curvilignes de la courbe, le filtrage basé sur la vitesse semble apporter un gain significatif.

L'introduction d'un support physique semble donc permettre un meilleur contrôle du geste de manipulation. L'amélioration du contrôle se traduit notamment par un gain significatif de précision lors de manipulations dans des sections linéaires. Dans les sections curvilignes, il semblerait que l'introduction du seul support physique ne soit pas suffisante. L'ajout du filtrage basé sur la vitesse semble, dans ce cas, nécessaire afin d'apporter un gain significatif au niveau de la précision. Nous pensons que, dans les sections curvilignes, l'information visuelle est plus difficile à remplacer que dans les sections linéaires. En effet, pour les sections linéaires, il est plus facile pour l'utilisateur d'identifier localement la direction à suivre. Il est donc peut être plus facile d'exploiter l'information haptique passive afin de corriger le mouvement de suivi. Dans les sections curvilignes, l'estimation de la courbure semble fortement dépendre de l'information visuelle [Legault 07]. Dans ce cas, l'information portée par d'autres canaux sensori-moteurs peut être négligée.

Le support physique semble donc apporter un meilleur contrôle du geste, permettant à l'utilisateur d'accroître sa précision lors de la tâche de suivi de trajectoires. Coupler

l'introduction d'un support physique avec un filtrage du mouvement basé sur la vitesse permet d'accroître le contrôle du geste, spécialement lorsque localement, la courbure du tracé est importante.

Temps de réalisation

Nous l'avons vu, le support physique permet d'accroître la précision des utilisateurs. Il semblerait que cette amélioration ne se fasse pas au prix d'un allongement des temps de réalisation. En effet, les données du Tableau 6.2 indiquent que la différence entre CV et SIT n'est pas significative.

Ce n'est pas le cas lorsque l'on introduit le filtrage basé sur la vitesse. Cette méthode de filtrage permet d'accentuer le ralentissement du curseur lorsque l'utilisateur diminue la vitesse de manipulation. La distance totale parcourue, physiquement par la main dans l'espace moteur, est donc accrue. Ceci a pour conséquence immédiate d'allonger les temps de réalisation.

D'après les remarques des différents sujets, bien que le filtrage ait été perçu comme un atout dans les portions curvilignes, celui-ci a été jugé comme non adapté dans les portions linéaires. Dans ce type de portion de la trajectoire, le filtrage n'apporte aucun gain en matière de performances. Ceci est confirmé par l'analyse de la précision dans les différentes portions des courbes. Il n'est donc pas adapté dans ce cas de figure. Il ne fait qu'introduire une augmentation des temps de manipulation sans introduire de gain en matière de précision.

6.4.2 Confort d'utilisation

Jusqu'à présent, l'analyse des données semble indiquer clairement, que l'introduction d'un support physique dans ce type de tâche permet d'atteindre de meilleures performances. Les données quantitatives semblent donc mettre en avant la supériorité des contraintes physiques par rapport aux contraintes virtuelles dans des tâches à contrôle continu. Nous nous intéressons maintenant à l'évaluation subjective des utilisateurs.

Avec une moyenne aux alentours de 6,5 à la première question, les techniques CV et SIT sont jugées comme intuitives par les utilisateurs. Ce n'est pas le cas pour la technique SITF. L'introduction du filtrage semble réduire significativement l'intuitivité de la manipulation. De nombreux utilisateurs ont indiqué que le comportement du curseur avec cette technique est souvent difficile à comprendre.

Les techniques introduisant un support physique sont jugées comme plus adaptées à la tâche de suivi de trajectoires que la technique basée sur les contraintes virtuelles. De plus, les utilisateurs considèrent que ces deux techniques permettent de suivre précisé-

ment les trajectoires. Le support physique est donc considéré par les utilisateurs comme un facteur important permettant d'accroître la précision de la manipulation. Toutefois, la technique SITF n'a quant à elle pas été jugée comme plus adaptée ou plus précise que la technique SIT. Le filtrage n'a donc pas été perçu comme un atout dans la manipulation.

L'introduction du support physique semble réduire la fatigue. La réduction observée entre CV et SIT est de l'ordre de 45 %. De plus, bien que la manipulation soit plus longue avec le filtrage des mouvements, il n'a toutefois pas introduit un quelconque accroissement de la fatigue. Ces deux résultats semblent donc indiquer que le support physique permet de réduire significativement la fatigue.

Finalement, nous pouvons constater que les utilisateurs préfèrent nettement la technique SIT par rapport aux techniques SITF et CV. Ici encore, le résultat indique l'inadéquation du filtrage à la tâche. Nous pensons que ce n'est pas le filtrage à proprement parler qui est inadapté, mais plutôt la mise en oeuvre qui, dans ce cadre, ne semble pas convenir. Ceci est notamment dû aux ralentissements du curseur dans les portions linéaires qui ont frustré les utilisateurs.

6.5 Conclusion

Dans le chapitre précédent, nous nous sommes intéressés à l'influence du support physique sur les performances des utilisateurs lors de tâches à contrôle final. Les résultats alors obtenus semblent indiquer que l'introduction d'un support physique dans des tâches à contrôle final n'a aucune influence sur la précision que les utilisateurs peuvent atteindre. Au vu des résultats de la présente étude, le constat ne semble pas identique pour des tâches à contrôle continu.

Dans la tâche de suivi de trajectoires, il semblerait que le support physique permette un meilleur contrôle du geste de manipulation. En effet, par rapport à une situation où l'utilisateur ne bénéficie d'aucune contrainte physique, avec le support physique, les utilisateurs sont capables de suivre plus précisément des trajectoires de complexités différentes. Ce gain est d'autant plus appréciable qu'il semblerait que l'introduction d'un support physique n'accroisse pas les temps de réalisation.

Nous nous sommes également intéressés ici à l'influence du filtrage basé sur la vitesse de déplacement sur les performances des utilisateurs. Le bilan est mitigé. Il semblerait que le filtrage puisse permettre d'améliorer la précision des utilisateurs dans des portions curvilignes du suivi de trajectoires. Toutefois, aucun gain n'a été observé dans les sections linéaires. De plus, l'utilisation de ce type de filtrage nécessite de parcourir des distances plus importantes dans l'espace moteur. La conséquence logique d'une telle demande est une augmentation significative des temps de réalisation. Il semblerait donc qu'un filtrage basé sur la vitesse ne soit pas adapté. Néanmoins, un filtrage basé sur

la courbure du mouvement pourrait quant à lui être plus approprié. Ce dernier éviterait de rallonger inutilement les temps de manipulation dans des portions linéaires et permettrait un suivi plus précis dans les zones courbes.

La simulation d'une interface tactile semble donc appropriée à la tâche de suivi de trajectoires. Elle permettrait une amélioration de la précision du geste de l'utilisateur. Par rapport à une situation de manipulation libre, le support physique introduit par la simulation d'une interface tactile permettrait d'accroître le confort de manipulation en réduisant grandement la fatigue.

Recommandations de conception pour l'interaction 3-D

À présent, nous allons faire le bilan des résultats que nous avons obtenus dans les différentes études que nous avons menées. Ce bilan nous permet de définir des recommandations de conception de techniques d'interaction. Par la suite, nous utilisons ces recommandations afin de définir de nouvelles techniques d'interaction. Ces techniques sont mises en oeuvre et utilisées dans le cadre de deux applications de modélisation.

7.1 Bilan

L'objectif de ces trois premières études était de quantifier les apports de deux des caractéristiques importantes de la simulation d'une interface tactile pour l'interaction 3-D :

1. La contrainte physique introduite par le support physique utilisé ;
2. La décomposition de la tâche introduite par la contrainte physique, qui nécessite de décomposer l'ensemble des tâches de dimensionnalités supérieures à trois.

Dans les deux premières études que nous avons menées, nous avons quantifié la coordination des utilisateurs dans des tâches à contrôle final, le positionnement et l'orientation dans un espace 3-D. Nous avons pu constater qu'en utilisant une technique similaire à une technique de manipulation directe, permettant à l'utilisateur de combiner facilement la manipulation de l'ensemble des DDL de la tâche, les utilisateurs décomposent partiellement la tâche. Les résultats de ces deux premières études semblent indiquer que la décomposition des tâches à contrôle final, induite par la simulation d'interface tactile, n'est pas forcément un inconvénient. Elle peut même constituer un atout dans la réalisation de la tâche. Toutefois, lorsqu'il est nécessaire de décomposer la tâche, se pose la

question du choix de la décomposition.

Durant la seconde étude que nous avons conduite, nous nous sommes également intéressés à l'influence du choix de la décomposition ainsi qu'à l'influence du support physique sur les performances des utilisateurs dans une tâche à contrôle final. Dans une tâche de positionnement 3-D, nous avons comparé deux choix de décomposition, la première isolant le paramètre de hauteur, la seconde isolant le paramètre de profondeur. Les résultats que nous obtenons semblent indiquer que le choix de la décomposition a une influence significative sur les performances des utilisateurs. En effet, l'isolation du paramètre de hauteur ne semble pas appropriée à la tâche de positionnement. Cette décomposition n'introduit aucun gain en matière de précision. Néanmoins, lorsque la décomposition isole le paramètre de profondeur, la précision atteinte par les utilisateurs est meilleure. Au travers des évaluations subjectives des utilisateurs, il ressort que cette décomposition rend la manipulation intuitive. La décomposition peut donc permettre de résoudre des problèmes de manipulation, intrinsèques à l'interaction 3-D, comme la manipulation de la profondeur.

Toutefois, dans cette deuxième étude, nous avons pu constater que l'introduction d'un support physique n'a aucune influence sur les performances des utilisateurs. Le seul bénéfice apparent introduit par le support physique est une réduction de la fatigue. Le confort de manipulation se voit donc accru par rapport à une situation de manipulation libre. De plus, l'introduction d'une décomposition dans la tâche de positionnement 3-D accroît de manière importante les temps de réalisation par rapport à une technique de manipulation directe. Une telle augmentation indique clairement que ce type de techniques est inapproprié à cette tâche. Ce n'est toutefois pas le cas pour l'orientation 3-D. Les utilisateurs semblent tirer profit de la décomposition, tout au long de la tâche. Ceci peut être dû à la complexité inhérente à la tâche d'orientation. Cette complexité peut favoriser l'utilisation de stratégies basées sur la décomposition.

Enfin, dans une troisième étude, nous avons étudié l'influence du support physique dans une tâche à contrôle continu. Pour ce type de tâche, nous ne nous intéressons pas à la décomposition puisque celle-ci n'est pas possible. Nous nous sommes également intéressés à l'influence d'une méthode de filtrage du mouvement basé sur la vitesse de déplacement sur les performances. Les résultats semblent indiquer que l'introduction d'un support physique permet d'accroître la précision des utilisateurs dans ce type de tâche. Nous pensons que la présence d'une information haptique passive permet à l'utilisateur une meilleure perception du geste. En conséquence, l'utilisateur est capable de mieux contrôler son mouvement et d'atteindre de meilleures précisions. En ce qui concerne la méthode de filtrage, elle semble appropriée dans les sections curvilignes des trajectoires. Elle semble permettre d'avoir un meilleur contrôle lorsque la courbure du mouvement est importante. Toutefois, le filtrage étant basé sur la vitesse de déplacement, l'effet de filtrage se fait également ressentir dans les sections linéaires du suivi de

trajectoires. Dans ce cas de figure, le filtrage semble inapproprié. Il réduit alors significativement l'amplitude des déplacements réalisés par le curseur et rend la manipulation difficile.

7.2 Recommandations pour des tâches d'interaction 3-D

Tâches à contrôle final

Recommandation 01 : pour les tâches à structure perceptuelle intégrable, les techniques doivent permettre de combiner les DDL de la tâche durant la phase d'approche. Il est possible alors que l'utilisateur puisse tirer profit du parallélisme afin de réaliser rapidement cette phase.

Recommandation 02 : pour les tâches à structure perceptuelle intégrable, si l'objectif de la tâche est la précision, les techniques doivent permettre de séparer les DDL de la tâche durant la phase de contrôle. La séparation peut alors permettre d'accroître le contrôle des utilisateurs et d'améliorer la précision.

Recommandation 03 : pour les tâches à structure perceptuelle intégrable, si l'objectif de la tâche n'est pas la précision, les techniques ne doivent pas nécessiter de séparer les DDL de la tâche durant la phase de contrôle.

Tâches à contrôle continu

Recommandation 04 : si l'objectif est la précision, les techniques doivent introduire une contrainte physique au cours de la réalisation de la tâche.

Recommandation 05 : si l'objectif est la précision, les techniques doivent introduire une fonction de transfert non-linéaire dans les parties curvilignes.

Recommandation 06 : si les sessions de manipulation sont longues, les techniques d'interaction doivent introduire un support physique afin de réduire la fatigue.

7.3 Exploitations des recommandations

Nous utilisons ces résultats afin de mettre en place de nouvelles techniques d'interaction, dont l'objectif est d'offrir un meilleur contrôle et une meilleure précision dans la manipulation. Nous proposerons deux techniques d'interaction.

DIOD : Dynamic Decomposition and Integration of Degrees of Freedom

Pour cette technique, nous utilisons les recommandations 01, 02 et 03. L'utilisateur souhaite pouvoir intégrer la manipulation de plusieurs DDL à la fois lorsque celui-ci s'approche grossièrement de la cible. À l'opposé, lorsque celui-ci se trouve à proximité de la cible, il souhaite pouvoir décomposer la tâche afin de pouvoir se concentrer sur la manipulation de DDL spécifiques. Cette décomposition doit, entre autre, permettre d'isoler la profondeur. Nous proposons donc une technique permettant de combiner les DDL en phase balistique afin de permettre une approche rapide. Cette même technique permet de décomposer la tâche en phase de contrôle sans nécessiter aucun changement de mode ou d'activation de contraintes virtuelles explicites.

CrOS : Cursor On Surface

En modélisation géométrique, de nombreux opérateurs nécessitent de pouvoir travailler sur la surface même de l'objet (sculpture, coloration, opérations topologiques, *etc.*). Dans le cas où l'utilisation de périphériques à retour d'effort est impossible, ce type de manipulations est en général difficile. Elles sont généralement réalisées à l'aide d'une technique de manipulation directe. Le curseur manipulé est alors projeté sur la surface de l'objet afin de s'assurer que la manipulation se fasse à même la surface.

Pour cette technique, nous utilisons les recommandations 04, 05 et 06. La sculpture, la coloration, la définition d'une zone à supprimer sur des maillages sont des manipulations qui peuvent s'apparenter à du suivi de trajectoires sur des surfaces d'objets 3-D. Nous proposons donc une technique permettant de manipuler un curseur se déplaçant sur la surface 3-D. L'utilisateur déplace le curseur en utilisant la mise en oeuvre de la simulation d'une interface tactile que nous proposons.

Utilisation

L'ensemble de ces techniques sont incorporées dans deux applications développées pour notre environnement de RV :

1. **VRModelis** : une application de modélisation géométrique que nous avons développée. Cette application incorpore l'ensemble des outils de modélisation permettant d'effectuer les opérations de bases de modélisation (chargement, extrusion, déformation, sculpture, coloration, *etc.*) ;
2. **VR Terrain Editor** : une application de modélisation de terrain. Cette application utilise la structure de données développée dans le cadre du projet DNA (ANR

masses de données 07 - 10) [Peytavie 09]. Dans cette application, l'utilisateur peut modéliser et ajouter des détails naturels dans des scènes naturelles complexes.

Troisième partie

Amélioration du contrôle de l'utilisateur

DIOD : Decomposition and Integration of Degrees of Freedom

Sommaire

8.1	Mise en oeuvre	132
8.2	Évaluation de DIOD	135
8.2.1	Techniques comparées	136
8.2.2	Protocole expérimental	137
8.3	Résultats	140
8.3.1	Positionnement de précision	140
8.3.2	Positionnement répété	143
8.3.3	Évaluation subjective	146
8.4	Analyse	147
8.4.1	Stratégie	147
8.4.2	Performances	148
8.4.3	Évaluation subjective	150
8.5	Conclusions	150

Les résultats obtenus lors de l'étude portant sur la tâche de positionnement 3-D, ainsi que durant l'étude de l'influence de la décomposition, semblent indiquer plusieurs choses :

- L'utilisateur est capable de tirer parti de la manipulation simultanée de l'ensemble des trois DDL de la tâche. La coordination des DDL semble être avantageuse principalement en début de tâche, durant la phase balistique, lorsque l'utilisateur se rapproche grossièrement de la cible ;
- L'utilisateur semble décomposer partiellement la tâche de positionnement. Cette décomposition apparaît particulièrement en fin de tâche, durant la phase de contrôle,

lorsque l'utilisateur corrige la position de l'objet dans le but d'atteindre un seuil de précision maximal.

Dans ce chapitre, nous proposons une technique hybride pour la tâche de positionnement 3-D, *Decomposition and Integration Of Degrees of freedom* (notée DIOD). Cette technique a pour objectif de fournir un contrôle adapté, en matière de coordination des DDL, au cours des deux phases de la tâche. Elle permet de manipuler simultanément l'ensemble des dimensions de l'espace lorsque l'utilisateur s'approche grossièrement de la cible. Lorsque l'utilisateur corrige la position de l'objet manipulé, la technique passe à un mode de contrôle permettant de manipuler moins de DDL à la fois afin de permettre à l'utilisateur de se concentrer sur la manipulation de DDL particuliers.

Après avoir décrit la mise en oeuvre de cette technique, nous évaluons DIOD, en la comparant à deux autres techniques existantes : la manipulation directe et PRISM.

8.1 Mise en oeuvre

L'implémentation que nous proposons s'inspire de la technique PRISM. Nous le rappelons, le principe de cette dernière est le suivant :

- Lorsque l'utilisateur se déplace rapidement, il désire s'approcher grossièrement de la cible et n'a dans ce cas aucun besoin de précision ;
- Lorsque l'utilisateur se déplace lentement, l'utilisateur est à proximité de la cible et, dans ce cas, a besoin d'un contrôle précis.

En partant de ce principe, Frees *et al.* [Frees 05] proposent de mettre à l'échelle le déplacement de l'objet en fonction de la vitesse de déplacement du périphérique d'interaction. Les détails de la méthode utilisée ont été présentés dans le Chapitre 1.

Nous proposons d'adapter cette technique en utilisant les observations des études expérimentales que nous avons menées dans le contexte de la tâche de positionnement (cf. Chapitre 5). Notamment, nous avons pu remarquer que la décomposition ne semble pas adaptée durant la phase d'approche. Celle-ci ne fait qu'accroître les temps de manipulation. L'utilisateur doit donc pouvoir manipuler simultanément les DDL de la tâche afin de se rapprocher rapidement de la cible.

Toutefois, à proximité de la cible, l'utilisateur doit pouvoir corriger la position de l'objet manipulé. Dans ce cas, la coordination des DDL semble inappropriée. Sans contraintes particulières, la manipulation d'une dimension particulière est impossible. Néanmoins, durant la phase de contrôle, l'utilisateur doit pouvoir ajuster la position de l'objet selon des dimensions particulières (comme par exemple la profondeur). Il doit donc être en mesure de manipuler un DDL, sans affecter les autres DDL.

Pour obtenir le comportement voulu avec DIOD, nous proposons la fonction de transfert suivante. Nous considérons $\vec{D}_{\text{périphérique}}$, le vecteur de déplacement du périphérique d'interaction. Les trois composantes du vecteur de déplacement sont notées D_x , D_y et D_z . V , V_x , V_y et V_z sont les vitesses instantanées associées à ces déplacements. La fonction de transfert est définie de la manière suivante :

$$D_{\text{objet}}^{\vec{}} = \begin{cases} D_{\text{périphérique}}^{\vec{}} & , \text{ si } V \geq V_{\text{max}} \\ \begin{pmatrix} D'_x \\ D'_y \\ D'_z \end{pmatrix} & , \text{ sinon} \end{cases} \quad (8.1)$$

D'_x , D'_y et D'_z sont définis de la façon suivante :

$$D'_i = D_i \times f(V_i), \forall i \in \{x, y, z\}$$

Avec

$$f(V_i) = \begin{cases} 0 & , \text{ si } V_i < V_{\text{min}} \\ \min(1, \frac{V_i}{V_{\text{scale}}}) & , \text{ sinon} \end{cases} \quad (8.2)$$

Par rapport à PRISM, nous considérons le mouvement de translation comme la variation d'un ensemble de DDL. En fonction des besoins, ces DDL peuvent être combinés ou manipulés indépendamment les uns des autres. Plus particulièrement, si le déplacement effectué a une vitesse élevée, la manipulation se fait à une échelle 1 : 1. L'utilisateur peut alors facilement combiner plusieurs DDL simultanément. Toutefois, lorsque la vitesse des mouvements diminue, chacune des dimensions est filtrée indépendamment des autres. Comme illustré en Figure 8.1, le seuillage basé sur la vitesse minimale facilite la manipulation d'un DDL à la fois. Cette décomposition facilite la manipulation de dimensions spécifiques en fin de tâche. Le contrôle qu'offre notre technique semble donc adapté, à la fois en phase balistique ainsi qu'en phase de contrôle.

Le filtrage par DDL permet une transition douce entre la première et la deuxième phase. Comme illustré en Figure 8.2, le mouvement dans chacune des dimensions est progressivement réduit jusqu'à ce qu'il passe sous le seuil de vitesse minimal. Une fois le seuil atteint, les mouvements selon cette dimension sont réduits à zéro par le filtrage.

Cette technique, dans le principe, ressemble à l'utilisation des *GearBox widget* proposés par Osawa *et al.* [Osawa 07]. Cette technique permet à l'utilisateur de déplacer un objet à l'aide de la manipulation directe. L'utilisateur peut également déplacer l'objet selon un DDL particulier en utilisant un composant graphique appelé *GearBox*. Ce composant graphique, illustré en Figure 8.3, permet de modifier une valeur avec différents niveaux de précision. Chaque niveau de précision est représenté par une roue crantée.

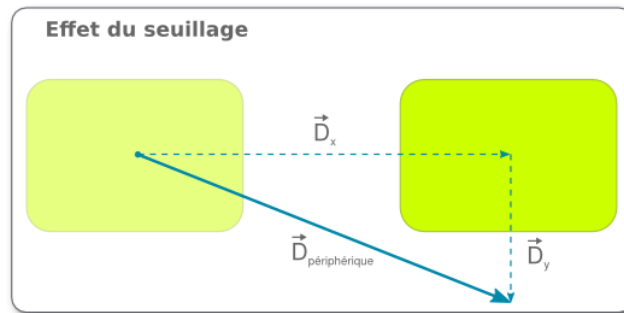


FIGURE 8.1 – La vitesse instantanée selon la dimension y n'est pas assez élevée, elle est donc réduite à 0 par le filtrage. Le filtrage transforme un mouvement **presque unidimensionnel** en un mouvement **unidimensionnel**.

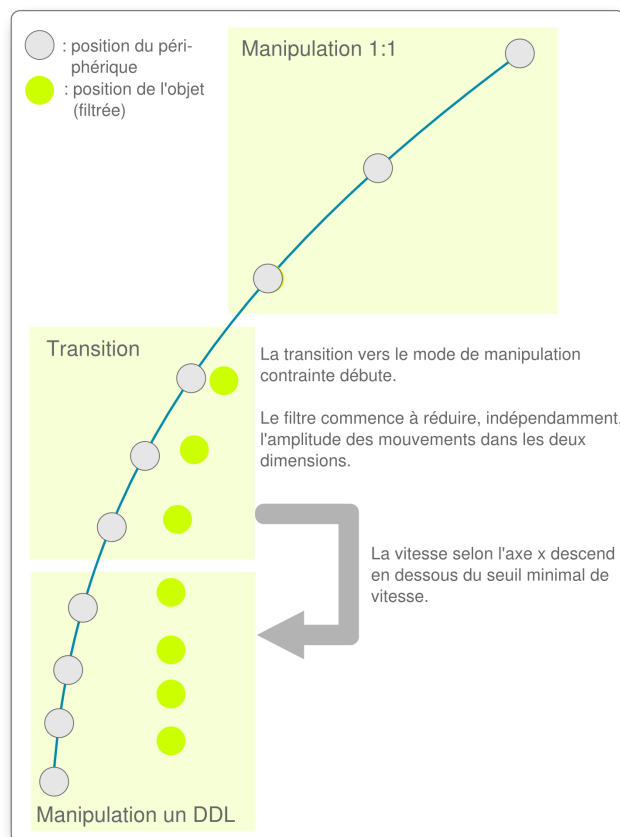


FIGURE 8.2 – La transition entre la phase de manipulation à l'échelle 1 :1 et la phase de manipulation contrainte se fait en douceur grâce au filtrage par dimension. Le mouvement est progressivement contraint à une manipulation à un DDL.

L'utilisateur peut tourner chacun des engrenages pour modifier la valeur qui lui est associée. En manipulant un engrenage de grande taille l'utilisateur effectue d'importantes modifications de la valeur. À l'inverse, la manipulation d'un engrenage de petite taille permet d'effectuer de petites modifications.

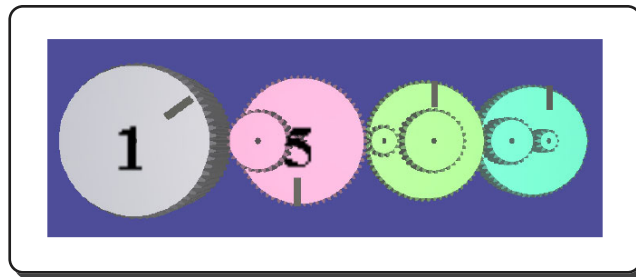


FIGURE 8.3 – La représentation graphique du composant GearBox. Illustration extraite de [Osawa 07].

L'utilisation d'un composant graphique lors du positionnement 3-D introduit de multiples sélections - manipulations qui peuvent influencer la réalisation de la tâche :

- En introduisant des sous-tâches intermédiaires, l'utilisation d'un composant graphique peut accroître significativement les temps de manipulation ;
- L'utilisateur doit constamment alterner entre des phases de manipulation de l'objet et des phases de sélection - manipulation du composant graphique.

La technique que nous proposons permet de passer d'un mode de contrôle grossier, à un mode de contrôle plus fin, de manière automatique, en utilisant la vitesse instantanée comme un indicateur des intentions de l'utilisateur. Nous n'utilisons pas ici de contraintes explicites (*i.e.* spécifiées explicitement par l'utilisateur au travers de l'activation d'un mode de manipulation spécifique) ou de composants graphiques intermédiaires.

Nous avons mis en place une étude expérimentale afin de mesurer les performances de cette technique par rapport aux techniques existantes.

8.2 Évaluation de DIOD

Dans une étude préliminaire, nous avons évalué informellement la technique DIOD. Les résultats que nous avons obtenus indiquent que la technique DIOD offre un contrôle approprié durant les différentes phases de réalisation de la tâche de positionnement. Nous proposons d'étudier, dans une nouvelle expérimentation, cette question ainsi que l'impact d'une stratégie basée sur la décomposition sur les performances des utilisateurs.

8.2.1 Techniques comparées

Dans le cadre de cette étude, nous comparons trois techniques d'interaction. Aucune des trois techniques n'est une technique de manipulation avec colocalisation.

Manipulation directe

Cette technique d'interaction a déjà été utilisée dans le cadre de l'étude décrite dans le Chapitre 5. Durant ce protocole, nous utilisons la même technique. Pour saisir l'objet, l'utilisateur active les capteurs de pression du pouce et de l'index en pinçant les deux doigts. Il translate alors l'objet en déplaçant la main dans l'environnement tout en maintenant cette posture. Aucune fonction de transfert n'est utilisée dans le cadre de cette technique.

PRISM

La technique PRISM est la technique décrite par Frees *et al.* [Frees 05]. Par rapport à la technique proposée, nous n'utilisons pas de méthode de **recouvrement d'erreur**. Cette méthode permet de repositionner la main à la position même de l'objet manipulé sans avoir à relâcher l'objet. Elle n'est pas nécessaire ici car nous nous trouvons dans une situation de manipulation non colocalisée.

Les seuils utilisés dans notre implémentation sont les suivants :

- $V_{min} = 0,05m, s^{-1}$;
- $V_{scale} = 1,15m, s^{-1}$.

DIOD

L'implémentation de la technique DIOD est identique à celle proposée en Section 8.1. Nous précisons simplement les différents seuils utilisés dans le cadre de ce protocole. Les seuils, comme pour la technique PRISM, ont été déterminés de manière empirique après une phase d'essai dans l'environnement.

- $V_{min} = 0,002m, s^{-1}$;
- $V_{scale} = 0,95m, s^{-1}$;
- $V_{max} = 1,15m, s^{-1}$.

8.2.2 Protocole expérimental

Nous décrivons à présent en détails la méthodologie utilisée dans le cadre de cette étude expérimentale.

Procédure

Douze sujets volontaires ont participé à cette étude expérimentale. Les sujets ont des niveaux d'expériences variables en ce qui concerne la manipulation en environnements de RV. De même que pour les protocoles précédents, nous utilisons un plan expérimental intra-sujet. Les sujets participant à cette étude effectuent l'ensemble des tâches avec les trois techniques.

Le protocole débute par une phase d'apprentissage durant laquelle aucune mesure n'est prise. Au cours de cette première phase, les utilisateurs testent les différentes techniques d'interaction. Cette phase nous permet également de présenter aux utilisateurs les deux tâches de positionnement utilisées au cours de cette étude. La phase d'apprentissage consiste à réaliser quatre essais pour la première tâche ainsi que trois essais pour la deuxième tâche en utilisant chacune des trois techniques.

À l'issue de cette phase, les utilisateurs débutent la phase de test. Pour chacune des techniques, l'utilisateur effectue un bloc de vingt quatre positionnements pour la première tâche et un bloc de douze positionnements pour la deuxième tâche. Au cours du protocole, les utilisateurs sont invités à faire des commentaires à haute voix. L'ordre de présentation des techniques est contre balancé à l'aide d'une répartition basée sur un carré latin afin d'éviter tout effet d'ordre sur les résultats.

Tâches

Durant ce protocole, nous utilisons deux tâches de positionnements aux objectifs différents. La première tâche est une tâche de **positionnement répété** (notée PR). Cette tâche est utilisée par Frees *et al.* [Frees 05] dans le protocole de validation de la technique d'interaction PRISM. Cette tâche consiste à positionner un objet (ici une sphère), à une position cible (représentée par le centre d'un cube), le plus grand nombre de fois possible au cours d'un laps de temps fixe. Dans le protocole expérimental de Frees *et al.*, la durée est fixée à trois minutes. Afin d'éviter de trop longues sessions de tests, dans le cadre de notre protocole, nous avons fixé la durée à une minute. Pour cette tâche, un objet est considéré comme positionné lorsque la distance entre le centre de la sphère et le centre du cube se trouve sous un seuil de distance. Nous avons utilisé quatre seuils de distances correspondant à quatre niveaux de difficulté de la tâche (Facile, Moyen, Difficile, Très Difficile). Pour chaque niveau de difficulté, les seuils de précision sont :

- Facile : 1 cm ;
- Moyen : 3,5 mm ;
- Difficile : 2 mm ;
- Très difficile : 1,2 mm.

Ces seuils de précision sont identiques à ceux utilisés par Frees *et al.*. Afin d'éviter des positionnements involontaires, nous considérons un objet comme positionné, lorsque pendant un laps de temps de 0,8 s, la distance entre le centre de la sphère et le centre du cube, est inférieure au seuil de précision. Une fois le positionnement effectué, l'objet est automatiquement replacé à sa position initiale dans l'environnement 3-D. La position initiale de la sphère ne change pas au cours de l'expérimentation. Une illustration de la tâche est donnée en Figure 8.4.

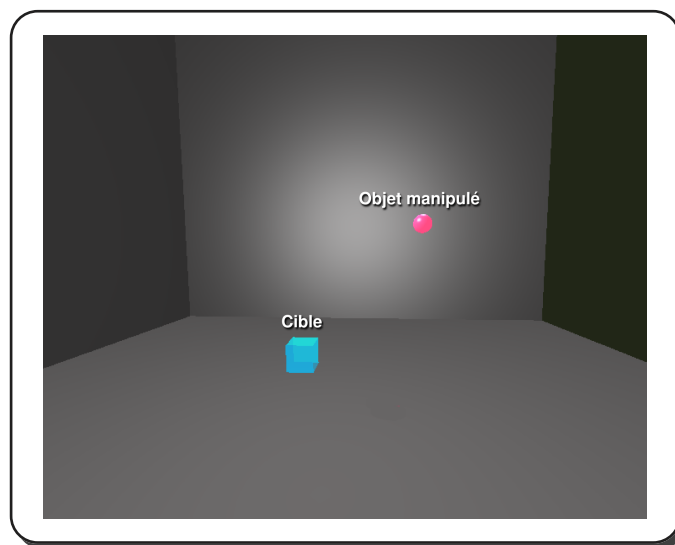


FIGURE 8.4 – Illustration du retour visuel associé à la tâche PR. Le sujet doit positionner la sphère dans le cube le plus de fois possible au cours du temps autorisé.

La deuxième tâche de positionnement est une tâche de **positionnement de précision** (notée PP). Cette tâche est similaire à la tâche utilisée dans l'étude décrite au Chapitre 5. Elle consiste à positionner, le plus précisément possible, l'objet manipulé à une position cible. Pour cette tâche également, la position de la cible ne varie pas au cours des essais. Afin de permettre aux utilisateurs d'atteindre une précision suffisante, nous avons défini un retour visuel permettant de connaître exactement le niveau de précision de la position courante. Ce retour visuel permet d'évaluer la distance à la cible mais également d'évaluer la distance selon chacun des DDL. Tout ceci est illustré en Figure 8.5.

Le sujet valide la tâche lorsqu'il pense avoir atteint un niveau de précision suffisant. Afin de valider la tâche courante, le sujet met la main dominante dans un composant

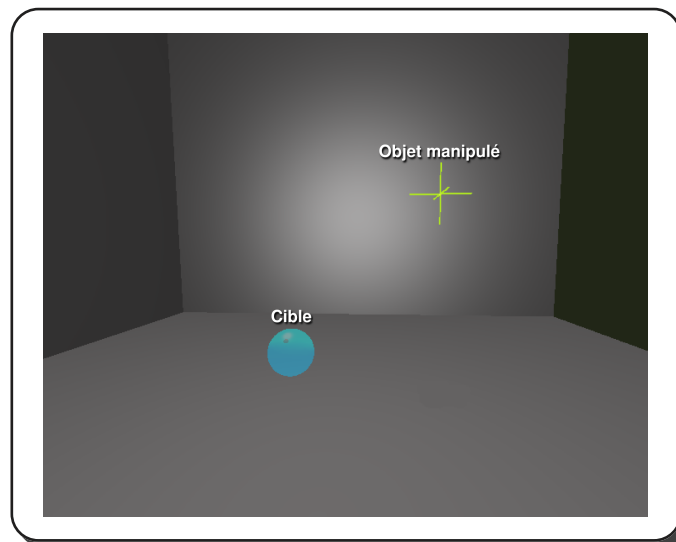


FIGURE 8.5 – Visuel de la tâche de positionnement de précision. L'intersection entre la croix et la sphère permet d'évaluer précisément la distance selon chacune des dimensions.

graphique de validation pendant une seconde. Ce composant est représenté par un cube rouge dans le coin inférieur droit de l'environnement. Une fois la validation effectuée, les sujets rentrent dans une phase de transition où aucune mesure n'est prise. C'est durant cette phase que les utilisateurs sont invités à faire des commentaires et / ou à se reposer. Pour commencer une nouvelle tâche, le sujet effectue la même séquence de validation.

Mesures

Au cours de cette étude, nous utilisons l'ensemble des mesures suivantes afin d'évaluer à la fois les performances mais également les comportements des utilisateurs :

- Nombre de positionnements (noté NPR) : le nombre de positionnements effectués lors de la tâche PR ;
- Erreur euclidienne (notée E , en mm) : distance euclidienne entre l'objet manipulé et la cible à la fin de la tâche ;
- Erreur selon les DDL X, Y et Z (notée respectivement E_x , E_y et E_z , en mm) : pour chacune des dimensions de l'espace 3-D, la distance entre la position finale de l'objet manipulé et la cible ;
- Temps de réalisation (noté TRE, en seconde) : temps écoulé entre le premier et le dernier mouvement de l'objet manipulé ;
- Temps de manipulation (noté TM, en seconde) : temps effectif durant lequel l'ob-

- jet manipulé a été déplacé dans la scène. La différence entre TRE et TM permet d'estimer le temps durant lequel l'utilisateur évalue la tâche ;
- Coordination (notée NDC) : calcul de la mesure NDC à partir de la vitesse de déplacement de l'objet manipulé ;
 - Pourcentages de temps passé à manipuler un, deux et trois DDL (notés P_{1DDL} , P_{2DDL} et P_{3DDL}) ;
 - Pourcentages de temps passé à manipuler un, deux et trois DDL en début et fin de tâche. La distinction entre le début et la fin de la tâche se fait en utilisant un seuil de précision. Avant que l'objet n'ait atteint ce seuil de précision, nous considérons que l'utilisateur est en début de tâche. Une fois le seuil atteint, l'utilisateur est en fin de tâche. Le seuil de précision est défini comme $5 \times E$. Le seuil est spécifique à chaque réalisation ;
 - Évaluation subjective : nous demandons aux sujets de remplir le questionnaire donné en Annexe C. Cette évaluation consiste en un ensemble de questions, évaluant les techniques selon différents critères grâce à une échelle de Likert à 7 niveaux. L'objectif est de recueillir les impressions subjectives des utilisateurs sur les trois techniques d'interaction.

Les mesures de précisions (*i.e.* E , E_x , E_y et E_z) sont utilisées uniquement dans le cas de la tâche de positionnement de précision.

8.3 Résultats

Nous présentons dans un premier temps les résultats obtenus durant la tâche PP avant de présenter les résultats concernant la tâche PR.

De la même manière que pour les études précédentes, nous utilisons une ANOVA à mesures répétées. La technique d'interaction est définie comme la variable dépendante. Les facteurs dépendants sont les mesures prises lors du protocole expérimental. Nous utilisons une comparaison paire à paire avec une correction de Benjamini et Hochberg.

8.3.1 Positionnement de précision

Précision

Les résultats concernant la précision sont donnés dans le Tableau 8.1. Les résultats semblent indiquer que la technique d'interaction a une influence significative sur E , E_x et E_z . Pour l'erreur euclidienne, les différences entre MD et les autres techniques sont significatives ($p - \text{value} < 0,001$). La différence entre DIOD et PRISM indique une tendance ($p - \text{value} = 0,084$). Il semblerait donc que les techniques PRISM et DIOD

TI	E	E_x	E_y	E_z
MD	3,71 mm	0,82 mm	1,24 mm	2,63 mm
PRISM	2,88 mm	0,79 mm	1,33 mm	1,80 mm
DIOD	2,54 mm	0,52 mm	1,21 mm	1,72 mm
F	$F(2,22) = 12,16$	$F(2,22) = 15,98$	$F(2,22) = 0,72$	$F(2,22) = 11,21$
$\rho - value$	$< 0,001$	$< 0,001$	0,50	$< 0,001$

TABLE 8.1 – L'erreur moyenne pour chacune des techniques comparées.

soient plus précises que MD.

Nous pouvons également noter que les différences pour E_x sont significatives entre DIOD et les autres techniques ($\rho - value < 0,001$). La différence entre PRISM et MD n'est pas significative. En ce qui concerne E_z , les différences entre MD et les autres techniques sont statistiquement significatives ($\rho - value < 0,001$). La différence entre DIOD et PRISM n'est pas significative ($\rho - value = 0,72$).

Temps de réalisation

TI	TRE	TM
MD	14,31 s	7,48 s
PRISM	10,60 s	6,19 s
DIOD	13,55 s	7,33 s
F	$F(2,22) = 5,48$	$F(2,22) = 3,68$
$\rho - value$	0,012	0,042

TABLE 8.2 – Pour chacune des techniques utilisées les temps moyens d'exécution et de manipulation.

Dans le Tableau 8.2, nous pouvons remarquer que la technique d'interaction a une influence significative sur les temps de réalisation et de manipulation. Pour TRE, les différences entre PRISM et les autres techniques sont significatives ($\rho - value < 0,001$). La différence entre MD et DIOD n'est pas significative.

En ce qui concerne TM, les résultats sont similaires. Les différences entre PRISM et les autres techniques sont statistiquement significatives (PRISM vs. DIOD : $\rho - value = 0,001$; PRISM vs. MD : $\rho - value < 0,001$). Il semblerait donc que la technique PRISM soit plus rapide que les autres techniques d'interaction.

Coordination

TI	NDC
<i>MD</i>	2,26
<i>PRISM</i>	2,26
<i>DIOD</i>	2,06
<i>F</i>	$F(2, 22) = 170,45$
$\rho - value$	$< 0,001$

TABLE 8.3 – Coordination moyenne pour chaque technique.

La technique d'interaction semble avoir une influence sur la coordination moyenne au cours de la tâche. Les données du Tableau 8.3 indiquent que les utilisateurs décomposent plus la tâche avec DIOD qu'avec les autres techniques. Les différences entre DIOD et les autres techniques sont significatives ($\rho - value < 0,001$). La différence entre PRISM et MD n'est pas significative ($\rho - value = 0,66$).

TI	P_{1DDL}	P_{2DDL}	P_{3DDL}
<i>MD</i>	3,2 %	40,0 %	56,8 %
<i>PRISM</i>	5,1 %	40,9 %	54,1 %
<i>DIOD</i>	21,9 %	46,2 %	31,9 %
<i>F</i>	$F(2, 22) = 468,73$	$F(2, 22) = 58,86$	$F(2, 22) = 36,30$
$\rho - value$	$< 0,001$	$< 0,001$	$< 0,001$

TABLE 8.4 – Moyennes des proportions de temps de manipulation de un, deux et trois DDL par rapport au temps de manipulation total.

D'après les données indiquées dans le Tableau 8.4, la technique d'interaction a une influence significative sur la proportion de temps passé à manipuler un, deux et trois DDL. Pour P_{1DDL} et P_{3DDL} , seules les différences entre DIOD et les autres techniques sont significatives ($\rho - value < 0,001$). Les différences entre MD et PRISM ne sont pas significatives ($\rho - value > 0,1$). En ce qui concerne P_{2DDL} , les différences entre DIOD et les autres techniques sont significatives (DIOD vs. MD : $\rho - value < 0,001$; DIOD vs. PRISM : $\rho - value < 0,001$). La différence entre PRISM et MD indique une tendance ($\rho - value = 0,061$).

Décomposition en fin de tâche

Les données du Tableau 8.5 confirment les résultats que nous avons obtenus lors des études précédentes. Les utilisateurs semblent vouloir décomposer la tâche lorsque

TI	1 DDL		2 DDL		3 DDL	
	Avant	Après	Avant	Après	Avant	Après
<i>MD</i>	3,7 %	3,3 %	33,3 %	42,2 %	61,9 %	54,5 %
<i>PRISM</i>	4,3 %	6,1 %	34,4 %	45,8 %	62,4 %	48,0 %
<i>DIOD</i>	15,4 %	35,6 %	34,9 %	51,7 %	49,6 %	12,6 %
<i>F</i>	$F(2,22) = 46,06$	$F(2,22) = 79,84$	$F(2,22) = 0,28$	$F(2,22) = 4,22$	$F(2,22) = 15,2$	$F(2,22) = 67,32$
<i>p – value</i>	< 0,001	< 0,001	0,75	0,03	< 0,001	< 0,001

TABLE 8.5 – Moyennes des proportions de temps de manipulation de un, deux et trois DDL en début et en fin de tâche par rapport au temps passé en début et fin de tâche.

l'objet manipulé se trouve à proximité de la cible. Nous comparons les pourcentages de temps de manipulation, avant et après le seuil de précision, à l'aide d'un test de Student pour chacune des techniques :

- **1 DDL** : MD : $p - value = 0,85$; PRISM : $p - value = 0,13$; DIOD : $p - value < 0,001$;
- **2 DDL** : MD : $p - value = 0,001$; PRISM : $p - value < 0,001$; DIOD : $p - value < 0,001$;
- **3 DDL** : MD : $p - value = 0,005$; PRISM : $p - value < 0,001$; DIOD : $p - value < 0,001$.

Les résultats indiquent qu'avec la technique MD et PRISM, la proportion de temps passé à manipuler un DDL ne croît pas après le seuil de précision. Néanmoins, le pourcentage de temps passé à manipuler deux DDL augmente significativement alors que le temps passé à manipuler trois DDL réduit de manière significative. Cela semble indiquer que l'utilisateur tente de décomposer le positionnement en fin de tâche même avec des techniques qui ne favorisent pas cette stratégie. Avec la technique DIOD, la décomposition en fin de tâche est plus importante. La proportion de temps passé à manipuler un et deux DDL augmente significativement alors que la proportion passé à manipuler trois DDL diminue significativement.

8.3.2 Positionnement répété

Le Tableau 8.6 indique le nombre de réalisations pour chacune des techniques. La technique a une influence significative sur le nombre de positionnements effectués pour les niveaux de difficulté Moyen, Difficile et Très difficile. Pour chacun de ces niveaux de difficulté, les $p - value$ pour chaque comparaison possible sont données dans le Tableau 8.8. Dans le Tableau 8.7, nous retrouvons le pourcentage de différence entre DIOD et les autres techniques. Nous pouvons observer que plus le niveau de difficulté augmente,

TI	Facile	Moyen	Difficile	Très difficile
<i>MD</i>	19,04	12	7,54	3,08
<i>PRISM</i>	18,21	12,96	9,12	5,25
<i>DIOD</i>	19,08	14,21	10,79	7,46
<i>F</i>	$F(2,22) = 1,42$	$F(2,22) = 7,00$	$F(2,22) = 10,70$	$F(2,22) = 25,11$
<i>p – value</i>	0,26	0,004	< 0,001	< 0,001

TABLE 8.6 – Nombres moyens de réalisations pour chacune des techniques et chacun des niveaux de difficulté.

	DIOD vs PRISM	DIOD vs DM
Easy	6,8 %	8,1 %
Medium	9,6 %	18,3 %
Hard	10,1 %	43,1 %
Very Hard	47,2 %	103 %

TABLE 8.7 – Différences de performance entre DIOD et les autres techniques.

plus l'écart entre DIOD et les autres techniques est important.

	Moyen	Difficile	Très difficile
<i>MD vs. PRISM</i>	0,24	0,053	0,005
<i>DIOD vs. MD</i>	0,024	< 0,001	< 0,001
<i>DIOD vs. PRISM</i>	0,191	0,053	0,005

TABLE 8.8 – Pour chaque niveau de difficulté, les *p – value* pour l'ensemble des comparaisons possibles.

Coordination

Les données du Tableau 8.9 indiquent que la technique a une influence significative sur la coordination moyenne. Les résultats sont similaires à ceux obtenus lors de la tâche PP. La décomposition est plus importante avec DIOD qu'avec les autres techniques (*p – value* < 0,001) alors que la différence entre PRISM et MD n'est pas significative (*p – value* = 0,86).

Si nous regardons les résultats de l'analyse portant sur la décomposition, comme pour la tâche PP, elle semble plus importante pour DIOD que pour les autres techniques. Pour les trois pourcentages, les différences entre MD et PRISM ne sont pas significatives

TI	NDC
<i>MD</i>	2,30
<i>PRISM</i>	2,31
<i>DIOD</i>	2,16
<i>F</i>	$F(2, 22) = 13,17$
$\rho - value$	$< 0,001$

TABLE 8.9 – Coordination moyenne pour chaque technique.

TI	P_{1DDL}	P_{2DDL}	P_{3DDL}
<i>MD</i>	3,85 %	32,16 %	63,98 %
<i>PRISM</i>	4,73 %	29,35 %	65,92 %
<i>DIOD</i>	15,32 %	21,63 %	63,04 %
<i>F</i>	$F(2, 22) = 73,91$	$F(2, 22) = 25,33$	$F(2, 22) = 2,45$
$\rho - value$	$< 0,001$	$< 0,001$	0,11

TABLE 8.10 – Moyennes des proportions de temps de manipulation de un, deux et trois DDL par rapport au temps de manipulation total.

($\rho - value > 0,05$). Toutefois, les différences entre DIOD et les autres techniques sont significatives pour le pourcentage de temps passé à manipuler un et deux DDL ($\rho - value < 0,001$).

Décomposition en fin de tâche

Nous avons également comparé le temps passé à manipuler un, deux et trois DDL avant et après le seuil de précision.

TI	1 DDL		2 DDL		3 DDL	
	Avant	Après	Avant	Après	Avant	Après
<i>MD</i>	2,9 %	7,5 %	28,1 %	42,8 %	68,9 %	49,65 %
<i>PRISM</i>	4,1 %	7,9 %	30,6 %	41,5 %	65,4 %	50,6 %
<i>DIOD</i>	8,7 %	23,2 %	28,1 %	31,2 %	63,2 %	45,5 %
<i>F</i>	$F(2, 13) = 16,39$	$F(2, 22) = 256,15$	$F(2, 13) = 0,21$	$F(2, 22) = 46,08$	$F(2, 13) = 1,25$	$F(2, 22) = 5,95$
$\rho - value$	$< 0,001$	$< 0,001$	0,81	$< 0,001$	0,32	0,008

TABLE 8.11 – Moyennes des proportions de temps de manipulation de un, deux et trois DDL en début et en fin de tâche par rapport au temps passé en début et fin de tâche.

De la même manière que pour la tâche PP, nous effectuons un test de Student afin de voir si la différence de pourcentage de manipulation de un, deux et trois DDL est significative. Les p – *value* des comparaisons sont les suivantes :

- **1 DDL** : MD : p – *value* < 0,001 ; PRISM : p – *value* < 0,001 ; DIOD : p – *value* < 0,001 ;
- **2 DDL** : MD : p – *value* < 0,001 ; PRISM : p – *value* = 0,006 ; DIOD : p – *value* = 0,43 ;
- **3 DDL** : MD : p – *value* = 0,001 ; PRISM : p – *value* < 0,001 ; DIOD : p – *value* = 0,001.

Au cours de la tâche PR, il semblerait que les utilisateurs décomposent la tâche à proximité de la cible. Ce résultat est également similaire à celui obtenu dans le tâche PP.

8.3.3 Évaluation subjective

	MD	PRISM	DIOD	F	p – <i>value</i>
Q1	6,4	6,4	6,4	$F(2, 20) = 1,0$	0,38
Q2	2,0	4,8	6,7	$F(2, 20) = 91,65$	< 0,001
Q3	6,4	6,1	6,2	$F(2, 20) = 1,1$	0,35
Q4	4,0	5,4	6,3	$F(2, 20) = 8,33$	0,002
Q5	6,6	5,9	6,4	$F(2, 20) = 2,45$	0,11
Q6	3,7	5,5	6,5	$F(2, 20) = 15,44$	< 0,001
Q7	3,9	5,5	6,5	$F(2, 20) = 12,56$	< 0,001
Q8	2,4	5,4	6,8	$F(2, 20) = 41,67$	< 0,001
Q9	1,4	4,5	6,7	$F(2, 20) = 91,33$	< 0,001
Q10	6,4	6,4	6,4	$F(2, 20) = 0,92$	0,41
Q11	1,9	4,6	6,6	$F(2, 20) = 57,74$	< 0,001
Q12	6,3	6,3	6,1	$F(2, 20) = 0,12$	0,88
Q13	4,3	5,5	6,4	$F(2, 20) = 13,5$	< 0,001
Q14	2,8	5,0	6,8	$F(2, 20) = 37,89$	< 0,001

TABLE 8.12 – Moyennes des notes obtenues lors de l'évaluation subjective. En gras, la meilleure note obtenue lorsque la technique a une influence significative.

Le Tableau 8.12 indique que la technique d'interaction a une influence sur les réponses données aux questions 2, 4, 6, 7, 8, 9, 11, 13 et 14. Pour ces questions, les p – *value* pour chacune des comparaisons sont données dans le Tableau 8.13. Les utilisateurs ont pensé que la technique DIOD est plus adaptée à la tâche de positionnement 3-D que les autres techniques. L'absence de différence pour Q3 semble indiquer que les utilisateurs ont pensé que DIOD offre un contrôle similaire aux techniques MD et

PRISM lors de la phase balistique. Toutefois, les utilisateurs semblent indiquer que DIOD offre un contrôle plus approprié à la tâche en phase de contrôle. De plus, la technique DIOD a été évaluée comme la technique préférée des utilisateurs.

	MD vs. PRISM	MD vs. DIOD	DIOD vs. PRISM
<i>Q2</i>	< 0,001	< 0,001	< 0,001
<i>Q4</i>	0,052	0,003	0,15
<i>Q6</i>	< 0,001	< 0,001	0,045
<i>Q7</i>	0,01	< 0,001	0,086
<i>Q8</i>	< 0,001	< 0,001	0,007
<i>Q9</i>	< 0,001	< 0,001	< 0,001
<i>Q11</i>	< 0,001	< 0,001	< 0,001
<i>Q13</i>	0,004	< 0,001	0,028
<i>Q14</i>	< 0,001	< 0,001	0,001

TABLE 8.13 – p – value pour l'ensemble des comparaisons.

8.4 Analyse

Nous allons à présent analyser les résultats obtenus en matière de stratégies de réalisation, de précisions, de temps de réalisation et d'évaluations subjectives.

8.4.1 Stratégie

Les résultats obtenus concernant la manipulation des DDL mettent en avant d'importantes différences entre DIOD et PRISM ou MD. Les pourcentages de temps passés à manipuler un, deux et trois DDL simultanément montrent que DIOD facilite grandement la décomposition de la tâche. Que se soit pour PP ou PR, le pourcentage de temps passé à manipuler un et deux DDL est significativement plus important avec DIOD qu'avec les autres techniques. De plus, les utilisateurs passent moins de temps à manipuler trois DDL simultanément avec DIOD.

En comparant la manipulation, avant et après le seuil de précision, nous constatons que les utilisateurs ont tendance, avec DIOD, à décomposer la tâche durant la phase de contrôle. Alors que le pourcentage de temps passé à manipuler trois DDL réduit de 74,6 % durant PP et de 28 % durant PR, le pourcentage de temps passé à manipuler un et deux DDL augmentent respectivement de 131 % et 48 % durant la tâche PP et de 166,1 % et 11 % durant la tâche PR. Ce résultat suggère que les sujets sont capables d'intégrer partiellement la manipulation des DDL durant la phase balistique, alors qu'ils

exploitent les caractéristiques de la technique afin de décomposer la tâche durant la phase de contrôle. Ce résultat confirme que DIOD a le comportement escompté.

Nous pouvons observer un écart entre les résultats obtenus durant la tâche PR et les résultats obtenus durant la tâche PP. Ceci peut s'expliquer par le fait que, durant la tâche PR, la moitié des positionnements sont de niveaux Facile et Moyen. Pour ces deux niveaux de difficulté, la durée de la phase de contrôle est très limitée (voir même inexistante). Les utilisateurs n'ont pas besoin de précision et n'ont donc pas besoin de décomposer la tâche. Au contraire, lors de la tâche PP, les utilisateurs passent une part importante du temps à proximité de la cible afin d'effectuer des corrections successives, selon chacune des dimensions, sur la position de l'objet manipulé.

Nous pouvons observer la mise en place d'une stratégie de décomposition, mais dans de moindres proportions, pour les techniques PRISM et MD. Pour la technique MD et PRISM, le temps de manipulation de un DDL augmente, en moyenne sur les deux tâches, de 73,9 % pour MD et de 61,1 % pour PRISM. Le temps de manipulation de deux DDL augmente en moyenne de 39,5 % pour MD et de 34,5 % pour PRISM. L'accroissement de la manipulation de deux DDL est significatif dans les deux tâches alors que l'accroissement de la manipulation de un DDL est significatif que durant la tâche PR. Ces résultats semblent indiquer que, bien que les techniques MD et PRISM ne facilitent pas la décomposition de la tâche, les utilisateurs tentent néanmoins de réduire le nombre de DDL manipulés simultanément en fin de tâche. Ces résultats sont en adéquation avec ceux obtenus lors des précédentes études que nous avons menées. Il semblerait que, dans ce contexte, PRISM puisse aider l'utilisateur à mettre en place cette stratégie. En effet, le filtrage introduit par PRISM, permet par une manipulation appropriée, de décomposer partiellement la tâche. Toutefois, il semblerait qu'il soit difficile de manipuler un DDL particulier (à cause des tremblements ou de contraintes physiologiques). Ceci peut expliquer l'augmentation significative de la proportion de temps passé à manipuler deux DDL alors que l'augmentation du temps passé à manipuler un DDL est plus réduite.

Ces résultats nous conduisent à penser que la technique DIOD permet aux utilisateurs de mettre en place des stratégies de réalisations qu'ils essayent de mettre en place avec des techniques qui ne sont pas forcément adaptées. Cette stratégie consiste à intégrer la manipulation de l'ensemble des DDL durant la phase balistique. Durant la phase de contrôle, les utilisateurs décomposent la tâche en manipulant un DDL à la fois. Nous allons à présent nous intéresser à l'influence de ces stratégies sur les performances.

8.4.2 Performances

DIOD semble être la technique la plus efficace durant la tâche PR. Les résultats montrent des gains respectifs de 47,2 % et de 103 % par rapport à MD pour les ni-

veaux de difficulté Difficile et Très difficile. De plus, comme le montre les données du Tableau 8.7, les bénéfices par rapport à MD et PRISM semblent s'accroître avec la difficulté de la tâche. Nous pensons que ce résultat peut s'expliquer par la facilité à mettre en place une stratégie de décomposition avec DIOD. Comme nous l'avons indiqué dans la section précédente, les utilisateurs tentent d'aligner successivement l'objet manipulé selon les différents axes. Nous avons pu observer qu'après quelques essais, les utilisateurs alignent l'objet manipulé dans le plan X-Y avant d'aligner selon l'axe Z.

Cette stratégie peut être favorisée par le retour visuel de la tâche. La cible étant un cube, il est facile pour les utilisateurs d'évaluer la distance selon chaque direction. Si nous n'observons que de faibles différences pour les niveaux de difficulté Facile et Moyen, il est possible que ce soit dû au fait que la précision requise ne soit pas suffisamment importante pour que l'utilisateur puisse tirer parti des caractéristiques de DIOD. Cela semble confirmer que la décomposition n'est bénéfique que lorsqu'une importante précision est nécessaire.

Pour la tâche PP, les résultats sont différents. DIOD semble être plus précise avec un gain de 31,2 % par rapport à MD et de 11,7 % par rapport à PRISM. Toutefois, ce dernier résultat n'est pas significatif. L'analyse statistique révèle simplement une tendance. De plus, les temps de réalisation sont significativement plus importants pour DIOD et pour MD que pour PRISM (avec des augmentations respectives de 27,8 % et de 35 %). Ce résultat suggère que les stratégies ayant aidées les utilisateurs durant PR n'ont pas pu être exploitées ou n'ont pas été aussi efficaces durant PP.

Les différences importantes en matière de précision entre DIOD et MD, et les résultats obtenus durant PR, suggèrent différentes interprétations quant à l'augmentation des temps de réalisation et des temps de manipulation pour ces deux techniques. Pour MD, la technique étant sensible aux mouvements parasites, les utilisateurs ont dû repositionner à de nombreuses reprises l'objet à cause de mouvements involontaires en fin de tâche. Cela explique les temps de réalisation et manipulation plus élevés ainsi que la précision réduite.

Puisque DIOD et PRISM utilisent un filtrage basé sur la vitesse de déplacement, elles sont moins sujettes aux mouvements parasites. Contrairement à MD et PRISM, nous pensons que les utilisateurs ont senti qu'ils pouvaient atteindre de meilleures précisions avec DIOD. Les commentaires des utilisateurs permettent de justifier cette hypothèse. Certains utilisateurs nous ont confié que « *La technique DIOD fait exactement ce que je veux* », ou encore, « *Avec DIOD, j'ai l'impression que l'objet va directement dans la cible* ». Nous pensons que, par rapport à ce sentiment, les utilisateurs ont passé plus de temps à essayer de positionner précisément l'objet. Néanmoins, ce temps supplémentaire n'a pas permis d'introduire de gains suffisamment important en matière de précision. Nous pensons que cette limitation n'est pas liée à la technique, mais à des limites de perception dans ce type d'environnement [Hinckley 97].

La différence entre DIOD et PRISM en matière de précision selon la dimension x peut être un élément confirmant cette hypothèse. En effet, cette dimension semble être la dimension la moins difficile à appréhender au cours de l'interaction [Fiorentino 03]. Il est alors possible que ce soit la dimension dont la perception soit la moins problématique. La différence entre DIOD et PRISM indiquerait alors que, dans le cas où la perception ne limite pas le seuil de précision maximal atteignable, DIOD soit la technique offrant le plus de précision. L'absence de différences entre DIOD et PRISM selon les deux autres dimensions, pourrait être due à des limites liées à la perception et l'évaluation de la tâche selon ces deux dimensions.

8.4.3 Évaluation subjective

Les évaluations concernant la phase balistique ne montrent aucune différence entre les trois techniques. DIOD étant conçue pour décomposer la tâche de positionnement, dont on a montré qu'elle était efficace durant la phase de contrôle, ce résultat est surprenant. Nous pensions que les utilisateurs percevraient la possible décomposition du mouvement durant la phase balistique comme un inconvénient majeur. Néanmoins, les faibles différences observées semblent indiquer que les utilisateurs ont pu rapidement s'adapter aux caractéristiques de la technique.

D'un autre côté, concernant l'évaluation durant la phase de contrôle, les résultats indiquent que les utilisateurs ont clairement ressenti les bénéfices de la technique DIOD. Ils n'ont pas perçu la décomposition comme gênante. Au contraire, les commentaires que nous avons recueillis montre que « *Le contrôle offert par la technique à la fin de la tâche est totalement adapté* ». Avec MD, de nombreux utilisateurs nous ont indiqué que « *La technique est frustrante en fin de tâche* ». D'autres, nous ont également indiqué que « *Bien que PRISM soit mieux que MD, elle reste frustrante en fin de tâche* ». Les discussions informelles que nous avons eues avec les utilisateurs nous conduisent à penser que, si la fonction de correspondance non-linéaire permet d'atteindre une meilleure précision, elle nécessite d'effectuer des mouvements de plus grandes amplitudes. Cela entraîne une fatigue plus importante et une frustration accrue due à la difficulté à effectuer les ajustements désirés.

De plus, l'évaluation subjective indique une nette préférence des utilisateurs pour la technique DIOD.

8.5 Conclusions

La technique DIOD semble fournir un contrôle approprié tout au long de la tâche de positionnement 3-D. Elle permet un positionnement grossier aussi rapide que des

techniques classiques comme la manipulation directe. Elle semble également permettre d'atteindre une meilleure précision qu'avec des techniques basées sur un contrôle dynamique du CDR. De plus, cette technique est particulièrement appréciée des utilisateurs.

Nous utilisons cette technique dans l'ensemble des tâches de modélisation nécessitant le positionnement d'entités géométriques :

- Positionnement d'un modèle 3-D dans l'espace ;
- Positionnement de points de contrôle d'une courbe 3-D. Un exemple d'une courbe 3-D est donnée en Figure 8.6 ;
- Positionnement d'une contrainte de déformation.

De plus, le seuillage sur la vitesse minimale dans la méthode de filtrage DIOD permet des phases de manipulation linéaires et planaires. En effet, lorsque l'utilisateur ne dépasse pas la vitesse maximale, les dimensions sont filtrées indépendamment les unes des autres. Le seuillage permet alors de manipuler un DDL ou deux DDL à la fois. Ceci permet de manipuler facilement dans un plan ou selon un segment. Avec DIOD, il est donc possible de définir des courbes planaires sans activer des contraintes explicites. Il est également possible de définir des lignes droites encore une fois sans utiliser de contraintes explicites.

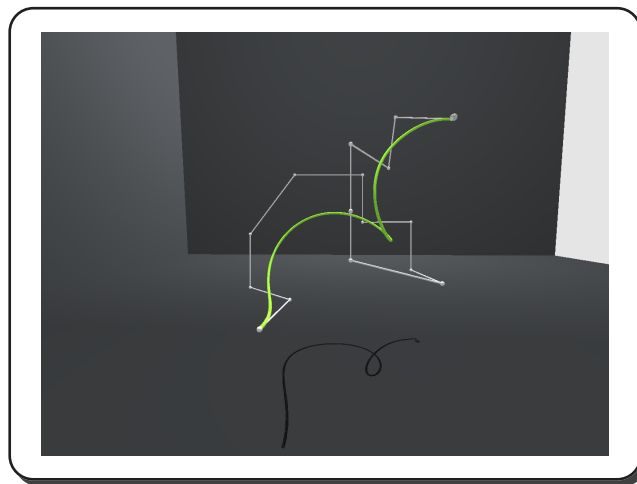


FIGURE 8.6 – Une courbe de Bézier 3-D est représentée par un ensemble de points de contrôle dans l'espace.

Nous proposons donc d'étudier et d'évaluer par des mesures quantitatives l'utilisabilité de cette technique dans des tâches de création de courbes 3-D. Dans le cas de positionnements relatifs (*i.e.* de positionnements d'un objet, l'objet cible, par rapport à un autre objet, l'objet manipulé), nous proposons de nous intéresser à la relation pouvant exister entre l'orientation de l'objet cible et le repère utilisé pour décomposer le

mouvement. En effet, la décomposition se fait actuellement selon le repère associé au monde virtuel. Toutefois, lorsque l'orientation de l'objet cible n'est pas alignée sur les axes du repère du monde virtuel, il est possible que ce repère ne soit pas adapté pour la décomposition des mouvements de l'objet manipulé. Nous proposons donc d'étudier l'influence de l'orientation de l'objet cible sur les stratégies de décomposition et leurs influences sur les performances. Les résultats obtenus permettrait de proposer une extension de DIOD décomposant le mouvement dans le repère le plus adapté pour chaque objet virtuel.

CrOS : Cursor On Surface

Sommaire

9.1	Introduction	154
9.2	Contexte	156
9.2.1	Méthodes existantes	156
9.2.2	Bilan	159
9.3	CrOS	159
9.3.1	Principe de la méthode	159
9.3.2	Compatibilité directionnelle	161
9.3.3	Extensions	161
9.4	Conclusions	164

En modélisation géométrique, de nombreux opérateurs nécessitent de manipuler sur la surface même des objets 3-D. C'est le cas par exemple pour l'opérateur de sculpture. Pour ce type de tâches (pouvant être assimilées à des tâches de suivi de trajectoires 3-D), il est possible d'utiliser des techniques classiques comme la manipulation directe [Shneiderman 97], le Go-Go [Poupyrev 96] ou encore HOMER [Bowman 97] et ses extensions [Wilkes 08]. Néanmoins, comme nous avons pu le voir dans le Chapitre 1, ces techniques n'offrent pas le niveau de précision nécessaire aux utilisateurs.

Nous proposons, dans ce chapitre, une nouvelle technique appelée *Cursor On Surface* (notée CrOS). Avant de présenter cette technique, nous présentons brièvement le contexte de ce travail.

9.1 Introduction

Les tâches de suivi de trajectoires 3-D sont en général complexes à réaliser, particulièrement lorsqu'elles sont effectuées au travers de techniques de manipulation sans contraintes. En l'absence de toute assistance, il est pratiquement impossible de manipuler sur la surface même de l'objet. Les concepteurs introduisent alors des contraintes virtuelles dans le but d'offrir aux utilisateurs un meilleur contrôle et une meilleure précision. Elles permettent de contraindre la manipulation sur la surface de l'objet en repositionnant automatiquement l'outil d'interaction à la position la plus proche sur la surface. Un exemple est donné en Figure 9.1. Toutefois, cette solution ne permet pas d'introduire de support physique au cours de la manipulation.

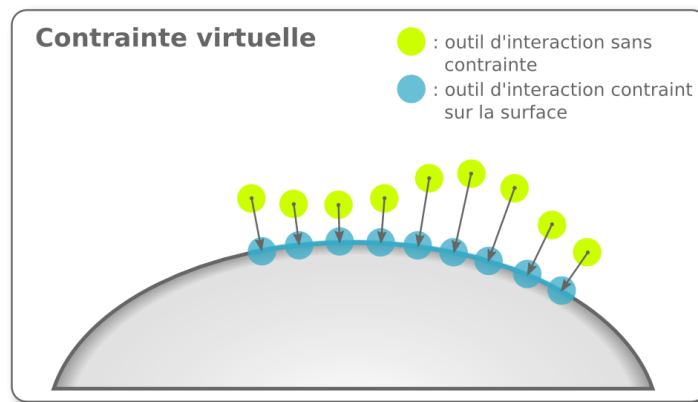


FIGURE 9.1 – Illustration de l'utilisation d'une contrainte virtuelle lors d'une tâche de peinture sur une surface. L'outil d'interaction est repositionné sur la surface de l'objet tout au long de la phase de manipulation.

Or, les résultats que nous avons obtenus lors de l'étude sur l'influence du support physique suggèrent qu'il peut permettre d'améliorer la précision des utilisateurs. Une solution consiste à introduire des contraintes physiques grâce à l'utilisation d'un dispositif à retour d'effort. Ce dernier permettrait de toucher la surface de l'objet lors de la manipulation. Le principe est identique à celui proposé dans les applications In-Touch [Gregory 00] et ArtNova [Foskey 05]. Dans ce cas, en plus de travailler à la surface même de l'objet, la surface de l'objet sert de support physique tout au long de la manipulation. La surface de l'objet étant localement assimilable à un plan, la tâche initialement de nature 3-D, devient alors une tâche de dimension deux. La réduction de la dimensionnalité de la tâche de trois à deux peut permettre d'accroître la précision de l'utilisateur [Hinckley 94]. Néanmoins, l'utilisation de dispositifs à retour d'efforts pose des problèmes. Notamment, la manipulation sur une surface fortement courbée – particulièrement dans des zones convexes – peut rendre le contrôle du geste difficile.

Nous proposons donc à présent, une technique permettant d'interagir sur la surface d'un objet 3-D en utilisant la simulation d'une interface tactile. Le support physique introduit par la présence de l'écran se trouvant à portée de main permet :

- De réduire la dimension de l'espace de manipulation et de la tâche. Réduire la dimensionnalité de la tâche peut permettre de réduire la charge cognitive induite durant la réalisation de la tâche et d'améliorer la précision [Hinckley 94] ;
- D'améliorer la précision lors de la manipulation par l'introduction d'un support physique ;
- D'accroître le confort de l'interaction en réduisant la fatigue induite.

L'idée est la suivante. L'utilisateur effectue les opérations de modélisation, non pas en utilisant une technique de manipulation directe, mais en utilisant un curseur se déplaçant sur la surface même de l'objet. Comme nous l'avons indiqué, l'utilisation de dispositifs à retour d'effort pour se déplacer à la surface d'un objet 3-D revient à se déplacer localement dans un espace 2-D. Partant de ce principe, il est donc possible de définir un déplacement sur la surface de l'objet 3-D à partir d'un vecteur dans un espace 2-D (dans notre cas, via la simulation d'une surface tactile). Cette affirmation n'est vraie que dans le cas où l'objet est une 2-variété¹. Un exemple est donné en Figure 9.2.

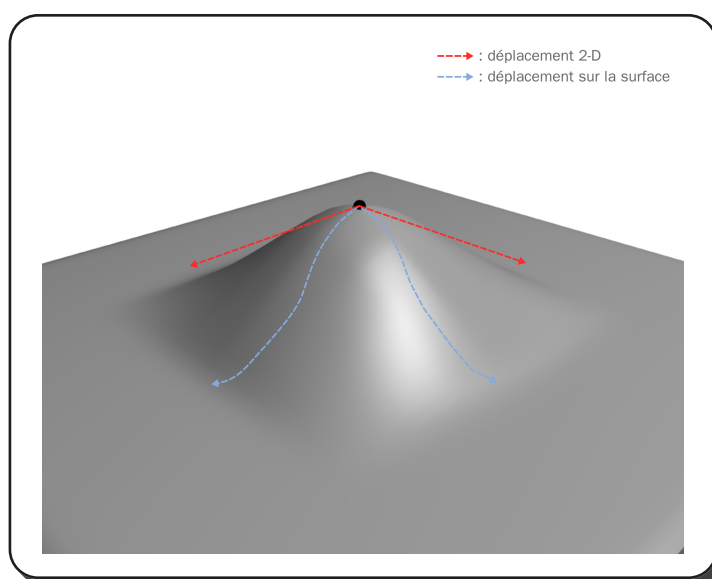


FIGURE 9.2 – Deux exemples de déplacements, définis dans un plan, transformés en déplacements sur la surface de l'objet. Ces déplacements géodésiques permettent de déplacer le curseur se trouvant sur le sommet de la bosse.

1. Les 2-variétés sont des surfaces, qui dans notre cas seront plongées dans $\mathbb{R}^3$. Ces surfaces ne possèdent pas d'auto-intersection.

Avant de présenter l'algorithme permettant de transformer les déplacements d'un point de contact sur une surface tactile en déplacements du curseur sur la surface d'un objet 3-D, nous présentons les méthodes similaires existantes.

9.2 Contexte

Dans un premier temps, nous présentons les méthodes existantes avant de décrire les contraintes que nous nous fixons lors de la mise en place de notre technique d'interaction.

9.2.1 Méthodes existantes

D'un point de vue géométrique, ce problème est assez similaire au problème de la paramétrisation de surfaces. La paramétrisation de surfaces est principalement utilisée afin de plaquer des textures sur des surfaces 3-D. Dans ce cas, une solution possible à notre problème pourrait consister à utiliser des algorithmes de paramétrisations locales ou globales de la surface [Hormann 08]. Toutefois, ces algorithmes sont en général coûteux en temps [Sheffer 06] et ne permettraient pas de garantir l'interactivité. L'introduction d'une latence dans la manipulation peut avoir des conséquences néfastes sur les performances. Une étude conduite par Pavlovych *et al.* [Pavlovych 09] montre qu'introduire une latence peut détériorer significativement les performances dans une tâche de pointage. La latence réduit la précision de l'utilisateur et augmente les temps de réalisation. Le contrôle des propriétés de la paramétrisation est un problème complexe. Ce type de solution ne permet donc pas de garantir que la relation définie entre les déplacements du point de contact et les déplacements du curseur sur la surface soient compréhensibles par l'utilisateur.

Cette propriété, essentielle, correspond aux principes de la théorie de la **compatibilité du stimuli et de la réponse** (notée SRC) [Worringham 98] présentée au Chapitre 2. Dans le cas de la manipulation d'un curseur (2-D ou 3-D), on parle plutôt de la **compatibilité directionnelle du stimuli et de la réponse** (notée DSRC), un cas particulier de la SRC. Dans leur étude, Worringham *et al.* [Worringham 98] montrent que dans le cas où les mouvements du pointeur sont dans la même direction que les mouvements effectués par le manipulateur, les temps de réaction sont meilleurs, les mouvements plus courts et les mouvements de corrections moins nombreux. Il faut donc que la relation entre les mouvements du point de contact sur la surface et les mouvements du curseur sur la surface soit compréhensible.

Dans des contextes particuliers, tels que la navigation 3-D, des techniques similaires à la paramétrisation de surfaces ont été proposées. Hanson *et al.* [Hanson 97] proposent

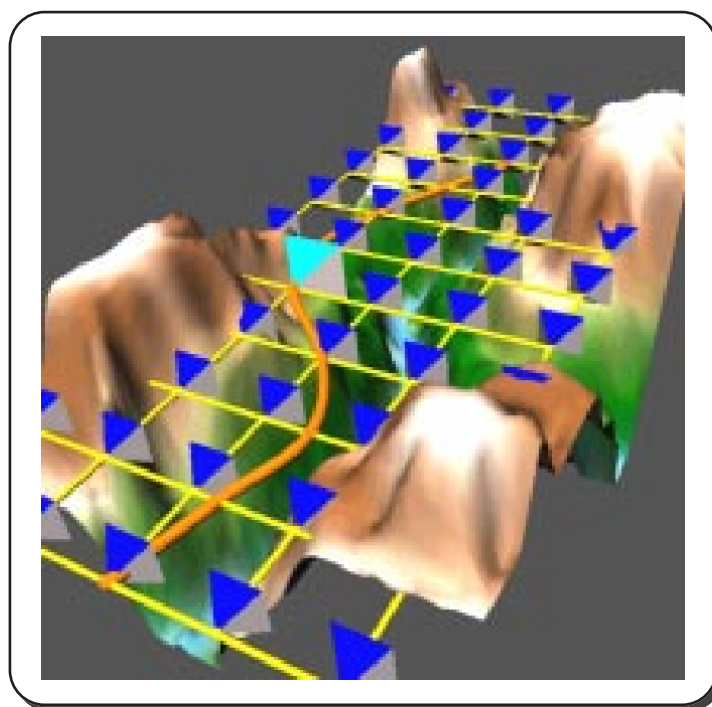


FIGURE 9.3 – L'utilisateur se déplace sur une surface en utilisant un périphérique 2-D pour naviguer dans une scène 3-D. Cette surface est indépendante de la structure de la scène virtuelle. Les sommets, en plus de définir la position du maillage, définissent l'orientation de la caméra (représentée par les pyramides). L'utilisateur peut se déplacer dans la scène, l'orientation de la vue se calculant automatiquement. Image extraite de [Hanson 97].

une méthode pour naviguer dans un espace 3-D en utilisant des périphériques 2-D, comme la souris. Leur méthode consiste à utiliser une surface de navigation afin de se déplacer dans une scène 3-D. Un exemple est donné en Figure 9.3. L'utilisateur modifie sa position dans l'environnement en modifiant sa position sur la surface de navigation. Leur approche pour calculer le déplacement sur la surface de navigation est assez similaire aux approches utilisées pour faire de la paramétrisation. Elle consiste à déplier la surface sur une surface 2-D. L'utilisateur va alors pouvoir se déplacer sur la surface de navigation en se déplaçant sur la surface dépliée. Un exemple est donné en Figure 9.4.

La technique proposée par Hanson *et al.* peut introduire des incompatibilités directionnelles du stimuli et de la réponse. Un cas de figure est illustré sur la Figure 9.5. Lorsque l'utilisateur passe de la face 3 à la face 4, un déplacement vers l'avant va se traduire par un déplacement vers l'arrière sur la surface. Pour tous les déplacements effectués sur la face 4, les directions avant et arrière sont inversées. Cette inversion peut rendre la navigation problématique.

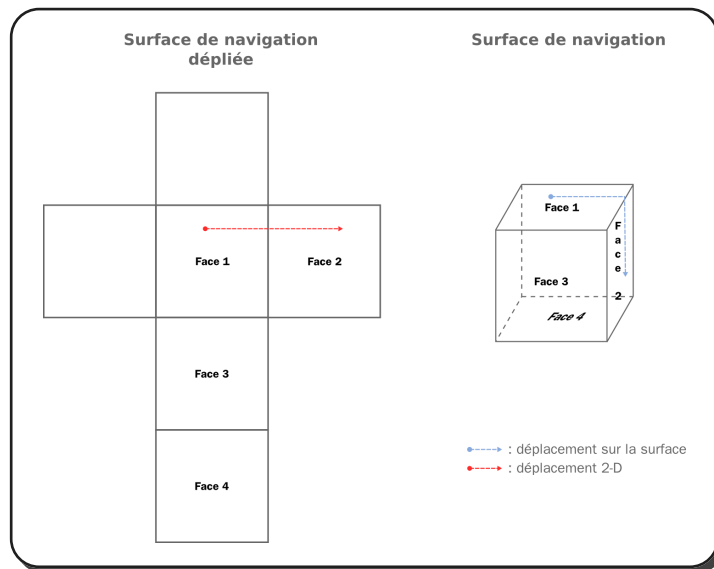


FIGURE 9.4 – Dans cet exemple, la surface de navigation est un cube. En se déplaçant sur la surface dépliée, l'utilisateur peut se déplacer sur la surface de navigation.

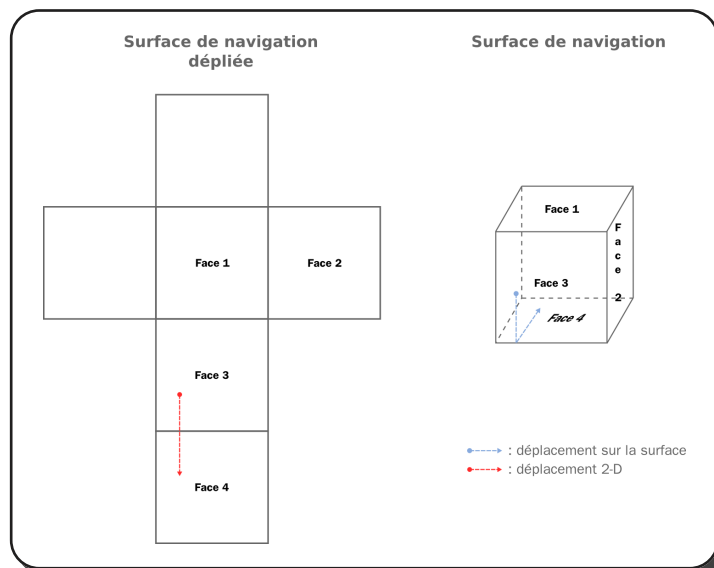


FIGURE 9.5 – En se déplaçant sur la face 4, les directions avant et arrière sont inversées.

9.2.2 Bilan

Comme nous l'avons indiqué dans la section précédente, la difficulté liée à la mise en place de cette technique réside dans le calcul du déplacement du curseur sur la surface de l'objet 3-D à partir de déplacements 2-D, donnés ici par un point de contact sur la surface de l'écran. Cette transformation, effectuée par la **fonction de placage**, va altérer la relation entre le déplacement que va effectuer l'utilisateur au travers du point de contact et le déplacement effectué par le curseur sur la surface. Cette relation doit rester compréhensible par l'utilisateur tout au long de la manipulation.

Contrainte 01 : l'algorithme doit essayer de respecter au mieux les principes de la SRC et en particulier le principe de DSRC.

La méthode de calcul du placage des mouvements 2-D peut avoir un coût en temps non négligeable. L'introduction d'une latence peut avoir une influence négative sur les performances des utilisateurs [Pavlovych 09]. Le calcul du déplacement doit donc introduire un temps de calcul faible afin de n'introduire aucune latence.

Contrainte 02 : l'algorithme doit introduire un temps de calcul faible afin de maintenir l'interactivité et ne pas introduire de latence.

Nous proposons un algorithme de placage simple respectant ces deux contraintes. Dans la section suivante, nous présentons le principe de la méthode mise en oeuvre pour le calcul de la fonction de placage. Les détails de la mise en oeuvre de cette méthode sont décrits dans l'Annexe [A](#).

9.3 CrOS

Nous décrivons le principe de la mise en oeuvre de la fonction de placage. Nous présenterons par la suite une méthode permettant de garantir une compatibilité directionnelle tout au long de la manipulation avant de décrire deux extensions à ce paradigme de manipulation.

9.3.1 Principe de la méthode

Afin de répondre aux deux contraintes que nous nous sommes fixées, nous proposons une solution se basant uniquement sur la géométrie locale de la surface de l'objet. Pour un point donné sur une surface, chaque direction contenue dans le plan tangent à la surface en ce point, peut être associée à un chemin géodésique déterminé par la géométrie locale de la surface (cf. Figure 9.2). Il est important de noter que cette affirmation n'est vraie que pour des 2-variétés.

Le chemin géodésique correspondant à un déplacement 2-D peut être considéré comme l'intersection locale entre un plan (que nous appellerons **plan géodésique**) contenant le déplacement 2-D passant par la position du curseur. La Figure 9.6 illustre la relation entre l'intersection du plan et de la surface et le chemin géodésique associé au déplacement 2-D. Toutefois, il existe une infinité de plans satisfaisant ces contraintes.

Dans le but de définir le plan géodésique, nous introduisons une contrainte supplémentaire. La normale du plan doit être contenue dans le plan dans lequel l'utilisateur manipule (appelé le **plan de contrôle**). Dans notre cas, le plan de contrôle est le plan contenant la surface utilisée pour mettre en oeuvre la simulation d'une interface tactile (*i.e.* le plan contenant la surface de l'écran se trouvant à portée de main). Cette contrainte permet de garantir une bonne compatibilité directionnelle entre le déplacement défini par l'utilisateur et le déplacement du curseur sur la surface.

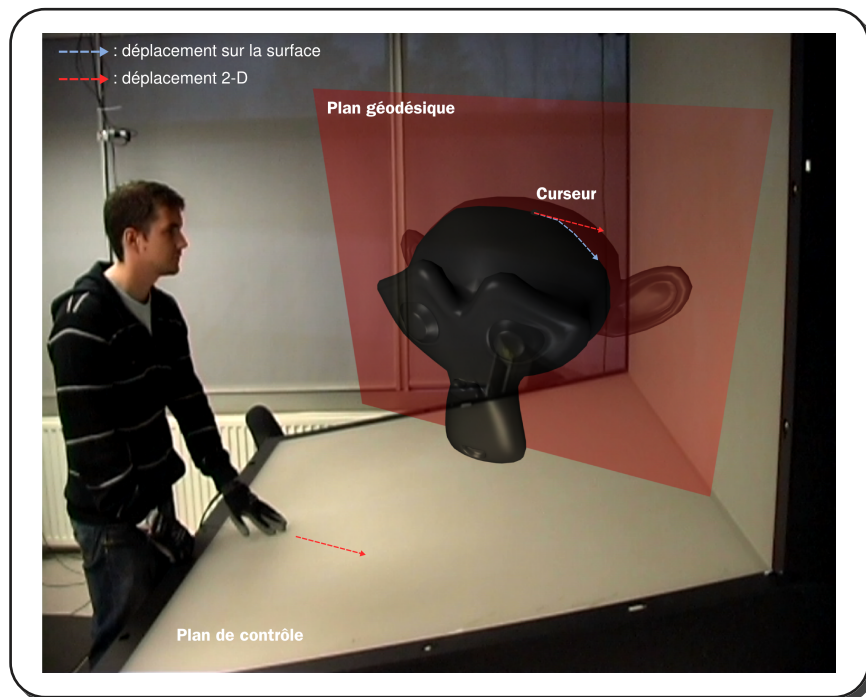


FIGURE 9.6 – Le plan utilisé contient le déplacement 2-D et passe par la position courante du curseur. Nous pouvons constater que le chemin géodésique associé au déplacement est une partie de l'intersection entre le plan et l'objet 3-D.

La méthode que nous utilisons pour calculer le chemin géodésique est la suivante :

1. Calculer localement l'intersection entre le plan et l'objet. Le calcul de l'intersection locale permet de réduire les temps de calcul ;
2. Le curseur est déplacé le long de cette intersection jusqu'à ce que la longueur

du déplacement géodésique soit égale à la longueur du déplacement effectué par l'utilisateur.

9.3.2 Compatibilité directionnelle

Dans l'algorithme que nous proposons, il peut y avoir des moments où la compatibilité directionnelle entre le mouvement de l'utilisateur et le mouvement du curseur n'est pas respectée. Un exemple d'incompatibilité est donné en Figure 9.7. Les incompatibilités apparaissent lorsque l'angle entre la normale du plan de contrôle et la normale de la surface à la position du curseur est supérieur à 90° (*i.e.* lorsque le produit scalaire est inférieur ou égal à 0).

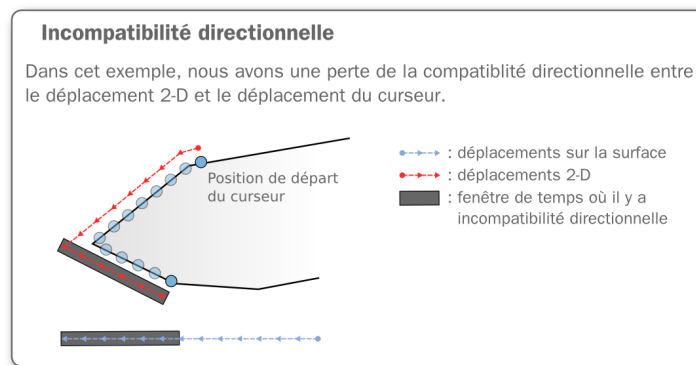


FIGURE 9.7 – Dans la fenêtre de temps indiquée, nous pouvons voir une incompatibilité entre le déplacement effectué par l'utilisateur et le déplacement du curseur.

Une méthode simple, consistant à réorienter progressivement la surface, permet d'éviter ces incompatibilités. Lorsque le produit scalaire devient négatif ou nul, il suffit de calculer la transformation géométrique entre la normale de la surface de l'objet à la position du curseur et la normale du plan de contrôle. Une fois la transformation calculée, par le biais d'une simple animation, il est alors possible de ré-orienter progressivement l'objet afin de rétablir la compatibilité directionnelle.

9.3.3 Extensions

La technique CrOS permet de manipuler sur la surface d'un objet à l'aide d'un point de contact. Nous proposons deux extensions. La première consiste à déplacer deux curseurs sur la surface introduisant ainsi la possibilité d'une interaction bi-manuelle. La deuxième modifie le CDR (*Control :Display Ratio*) afin d'enrichir ce paradigme d'interaction en ajoutant la possibilité d'avoir un retour pseudo-haptique.

Interaction bi-manuelle

La technique que nous proposons permet de déplacer un curseur sur une surface à l'aide d'un point de contact. Sur notre dispositif, nous sommes capables de suivre deux points de contact, un point de contact par main. Nous proposons d'étendre la technique CrOS aux interactions à deux mains. L'utilisateur peut alors déplacer deux curseurs simultanément sur la surface de l'objet. Cette extension est appelée **Bimanual CrOS** (notée B-CrOS).

Dans un premier temps, en affectant le même outil de modélisation aux deux curseurs, nous pouvons induire un parallélisme dans la manipulation. L'utilisateur peut effectuer la même opération en deux positions différentes de la surface. C'est le cas pour la coloration de la surface ou encore la sculpture.

Il est également possible d'accroître le nombre de DDL de la tâche pouvant être manipulés simultanément. Dans le cas de la déformation, en utilisant CrOS, l'utilisateur peut définir la position de la déformation sur la surface et déformer la surface selon la normale de la surface à la position du curseur. En utilisant deux curseurs, nous pouvons considérer que la position de la déformation est définie par le milieu du chemin géodésique rejoignant les deux curseurs. La direction de la déformation est définie par la normale à la surface au milieu du chemin géodésique. Enfin, le rayon de la zone de déformation est défini par la distance euclidienne entre les deux curseurs. Dans ce cas, il est possible de manipuler le rayon en plus de la position de la zone de déformation.

Pseudo-haptique

Le pseudo-haptique consiste à simuler des sensations haptiques sans utiliser de périphériques à retour d'effort [Lécuyer 04]. Pour se faire, Lécuyer *et al.* proposent une technique consistant à modifier dynamiquement le mouvement du curseur de la souris – *i.e.* modifier dynamiquement le CDR. Dans le cas de la souris, le CDR est défini de la façon suivante :

$$CDR = \frac{|\text{déplacement souris}|}{|\text{déplacement du curseur}|}$$

Dans leur étude, Lécuyer *et al.* se placent dans un contexte de manipulation d'images vues du dessus. Ils ont montré qu'en modifiant de manière appropriée le CDR, il est possible de simuler la sensation de passer sur des bosses ou dans des trous. Le principe est illustré en Figure 9.8.

Nous avons mis en place l'utilisation d'un retour pseudo-haptique dans le cas de la manipulation avec CrOS. Il peut être utile de ressentir les reliefs du terrain afin d'amé-

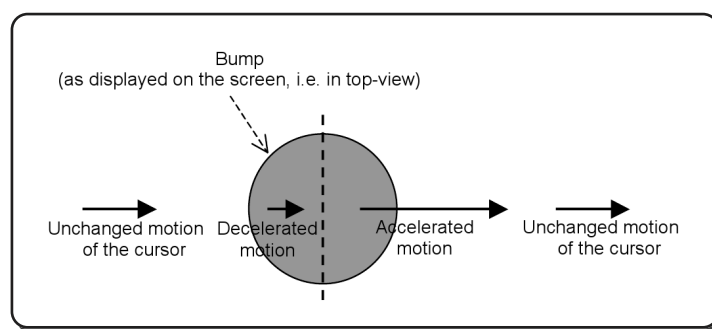


FIGURE 9.8 – Illustration des changements de CDR pour simuler une bosse. Image extraite de [Lécuyer 04].

liorer la perception de la forme sur laquelle travail l'utilisateur. Toutefois, dans notre cas, les données à disposition pour le calcul du CDR sont différentes de celles utilisées par Lécuyer *et al.*. Nous proposons donc d'adapter l'algorithme de calcul du CDR en utilisant la courbure moyenne sur la surface.

Le principe est le suivant. Si le curseur se déplace d'un point de la surface à un point où la courbure est plus élevée, le curseur **grimpe** sur une partie de la surface. À l'inverse, si le curseur se déplace à un point de la surface où la courbure est plus faible, le curseur **descend** une partie de la surface. Afin de donner l'illusion de grimper sur une surface, il faut que le CDR soit supérieur à 1. Dans ce cas le déplacement dans l'espace d'affichage est moins important que le déplacement dans l'espace moteur. Inversement, nous pouvons créer l'illusion que le curseur descend une partie de la surface en s'assurant que le CDR soit inférieur à 1. Un exemple est donné en Figure 9.9.

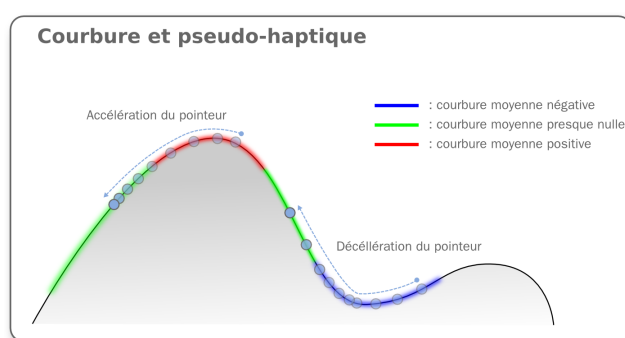


FIGURE 9.9 – Exemples de la modification de la vitesse de déplacement du pointeur en fonction de la courbure moyenne sur la surface.

Nous avons à disposition la courbure moyenne en tout point du maillage. La cour-

bure moyenne sur l'intégralité de chaque face du maillage est approximée par moyenne pondérée de la courbure en chaque sommet de la face. Dans l'exemple donné en Figure 9.10, la courbure moyenne à la position du curseur est obtenue de la manière suivante :

$$Courbure_{curseur} = \frac{A_P \times Courbure_P + A_Q \times Courbure_Q + A_R \times Courbure_R}{A_P + A_Q + A_R}$$

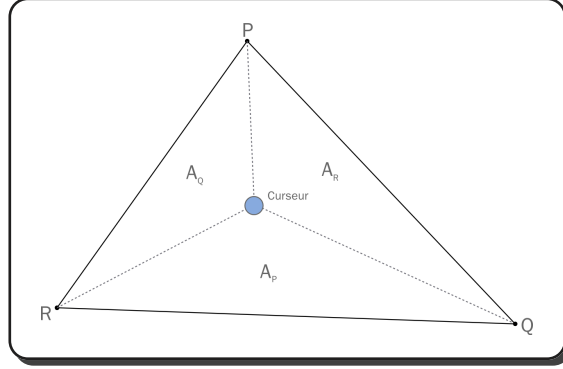


FIGURE 9.10 – Calcul de la courbure en se basant sur une moyenne pondérée.

Pour calculer la nouvelle position du curseur avec retour pseudo-haptique, nous utilisons la méthode suivante. Dans un premier temps, avec la technique CrOS, nous calculons la position du pointeur sans modification du CDR. Une fois la position obtenue, nous calculons la courbure moyenne en cette position (notée $Courbure_{np}$) ainsi que la courbure à la position initiale du curseur (notée $Courbure_{curseur}$). Afin de créer le retour pseudo-haptique, nous modifions la longueur du vecteur de déplacement sur la surface ($D_{surface}$) en utilisant la formule suivante :

$$\|D'_{surface}\| = \|D_{surface}\| \times e^{-k \times (Courbure_{np} - Courbure_{curseur})} \quad (9.1)$$

k est une valeur strictement positive fixée permettant de définir l'importance du retour pseudo-haptique. Plus sa valeur est élevée, plus l'effet du retour pseudo-haptique sera perceptible par l'utilisateur.

9.4 Conclusions

Nous avons proposé une technique permettant de manipuler à même la surface d'un objet 3-D. Cette technique se base sur la simulation d'une interface tactile. CrOS respecte les deux contraintes principales que nous nous sommes fixées. La technique essaie

de respecter au mieux les principes de la DSRC. De plus, elle introduit un faible temps de calcul ce qui n'induit aucune latence durant la manipulation.

D'après les résultats de l'étude menée dans le Chapitre 6, le support physique introduit par l'utilisation de la simulation d'une interface tactile permet d'accroître la précision des utilisateurs. De plus, le support physique permet également de réduire la fatigue introduite durant la manipulation. Ceci est d'autant plus intéressant que cette technique est utilisée pour un grand nombre des opérateurs de modélisation disponibles dans nos applications.

L'introduction d'interaction bi-manuelle permet d'enrichir le nombre de DDL manipulable simultanément. De plus, il est possible d'enrichir les outils de modélisation pouvant être mis en place (par exemple un outil de coupure sur une surface, ...). Nous envisageons donc de proposer de nouveaux outils de modélisation exploitant les deux points de contact sur la surface même de l'objet 3-D. L'introduction d'un retour pseudo-haptique peut permettre une meilleure appréhension de la géométrie de l'objet. En effet, les méthodes d'éclairage classiques ne permettent pas une bonne visualisation des faibles variations de la géométrie de l'objet 3-D. L'introduction d'un retour pseudo-haptique, faisant varier la vitesse de déplacement du curseur, peut permettre de mieux appréhender la géométrie de la surface sur laquelle l'utilisateur manipule. Nous envisageons d'évaluer ces différentes caractéristiques par la mise en place d'études expérimentales spécifiques.

Dans le chapitre suivant, nous présentons les deux applications de modélisation développées. La première, VRModelis, est une application qui a été entièrement développée dans le cadre de ce travail. Elle intègre les différentes techniques d'interaction que nous avons vues jusqu'à présent. La seconde est une application de modélisation et ajout de détails dans des scènes naturelles complexes.

Sommaire

10.1 Applications de modélisation en réalité virtuelle	168
10.1.1 VRModelis	168
10.1.2 VR Terrain Editor	172
10.1.3 Conclusion	177
10.2 Intégration des techniques d'interaction	177

Jusqu'à présent, nous avons proposé un ensemble de techniques adaptées aux différentes tâches de modélisation géométrique. À présent, nous allons décrire la manière dont ces techniques (**MP**, **DIOD**, **CRoS** ou encore **TIP**) sont intégrées dans le cadre de deux applications de modélisation en RV :

- **VRModelis** : cette application, entièrement développée dans le cadre de ce travail, constitue les prémices d'une application de modélisation géométrique en environnement de RV. Elle met à disposition certains outils de base, nécessaires pour la modélisation 3-D ;
- **VR Terrain Editor** : il s'agit d'une application de modélisation de terrains et d'ajout de détails dans des scènes naturelles complexes. Cette application utilise la structure de données développée dans le cadre du projet DNA (*Details in NATural scenes*) [Peytavie 09].

Avant de décrire l'intégration des techniques dans ces deux applications, nous présentons ces applications ainsi que les outils dont elles disposent.

10.1 Applications de modélisation en réalité virtuelle

Dans cette section, nous présentons les deux applications, **VRModelis** et **VR Terrain Editor**, dans lesquelles les différentes techniques d'interaction ont été intégrées. L'objectif est de fournir un cadre concret ainsi qu'une mise en situation, la plus réaliste possible, pour l'utilisation de nos techniques.

10.1.1 VRModelis

Nous avons développé, dans son intégralité, les bases d'un modèleur géométrique en environnement de RV. Cette application constitue le principal contexte applicatif dans lequel nous intégrons les techniques d'interaction que nous avons proposés. Cette application n'a pas pour objectif de proposer un modèleur rivalisant avec les modèleurs commerciaux (par exemple CatiaTM ou 3DSMaxTM) ou avec les modèleurs issus de la recherche académique (par exemple VLEGO [Kiyokawa 97]). L'objectif de cette application est d'offrir un cadre applicatif permettant d'intégrer les outils de modélisation proposés dans l'équipe IGG.

Gestion des objets

Dans une scène 3-D, nous trouvons deux types d'objets : les courbes et les surfaces. Les courbes sont définies par deux types de courbes : les courbes de Bézier et les courbes de Catmull-Rom. La première représentation permet de définir des courbes approximant un polygone de contrôle plongé dans un espace 3-D. La seconde représentation permet de définir des courbes d'interpolation. Celles-ci passent par un polygone de contrôle. Chaque point de contrôle définit une contrainte de position – la courbe devant passer par une position donnée dans l'espace – ainsi qu'une contrainte de tangence – la courbe devant être tangente au vecteur défini en ce point.

Les surfaces sont quant à elles représentées à l'aide de cartes combinatoires – en l'occurrence des 2-cartes permettant de représenter des 2-variétés. Cette structure de données, que nous ne détaillerons pas ici, nous permet d'accéder simplement et rapidement à des informations liées à la géométrie de l'objet. Elle permet également d'accéder à des informations d'ordres topologiques, telles que l'information d'adjacence et d'incidence entre les sommets, arêtes et faces du maillage.

Quel que soit le type d'objet, celui-ci est caractérisé par une position et une orientation dans l'espace 3-D. L'utilisateur doit donc pouvoir positionner et orienter n'importe quel objet de la scène.

Créer des objets

Une des opérations fondamentales dans un modelleur géométrique consiste à créer des objets 3-D. Ces objets peuvent être des courbes ainsi que des surfaces. Les représentations que nous utilisons pour définir les courbes se basent sur un polygone de contrôle. Ainsi, comme dans la majeure partie des applications de modélisation, la création de courbes passe par la définition du polygone de contrôle. L'utilisateur définit progressivement le polygone en positionnant les différents points de contrôle. Cette méthode est valable à la fois pour les courbes de Bézier et les courbes de Catmull-Rom.

En ce qui concerne la création de surfaces, nous proposons d'utiliser deux méthodes largement répandues dans le domaine de la modélisation :

- L'extrusion ;
- La révolution.

L'utilisation de ces deux techniques se déroule en deux étapes :

1. La définition du contour d'une forme 2-D (aussi appelé profil) ;
2. La définition de la trajectoire d'extrusion ou encore de l'angle de révolution utilisé pour créer le modèle 3-D.

Un exemple d'extrusion est donné en Figure 10.1. Dans cet exemple, le profil (défini par un cercle) est extrudé selon le chemin indiqué pour créer un modèle 3-D. L'extrusion et la révolution permettent de créer rapidement des objets 3-D complexes.

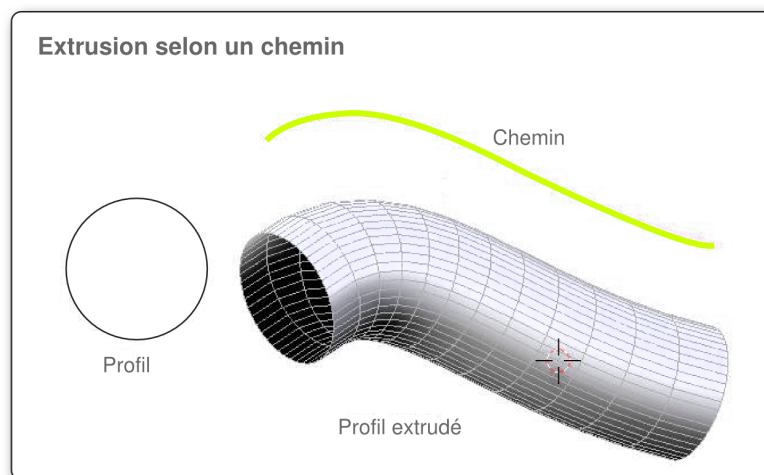


FIGURE 10.1 – Exemple d'extrusion selon un chemin.

Éditer les objets

L'édition des modèles 3-D peut se faire de nombreuses manières. La première possibilité consiste à pouvoir tout simplement manipuler les différents éléments constituant l'objet 3-D, sommets, arêtes et faces. L'utilisateur peut dans ce cas positionner et orienter ces différents éléments.

Les **outils de déformation** sont largement utilisés dans le processus de modélisation. Plusieurs outils de déformation sont disponibles dans cette application. Les trois méthodes de déformation sont : la déformation directe, la déformation par coloration et la sculpture.

Outils de déformations La **déformation directe** se base sur le modèle de déformation DOGME [Borrel 91] [Bechmann 03] [Gain 08], mis en place dans l'équipe IGG. Ce modèle se base sur la notion de contrainte. Une contrainte est définie par quatre paramètres :

1. La position de la contrainte ;
2. La forme de la contrainte ;
3. La fonction de pondération ;
4. La trajectoire de la contrainte.

La déformation s'effectue en appliquant une ou plusieurs contraintes sur le maillage (cf. Figure 10.2). L'utilisateur déforme le maillage en définissant dans un premier temps la forme ainsi que la fonction de pondération de la contrainte. Ceci se fait usuellement par le biais d'un menu, les formes et les fonctions de pondérations étant habituellement prédéfinies. Dans un second temps, l'utilisateur positionne la contrainte sur le maillage. Les sommets se trouvant dans la zone d'influence de la contrainte sont ceux qui seront déformés par la contrainte. Le déplacement des sommets est défini par la trajectoire de déformation. L'utilisateur doit donc pouvoir positionner la contrainte sur le maillage et définir la trajectoire de déformation.

La seconde méthode est appelée **déformation par coloration**. Elle est illustrée en Figure 10.3. Avec cette méthode, l'utilisateur colore la surface à déformer en utilisant un pinceau virtuel. L'intensité de la coloration va définir l'intensité de la déformation. Une zone non colorée ne sera pas déformée, une zone fortement colorée sera fortement affectée par la déformation. Une fois la coloration effectuée, l'utilisateur peut pousser ou tirer localement la surface dans le sens de la normale.

Le dernier outil de déformation est la **sculpture**. La forme de l'outil de sculpture que nous utilisons est sphérique. La sculpture consiste à pousser localement la surface de l'objet dans la direction opposée à la normale. Cet outil permet d'effectuer de petites modifications locales sur la surface de l'objet 3-D.

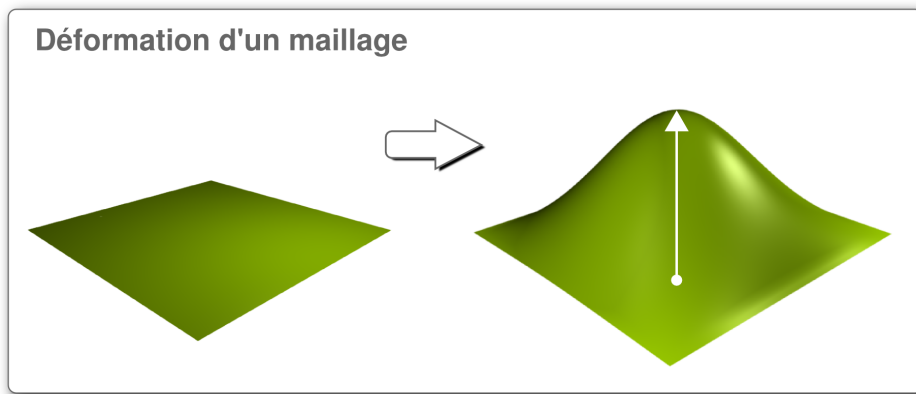


FIGURE 10.2 – À gauche le modèle 3-D original, à droite le modèle déformé. La contrainte est positionnée au centre du plan, la forme de la déformation est sphérique, la fonction de pondération utilisée est une gaussienne et la trajectoire de la déformation est un segment de droite.

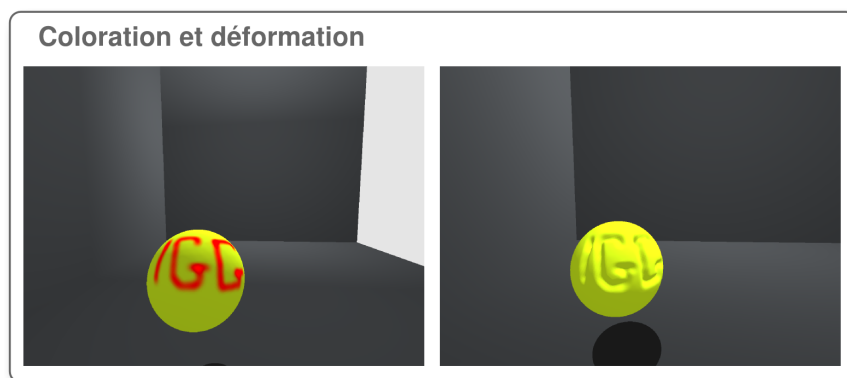


FIGURE 10.3 – Coloration et déformation de la surface d'une sphère.

Coloration et placage de textures Les modifications vues jusqu'ici ne permettent que d'éditer la géométrie de l'objet. Nous proposons à présent deux outils permettant de modifier l'aspect visuel des objets 3-D. Le premier outil permet de colorer localement la surface. Pour se faire, nous utilisons une technique de pinceau virtuel. D'une manière similaire à la déformation directe, le pinceau dispose de plusieurs paramètres modifiables par le biais du menu de l'application :

1. La forme du pinceau : sphérique ou cubique ;
2. La fonction de pondération : linéaire, dirac ou quadric.

Afin de colorer la surface de l'objet, l'utilisateur passe simplement le pinceau sur celle-ci. Un menu permet de choisir la couleur du pinceau. En utilisant la même métaphore, nous étendons la coloration au placage de textures de la surface. Toujours en utilisant le pinceau virtuel, l'utilisateur peut appliquer une texture sur la surface de l'objet. Une illustration de la coloration et de l'application d'une texture est donnée en Figure 10.4.

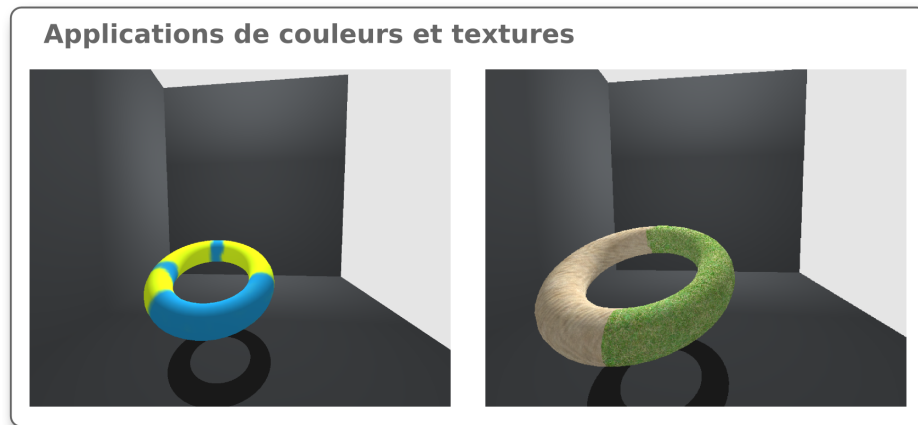


FIGURE 10.4 – À gauche, un exemple d'application d'une couleur sur la surface d'un tore. À droite, l'illustration de l'application de deux textures toujours sur un tore.

Extrusion de faces L'extrusion de faces est très souvent utilisée dans le but de définir la structure globale de l'objet manipulé. Elle permet également d'ajouter des polygones au modèle 3-D de l'objet. L'extrusion consiste à déplacer la face (généralement dans le sens de la normale) et à créer automatiquement de nouvelles faces adjacentes connectées au maillage d'origine. Une illustration du résultat de l'extrusion d'une face est donnée en Figure 10.5.

L'utilisation successive d'extrusions de face permet de définir rapidement la structure de l'objet. Les outils tels que les outils de déformations permettent par la suite de modifier plus finement le modèle 3-D. L'exemple de la création d'un modèle 3-D est donné en Figure 10.6.

10.1.2 VR Terrain Editor

Dans cette application, l'utilisateur peut modéliser des terrains ayant des géométries complexes. La structure de données utilisée ici est issue du projet DNA [Peytavie 09]. Le terrain est représenté par un ensemble de piles de matières (roc, sable, air, *etc.*).

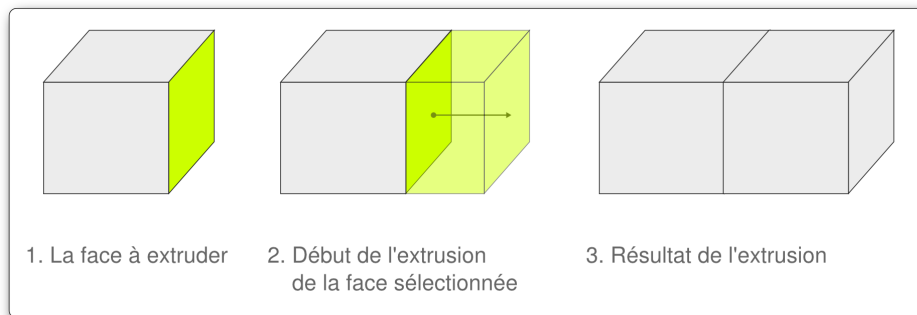


FIGURE 10.5 – Exemple de l'extrusion d'une des faces du cube.

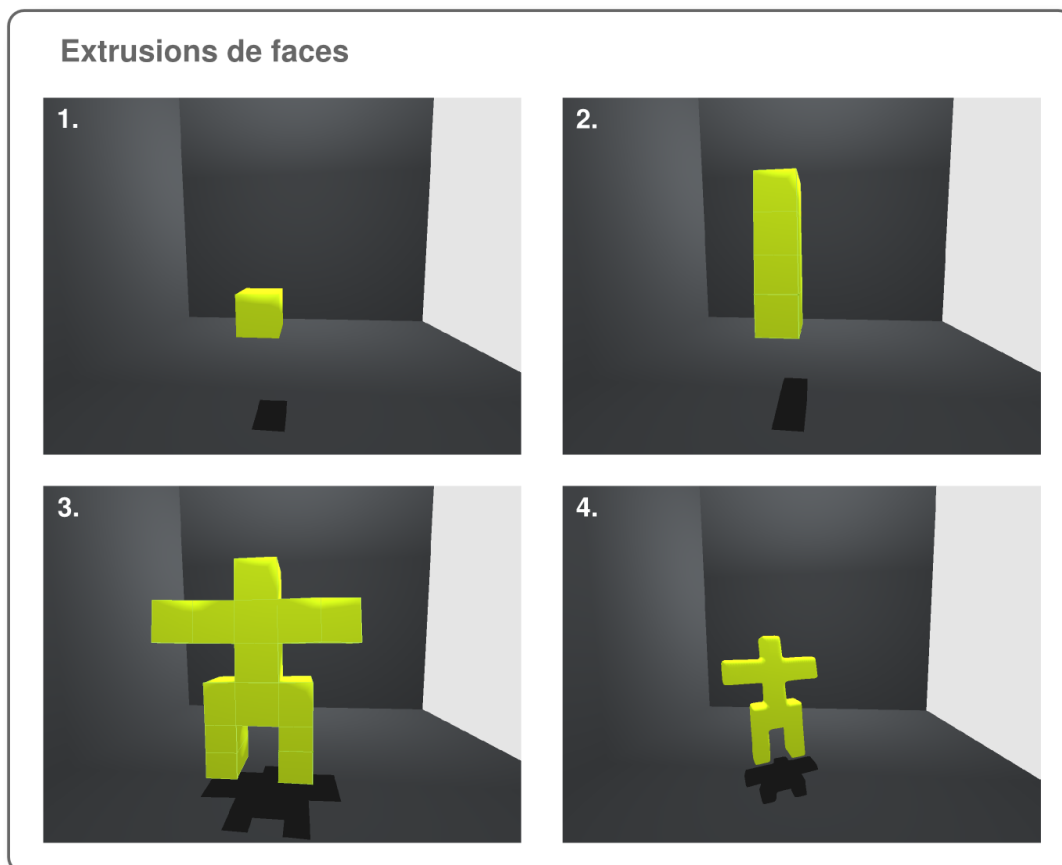


FIGURE 10.6 – L'extrusion successive de faces du modèle 3-D permet de créer grossièrement la forme.

Cette représentation permet de définir des structures au niveau du terrain qui ne sont pas représentables par des structures de données plus classiques, généralement à base de *height-field*¹. Il est donc possible, avec cette structure de données, de modéliser des arches ou encore des grottes.

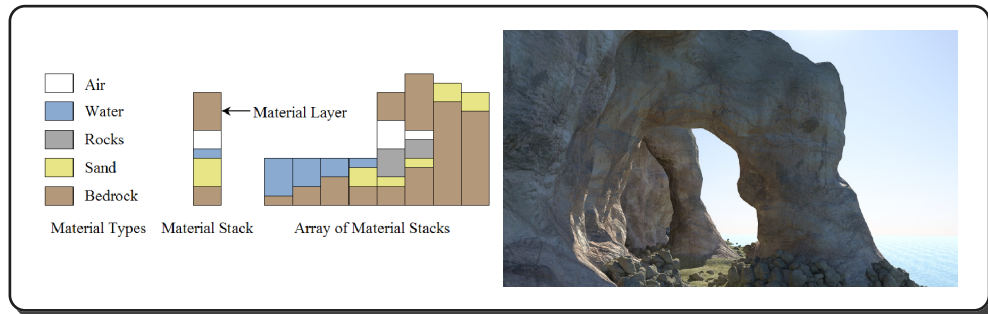


FIGURE 10.7 – À gauche, une illustration de la structure de données basée sur l'utilisation de piles de matières. À droite, l'illustration d'une arche. Images extraites de [Peytavie 09].

Dans cette application, plusieurs outils de modélisation sont disponibles, nous allons les présenter.

Modélisation

Dans cette application, l'utilisateur peut sculpter le terrain en supprimant et en ajoutant de la matière. Le premier outil permet de disperser localement de la matière sur le terrain. La matière est ajoutée de telle manière à ce qu'elle glisse (simulant ainsi le comportement d'éboulement) sur les parties du terrain qui sont en pente. La suppression de matière permet de simuler l'application d'une érosion locale. L'utilisation de ces deux outils est illustrée en Figure 10.8.

Un second outil, n'exploitant pas le comportement physique de la matière, permet d'ajouter des boules de matières à n'importe quel endroit du terrain. Après utilisation de cet opérateur, de la matière peut flotter dans l'espace au dessus du terrain. Cet outil permet, en outre, de créer rapidement des arches (cf. Figure 10.9). Il existe également l'outil inverse, qui va permettre de creuser la matière du terrain. Cet outil sert essentiellement à modéliser des grottes.

1. Une *height-field* (ou carte de hauteur) est une image à niveaux de gris. Le niveau de gris, en chaque pixel (x, y) de l'image, définit l'élévation du terrain à cette même position.

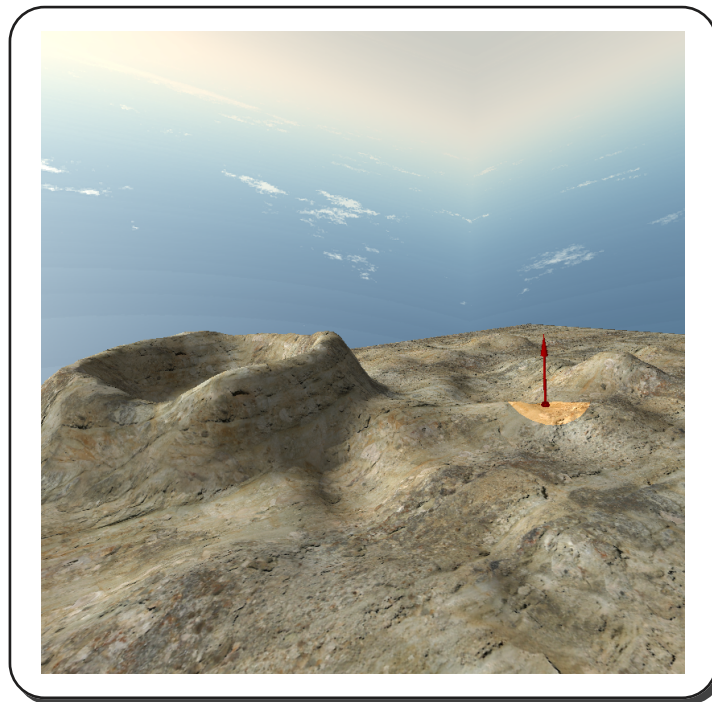


FIGURE 10.8 – Dispersion et érosion de roc sur le terrain afin de créer des irrégularités sur la surface du terrain.

Ajout de détails

L'objectif initial du projet dans lequel s'inscrit le développement de cette application est de pouvoir ajouter des détails dans une scène naturelle. Au travers des outils présentés jusqu'à présent, il est possible de disséminer différents types de matières sur l'ensemble du terrain. Il est également possible de disposer des éléments naturels dans la scène, plus particulièrement, des arbres sur la surface du terrain.

Nous effectuons cet ajout de deux façons différentes :

- Ajout individuel : l'utilisateur ajoute les arbres un à un. L'arbre est virtuellement planté à la position du curseur ;
- Ajout de forêts : l'utilisateur définit une zone sur le terrain. Une fois la zone fermée, elle décrit les limites de la forêt. Les arbres sont alors positionnés automatiquement.

L'illustration d'une scène dans laquelle des arbres ont été ajoutés est donnée en Figure 10.10

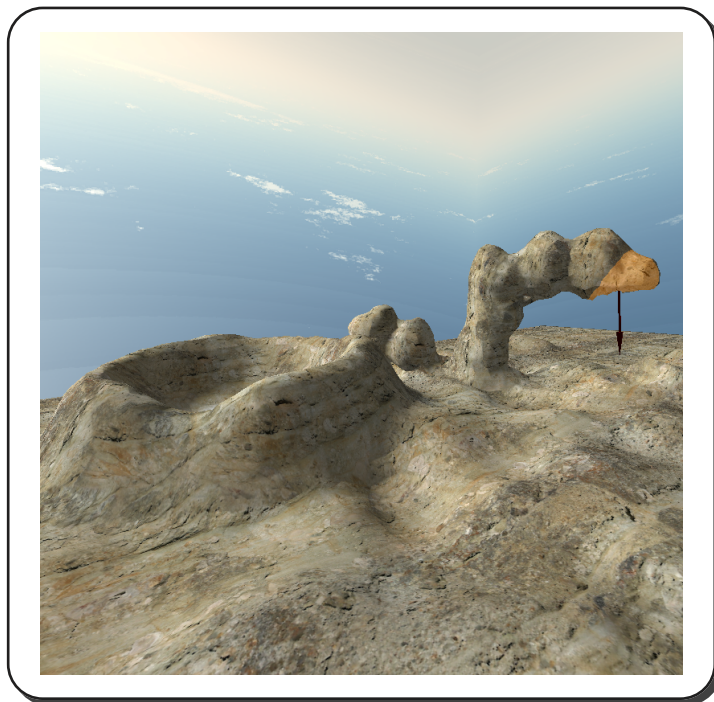


FIGURE 10.9 – Ajout de roc afin de créer une arche sur le terrain.

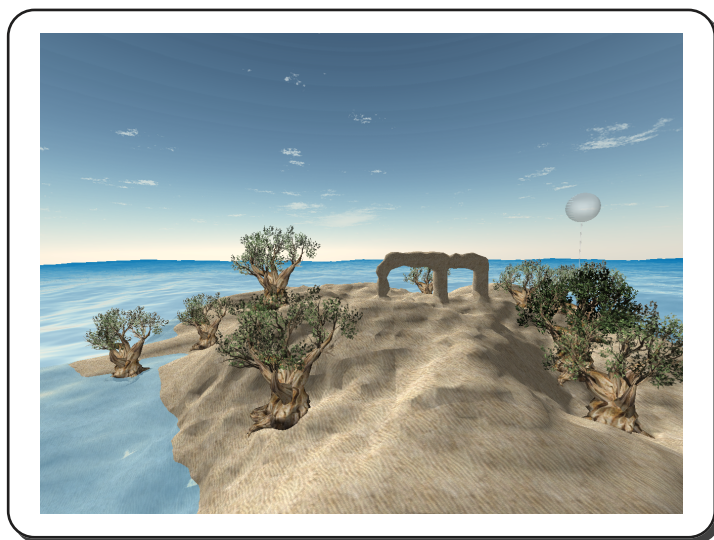


FIGURE 10.10 – Des arbres sont ajoutés afin d'enrichir la scène.

10.1.3 Conclusion

Nous avons présenté deux applications de modélisation, **VRModelis** et **VR Terrain Editor**. Ces deux applications fournissent un contexte applicatif dans lequel nous pouvons intégrer les techniques d'interaction proposées. Nous allons à présent décrire la mise en oeuvre de cette intégration.

10.2 Intégration des techniques d'interaction

Dans les deux applications que nous avons développées, il existe deux modes de manipulation :

- La **manipulation libre** ;
- La **manipulation contrainte**.

L'utilisateur peut passer d'un mode de manipulation à l'autre par le biais du menu.

Dans le premier mode de manipulation, l'utilisateur interagit à l'aide de la manipulation directe. Par conséquent, aucun filtrage n'est appliqué au curseur ou aux mouvements des objets dans le scène virtuelle. Que se soit dans **VRModelis** ou dans **VR Terrain Editor**, toutes les opérations de modélisation (déformation, coloration, texture, ajout de matière, *etc.*) se font avec cette technique. Aucune contrainte virtuelle ou physique n'est introduite au cours de la manipulation.

En mode d'interaction libre, seule l'utilisation de la métaphore du potier et de TIP sont disponibles². Dans les modes de modélisation, la métaphore du potier permet d'orienter rapidement le modèle 3-D dans l'espace. L'objectif est ici de permettre à l'utilisateur de tourner rapidement autour du modèle afin de faciliter l'évaluation des différentes éditions effectuées sur l'objet. Lorsque l'utilisateur se trouve dans un mode nécessitant le positionnement d'entités géométriques (par exemple positionnement 3-D, positionnement de points de contrôle d'une courbe, *etc.*), la technique TIP est disponible afin de faciliter un positionnement précis dans l'espace 3-D.

L'utilisateur peut, à tout moment, entrer dans le mode de manipulation contrainte. C'est le cas notamment lorsque ce dernier a besoin de précision dans la manipulation. C'est dans ce mode de manipulation que nous avons intégré les techniques **CrOS** et **DIOD**.

2. L'utilisation de la simulation d'une interface tactile n'est pas disponible pour l'ensemble des modes d'édition. Nous matérialisons donc la surface tactile par un simple retour visuel, afin d'indiquer à l'utilisateur que la simulation d'une interface tactile est actuellement disponible. Ce retour visuel consiste en une coloration partielle de la surface. Quand la coloration est activée, l'utilisation de la simulation d'une interface tactile est disponible.

Dans ce second mode de manipulation, nous introduisons des contraintes virtuelles et / ou physiques. Pour l'ensemble des outils de modélisation, les mouvements des objets de la scène et du curseur sont filtrés avec la technique DIOD. L'objectif est de fournir un contrôle précis lors des différentes tâches de positionnement dans l'espace 3-D. C'est le cas notamment dans les phases de sculpture du terrain, de positionnement des points de contrôle d'une courbe dans l'espace 3-D, *etc.*

Les contraintes physiques sont introduites toujours en utilisant la simulation d'une interface tactile. La technique MP reste à disposition, toujours afin de permettre une orientation rapide du modèle 3-D. Nous utilisons également la simulation d'une interface tactile dans différentes tâches :

- Les tâches de positionnement 3-D (positionnement de points de contrôle et positionnement du modèle 3-D par exemple) : en plus du filtrage introduit par DIOD, l'utilisateur peut utiliser la technique TIP toujours lors de positionnements d'entités géométriques dans l'espace 3-D ;
- Les tâches de modélisation (coloration, placage de textures, sculpture, déformation, *etc.*) : les utilisateurs peuvent éditer les modèles 3-D en utilisant la technique CrOS. Le curseur se trouve alors contraint sur la surface de l'objet 3-D ou du terrain. L'édition se fait alors au travers de ce curseur.

Le support introduit par la surface tout au long de la manipulation peut ici permettre de réduire la fatigue. Ce mode de manipulation peut être utile lors de longues sessions de manipulation. Dans le cas de la technique CrOS, l'objectif est également de fournir plus de précision, particulièrement lors de phases d'édérations fines.

Dans ce dernier chapitre, nous présentons un bilan de l'ensemble des travaux que nous avons effectués. Nous résumons les résultats obtenus, aussi bien durant les différentes études expérimentales que nous avons menées, qu'en matière de proposition de techniques d'interaction. Enfin, nous proposons quelques unes des perspectives que nous envisageons pour ce travail.

11.1 Bilan

Dans le cadre de ce travail, l'objectif était de proposer de nouvelles techniques d'interaction permettant d'améliorer le contrôle de l'utilisateur dans des applications de modélisation géométrique en environnement de RV. Avant de proposer et d'évaluer de nouvelles techniques d'interaction, nous avons essayé, de comprendre certaines des difficultés liées à la manipulation en environnement de RV. Pour cela, nous avons proposé un nouveau modèle théorique conceptualisant la problématique de manipulation en RV à la manipulation des degrés de liberté. En unifiant et enrichissant plusieurs modèles existants, notre modèle tente de prendre en compte des caractéristiques de la manipulation à différents niveaux :

- Au niveau du contrôle moteur du geste, en caractérisant le périphérique d'interaction et la tâche comme des ensembles de DDL. La technique d'interaction est modélisée comme un composant abstrait définissant la relation entre ces deux ensembles. Nous différencions alors plusieurs fonctions : fonction de correspondance, fonction de transfert et fonction de contrainte. Notre modèle propose également de tenir compte des contraintes physiques pouvant être introduites au niveau du périphérique d'interaction ;
- Au niveau de la tâche, en caractérisant non seulement la tâche par la configuration de ses DDL mais également en tenant compte du type de contrôle qu'elle

nécessite. À dimensionnalités égales, deux tâches peuvent nécessiter deux types de contrôle : un contrôle final et un contrôle continu. Cette caractéristique permet de tenir compte de l'influence de l'objectif de précision durant la réalisation de la tâche ;

- Au niveau cognitif, en essayant de formaliser la fonction d'intégration conceptuelle des DDL. Cette dernière permet de tenir compte de la structuration mentale de la tâche.

Dans le cadre de ces travaux, nous nous sommes intéressés à l'introduction de contraintes virtuelles et de contraintes physiques pour l'interaction 3-D. Ce choix se justifie par la volonté d'introduire l'utilisation d'interfaces tactiles pour l'interaction 3-D. Dans le cas de l'interaction 3-D, l'introduction de contraintes virtuelles et / ou physiques nécessitent de décomposer les tâches d'interaction. Nous nous sommes donc également intéressés à l'influence des caractéristiques de cette décomposition sur les performances des utilisateurs.

Afin de pouvoir étudier l'influence de la décomposition, nous devons être à même de pouvoir quantifier précisément la coordination des utilisateurs. Ainsi, nous sommes capables d'identifier des phénomènes d'intégration ou de décomposition de la manipulation des DDL dans des tâches d'interaction 3-D. Les métriques existantes ne permettant pas de quantifier, dans l'absolu, la coordination d'un geste, nous avons besoin d'une nouvelle métrique permettant de quantifier la coordination des mouvements. Nous avons proposé NDC, une métrique permettant de quantifier la coordination de n DDL. Par rapport aux mesures existantes, cette métrique ne dépend pas des caractéristiques de la tâche et permet de quantifier des niveaux de coordination intermédiaires. Après avoir validé cette nouvelle métrique, nous avons mené un ensemble d'études expérimentales dans le but de comprendre l'influence de l'introduction d'un support physique et de la décomposition sur les performances des utilisateurs.

Pour cela, nous avons effectué trois études expérimentales. Les deux premières ont pour objectif de quantifier les apports du support physique et de la décomposition lors de tâches à contrôle final. La dernière a pour objectif de mesurer les apports du support physique et de l'introduction d'une fonction de transfert basée sur la vitesse, dans une tâche à contrôle continu. À l'issue de l'analyse des différents résultats obtenus lors de ces expérimentations, nous avons proposé un ensemble de recommandations :

- Pour une tâche à contrôle final ;
 - Pour les tâches à structure perceptuelle intégrable, les techniques doivent permettre de combiner les DDL de la tâche durant la phase d'approche. L'utilisateur peut alors tirer profit du parallélisme afin de réaliser rapidement cette phase,
 - Pour les tâches à structure perceptuelle intégrable, si l'objectif de la tâche est la précision, les techniques doivent permettre de séparer les DDL de la tâche durant la phase de contrôle. La séparation permet d'accroître le contrôle des utilisateurs

- et d'améliorer la précision,
- Pour les tâches à structure perceptuelle intégrable, si l'objectif de la tâche n'est pas la précision, les techniques ne doivent pas nécessiter de séparer les DDL de la tâche durant la phase de contrôle,
- Pour une tâche à contrôle continu.
 - Si l'objectif est la précision, les techniques doivent introduire une contrainte physique au cours de la réalisation de la tâche,
 - Si l'objectif est la précision, les techniques doivent introduire une fonction de transfert non-linéaire dans les parties curvilignes,
 - Si les sessions de manipulation sont longues, les techniques d'interaction doivent introduire un support physique afin de réduire la fatigue,

Par la suite, nous exploitons ces recommandations afin de proposer et de mettre en place de nouvelles techniques d'interaction, adaptées aux besoins des utilisateurs :

- **La métaphore du potier** : cette technique utilise la simulation d'une interface tactile afin d'orienter un objet dans un espace 3-D. Cette technique semble permettre d'orienter rapidement et précisément un objet. Elle a l'avantage d'introduire un support physique au cours de l'interaction. Ce support physique permet de réduire la fatigue. De plus, l'utilisateur peut aisément tourner l'objet autour d'un axe de rotation spécifique, ce qui apparaît comme un avantage au cours des nombreuses phases de modélisation ;
- **DIOD** : cette technique est inspirée de la manipulation directe. Elle filtre les mouvements des utilisateurs afin d'offrir un niveau de coordination des DDL, adapté aux différentes phases de réalisation de la tâche de positionnement 3-D. Cette technique semble plus précise que les techniques déjà existantes ;
- **CrOS** : cette technique utilise également la simulation d'une interface tactile. Elle permet de déplacer un curseur sur la surface du modèle 3-D. Ce curseur peut être utilisé pour effectuer une grande variété de tâches de modélisation (coloration, sculpture, placage de textures, ...). Cette technique a pour avantage d'introduire un support physique tout au long de la réalisation des tâches. En plus de réduire la fatigue, il permet un gain dans la précision de la manipulation ;

Ces techniques sont intégrées dans deux applications de modélisation développées pour l'environnement de RV dont nous disposons. La première est une application de modélisation géométrique. L'utilisateur dispose d'un ensemble d'outils de bases permettant de modéliser des objets 3-D. Dans la seconde application, l'utilisateur modélise des scènes naturelles. Il peut créer des terrains à l'aide d'outils de modélisation classiques ou encore ajouter des détails naturels comme des arbres.

11.2 Perspectives

Par rapport au travail effectué, de nombreuses questions restent en suspens, notamment concernant la décomposition de la tâche.

11.2.1 Repère et construction mentale de la tâche

Que se soit pour la technique DIOD ou la métaphore du potier, la décomposition de la tâche s'effectue dans un repère cartésien dont les axes sont alignés sur le repère du monde virtuel. Il semblerait que cette décomposition soit appropriée et permette d'améliorer les performances des utilisateurs. Néanmoins, il est possible que des décompositions plus appropriées puissent être mises en place. C'est le cas, notamment, en choisissant un repère de travail différent.

Nous nous intéressons donc à la caractérisation du repère utilisé dans la conceptualisation de la tâche. La technique d'interaction peut, dans ce cas, jouer un rôle primordial. Par exemple, le repère dans lequel l'utilisateur conçoit la tâche de positionnement peut être différent, selon qu'il effectue le positionnement avec la manipulation directe ou avec la technique Go-Go [Poupyrev 96]. En effet, il est possible que l'utilisateur construise une représentation mentale de la tâche, dans un repère cartésien, avec la manipulation directe. Toutefois, en utilisant une technique comme le Go-Go, un système de coordonnées sphériques semble plus adapté à la représentation mentale de la tâche. Dans ce cas, le repère est centré sur l'utilisateur. La position de la main virtuelle est alors perçue, non pas dans un espace cartésien défini par trois coordonnées (x, y, z) , mais dans un espace défini par un repère sphérique. La position de l'objet est alors définie par trois coordonnées (ρ, θ, ϕ) . ρ est le rayon de la sphère centrée sur l'utilisateur, θ est la longitude et ϕ est la latitude.

Il peut être intéressant d'identifier les repères de manipulation associés aux techniques d'interaction afin de comprendre le comportement de l'utilisateur dans ses repères. Il est alors possible de proposer des méthodes de décomposition plus adaptées. Pour effectuer de telles études, il faut être capable de mesurer la coordination dans des repères tels que le système de coordonnées sphériques. Une première adaptation de la mesure NDC à un système de coordonnées sphériques est en cours d'étude.

11.2.2 Interaction hybride par l'utilisation d'un terminal mobile tactile

Nous avons pu constater que l'utilisation d'interfaces tactiles possède divers avantages par rapport à l'interaction 3-D (support physique et décomposition de la tâche).

La décomposition totale de la tâche n'est toutefois pas envisageable. Dans le cas de tâches à contrôle final, comme le positionnement, la décomposition introduit des temps de réalisation trop importants. C'est pourquoi, nous proposons une approche hybride, combinant interaction 3-D et interaction tactile, basée sur l'utilisation d'un terminal mobile tactile.

Les terminaux mobiles tactiles (par exemple iPhone™, HTC Touch™, ...) disponibles à ce jour sont suffisamment compacts pour pouvoir être utilisé par une seule main. La solution que nous proposons consiste à ajouter, à un terminal mobile tactile, des capteurs optiques permettant de suivre la position et l'orientation du périphérique dans l'espace de manipulation. Le terminal peut alors servir de périphérique d'interaction 3-D classique. La surface tactile reste à disposition afin de pouvoir décomposer la tâche en cours de réalisation. Avec cette approche, l'utilisateur peut utiliser le même périphérique afin de manipuler dans l'espace 3-D et de décomposer la tâche. Cette décomposition peut essentiellement servir à effectuer des ajustements selon des dimensions spécifiques. Cette approche permettrait de bénéficier du meilleur de l'interaction 3-D et du meilleur de l'interaction tactile dans les différentes phases de manipulation.

11.2.3 Information visuelle

Lors de l'évaluation de la technique DIOD, nous avons pu remarquer que la précision des utilisateurs semble fortement dépendre du retour visuel fourni durant la réalisation de la tâche. Plus particulièrement, il semblerait que l'insuffisance du retour visuel fourni au cours de la réalisation d'une tâche, puisse empêcher les utilisateurs de tirer pleinement profit de stratégies basées sur la décomposition. Notre travail sur l'amélioration du contrôle du geste d'interaction par la décomposition des DDL, devrait alors s'accompagner d'un travail sur la définition d'informations visuelles pertinentes facilitant la mise en place de telles stratégies.

Bibliographie

- [Accot 97] J. Accot & S. Zhai. *Beyond Fitts' law : models for trajectory-based HCI tasks*. In CHI '97 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 295–302, New York, NY, USA, 1997. ACM.
- [Accot 99] J. Accot & S. Zhai. *Performance evaluation of input devices in trajectory-based tasks : an application of the steering law*. In CHI '99 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 466–472, New York, NY, USA, 1999. ACM.
- [Bechmann 03] D. Bechmann & D. Gerber. *Arbitrary Shaped Deformations with DOGME*. The Visual Computer, vol. 19, no. 2-3, pages 175–186, may 2003.
- [Benjamini 95] Y. Benjamini & Y. Hochberg. *Controlling the False - Discovery Rate : A Practical and Powerful Approach to Multiple Testing*. Journal of the Royal Statistical Society, vol. 57, no. 1, pages 289 – 300, 1995.
- [Benko 07] H. Benko & S. Feiner. *Balloon Selection : A Multi-Finger Technique for Accurate Low-Fatigue 3D Selection*. In 3DUI, 2007 IEEE Symposium on 3D User Interfaces, 2007.
- [Borrel 91] P. Borrel & D. Bechmann. *Deformation of n-dimensional objects*. International Journal of Computational Geometry and Applications, vol. 1, no. 4, pages 427–453, 1991.
- [Botsch 04] M. Botsch & L. Kobbelt. *An intuitive framework for real-time free-form modeling*. ACM Trans. Graph., vol. 23, no. 3, pages 630–634, 2004.

- [Bowman 97] D. A. Bowman & L. F. Hodges. *An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments*. In SI3D '97 : Proceedings of the 1997 symposium on Interactive 3D graphics, pages 35–ff., New York, NY, USA, 1997. ACM.
- [Bowman 98] D. A. Bowman. *Interaction Techniques for Immersive Virtual Environments : Design*, 1998.
- [Bowman 99] D. A. Bowman. *Interaction Techniques for Common Tasks in Immersive Virtual Environments : Design, Evaluation, and Application*, 1999.
- [Butterworth 92] J. Butterworth, A. Davidson, S. Hench & M. T. Olano. *3DM : a three dimensional modeler using a head-mounted display*. In SI3D '92 : Proceedings of the 1992 symposium on Interactive 3D graphics, pages 135–138, New York, NY, USA, 1992. ACM.
- [Buxton 90] W. Buxton. *A three-state model of graphical input*. In INTERACT '90 : Proceedings of the IFIP TC13 Third International Conference on Human-Computer Interaction, pages 449–456, Amsterdam, The Netherlands, The Netherlands, 1990. North-Holland Publishing Co.
- [Casiez 04] G. Casiez. *Contribution à l'étude des interfaces haptiques. Le DigHaptic : un périphérique haptique de bureau à degrés de liberté séparés*. PhD thesis, Université des Sciences et Technologies de Lille, 2004.
- [Casiez 08] G. Casiez, D. Vogel, R. Balakrishnan & A. Cockburn. *The Impact of Control-Display Gain on User Performance in Pointing Tasks*. Human-Computer Interaction, vol. 23, no. 3, pages 215 – 250, 2008.
- [Chang 09] S. H. Chang, L. Stuart, B. Plimmer & B. Wünsche. *Origami simulator : a multi-touch experience*. In CHI EA '09 : Proceedings of the 27th international conference extended abstracts on Human factors in computing systems, pages 3889–3894, New York, NY, USA, 2009. ACM.
- [Chen 88] M. Chen, S. J. Mountford & A. Sellen. *A study in interactive 3-D rotation using 2-D control devices*. SIGGRAPH Comput. Graph., vol. 22, no. 4, pages 121–129, 1988.
- [Cohen 99] J. M. Cohen, L. Markosian, R. C. Zeleznik, J. F. Hughes & R. Barzel. *An interface for sketching 3D curves*. In I3D '99 : Proceedings of the 1999 symposium on Interactive 3D graphics, pages 17–21, New York, NY, USA, 1999. ACM.
- [De Amicis 01] R. De Amicis, M. Fiorentino & A. Stork. *Parametric Interaction for CAD Application in Virtual Reality Environment*. In International

- Conference on Design Tools and Methods in Industrial Engineering, pages 43–52, 2001.
- [Deisinger 01] J. Deisinger, R. Blach, G. Wesche, R. Breining & A. Simon. *Towards Immersive Modeling - Challenges and Recommendations : A Workshop Analyzing the Needs of Designers*. In In Proc. of the 6th Eurographics Workshop on Virtual Environments, pages 145–156, 2001.
- [Fiorentino 03] M. Fiorentino, P. A. Renzulli, A. Uva & P. Di Bari. *3D Pointing in Virtual Reality : Experimental study*, *International Conference on Tools And Methods Evolution In Engineering Design*. In XIII ADM - XV INGEGRAP International Conference on TOOLS AND METHODS EVOLUTION IN ENGINEERING DESIGN, pages 4–6, 2003.
- [Fitts 53] P.M. Fitts & C.M. Seeger. *S-R compatibility : spatial characteristics of stimulus and response codes*. *Journal of Experimental Psychology*, vol. 46, no. 3, pages 199–210, Sep 1953.
- [Fitts 54] P.M. Fitts. *The information capacity of the human motor system in controlling the amplitude of movement*. *Journal of Experimental Psychology*, vol. 47, pages 381–391, 1954.
- [Foskey 05] M. Foskey, M. A. Otaduy & M. C. Lin. *ArtNova : touch-enabled 3D model design*. In SIGGRAPH '05 : ACM SIGGRAPH 2005 Courses, page 188, New York, NY, USA, 2005. ACM.
- [Frees 05] S. Frees & G. D. Kessler. *Precise and Rapid Interaction through Scaled Manipulation in Immersive Virtual Environments*. In VR '05 : Proceedings of the 2005 IEEE Conference 2005 on Virtual Reality, pages 99–106, Washington, DC, USA, 2005. IEEE Computer Society.
- [Fuchs 06] P. Fuchs, G. Moreau, A. Berthoz & J-L. Vercher. *Le traité de la réalité virtuelle. volume 1 : L'homme et l'environnement virtuel*, volume 1 of *L'homme et l'environnement virtuel*. Mines Paris, third edition, 2006.
- [Gain 08] J. Gain & D. Bechmann. *A survey of spatial deformation from a user-centered perspective*. *ACM Transactions On Graphics (TOG)*, vol. 27, no. 4, october 2008.
- [Gregory 00] A. D. Gregory, S. A. Ehmann & M. C. Lin. *inTouch : Interactive Multiresolution Modeling and 3D Painting with a Haptic Interface*. In VR '00 : Proceedings of the IEEE Virtual Reality 2000 Conference, page 45, Washington, DC, USA, 2000. IEEE Computer Society.

- [Grossman 01] T. Grossman, R. Balakrishnan, G. Kurtenbach, G. Fitzmaurice, A. Khan & B. Buxton. *Interaction techniques for 3D modeling on large displays*. In I3D '01 : Proceedings of the 2001 symposium on Interactive 3D graphics, pages 17–23, New York, NY, USA, 2001. ACM.
- [Grossman 02] T. Grossman, R. Balakrishnan & K. Singh. *An Interface for Creating and Manipulating Curves Using a High Degree-of-Freedom Curve Input Device*. In CHI 2002 Conference Proceedings : ACM Conference on Human Factors in Computing Systems, pages 121–128, 2002.
- [Grossman 04] T. Grossman & R. Balakrishnan. *Pointing at trivariate targets in 3D environments*. In CHI '04 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 447–454, New York, NY, USA, 2004. ACM.
- [Guiard 87] Y. Guiard. *Asymmetric division of labor in human skilled bimanual action : the kinematic chain as a model*. In Journal of motor behavior, pages 486–517, 1987.
- [Hachet 02] M. Hachet & P. Guitton. *The interaction table : a new input device designed for interaction in immersive large display environments*. In EGVE '02 : Proceedings of the workshop on Virtual environments 2002, pages 189–196, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
- [Hachet 04] M. Hachet, P. Guitton, P. Reuter & F. Tyndiuk. *The CAT for efficient 2D and 3D interaction as an alternative to mouse adaptations*. ACM Trans. Graph., vol. 23, no. 3, pages 731–731, 2004.
- [Hancock 07] M. Hancock, S. Carpendale & A. Cockburn. *Shallow-depth 3d interaction : design and evaluation of one-, two- and three-touch techniques*. In CHI '07 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 1147–1156, New York, NY, USA, 2007. ACM.
- [Hanson 97] A. J. Hanson & E. A. Wernert. *Constrained 3D navigation with 2D controllers*. In VIS '97 : Proceedings of the 8th conference on Visualization '97, pages 175–ff., Los Alamitos, CA, USA, 1997. IEEE Computer Society Press.
- [Hart 88] S. G. Hart & L. E. Stavenland. *Development of NASA-TLX (Task Load Index) : Results of empirical and theoretical research*. In P. A. Hancock & N. Meshkati, editors, Human Mental Workload, chapitre 7, pages 139–183. Elsevier, 1988.

- [Hinckley 94] K. Hinckley, R. Pausch, J. C. Goble & N. F. Kassell. *A Survey of Design Issues in Spatial Input*. In Proc. ACM UIST'94 Symposium on User Interface Software & Technology, pages 213–222, 1994.
- [Hinckley 97] K. Hinckley, J. Tullio, R. Pausch, D. Proffitt & N. Kassell. *Usability analysis of 3D rotation techniques*. In UIST '97 : Proceedings of the 10th annual ACM symposium on User interface software and technology, pages 1–10, New York, NY, USA, 1997. ACM.
- [Hiraki 98] N. Hiraki, K. Kiyokawa, H. Takemura & N. Yokoya. *Imposing geometric constraints on virtual 3D objects within an immersive modeler*. In B. Gopalakrishnan & S. Murugesan, editeur, Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, volume 3517 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 215–226, October 1998.
- [Hormann 08] Kai Hormann, Konrad Polthier & Alia Sheffer. *Mesh parameterization : theory and practice*. In SIGGRAPH Asia '08 : ACM SIGGRAPH ASIA 2008 courses, pages 1–87, New York, NY, USA, 2008. ACM.
- [Interrante 06] V. Interrante, B. Ries & L. Anderson. *Distance Perception in Immersive Virtual Environments, Revisited*. vr, vol. 00, pages 3–10, 2006.
- [Interrante 07] V. Interrante, J. K. Kearney, D. Proffitt, J.E. Swan & W.B. Thompson. *Elucidating Factors that can Facilitate Veridical Spatial Perception in Immersive Virtual Environments*. In Virtual Reality Conference, 2007. VR '07. IEEE, pages 11–18, Charlotte, NC, 2007.
- [Jacob 92] R. J. K. Jacob & L. E. Sibert. *The perceptual structure of multidimensional input device selection*. In CHI '92 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 211–218, New York, NY, USA, 1992. ACM.
- [Jacob 94] R. J. K. Jacob, L. E. Sibert, D. C. McFarlane & M. P. Mullen. *Integrality and separability of input devices*. ACM Trans. Comput.-Hum. Interact., vol. 1, no. 1, pages 3–26, 1994.
- [John 85] B. E. John, P. S. Rosenbloom & A. Newell. *A theory of stimulus-response compatibility applied to human-computer interaction*. In CHI '85 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 213–219, New York, NY, USA, 1985. ACM.
- [John 96] B. E. John & D. E. Kieras. *Using GOMS for user interface design and evaluation : which technique ?* ACM Trans. Comput.-Hum. Interact., vol. 3, no. 4, pages 287–319, 1996.

- [Kameyama 97] K. Kameyama. *Virtual clay modeling system*. In VRST '97 : Proceedings of the ACM symposium on Virtual reality software and technology, pages 197–200, New York, NY, USA, 1997. ACM.
- [Keefe 05] D. Keefe & D. Laidlaw. *A Haptic Interface for Creating Smooth 3D Curves with Varying Line Weight*. In Proceedings SIGGRAPH, Posters Program, 2005.
- [Keefe 07] D. Keefe, R. Zeleznik & D. Laidlaw. *Drawing on Air : Input Techniques for Controlled 3D Line Illustration*. Transactions on Visualization and Computer Graphics, vol. 5, pages 1067–1081, 2007.
- [Keijser 07] J. Keijser, S. Carpendale, M. Hancock & T. Isenberg. *Exploring 3D Interaction in Alternate Control-Display Space Mappings*. In Wolfgang Sturzlinger, Yoshifumi Kitamura & Sabine Coquillart, editors, Proceedings of the 2nd IEEE Symposium on 3D User Interfaces (3DUI 2007, March 10–11, 2007, Charlotte, North Carolina, USA), pages 17–24, Los Alamitos, CA, 2007. IEEE Computer Society.
- [Kieras 97] D. E. Kieras, S. D. Wood & D. E. Meyer. *Predictive engineering models based on the EPIC architecture for a multimodal high-performance human-computer interaction task*. ACM Trans. Comput.-Hum. Interact., vol. 4, no. 3, pages 230–275, 1997.
- [Kitamura 99] Y. Kitamura, T. Higashi, T. Masaki & F. Kishino. *Virtual Chopsticks : Object Manipulation using Multiple Exact Interactions*. In VR '99 : Proceedings of the IEEE Virtual Reality, page 198, Washington, DC, USA, 1999. IEEE Computer Society.
- [Kiyokawa 97] K. Kiyokawa, H. Takemura, Y. Katayama, H. Iwasa & N. Yokoya. *VLEGO : A Simple Two-handed Modeling Environment Based On Toy Block*, 1997.
- [Kruger 05] R. Kruger, S. Carpendale, S. D. Scott & A. Tang. *Fluid integration of rotation and translation*. In CHI '05 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 601–610, New York, NY, USA, 2005. ACM.
- [LaViola 08] J.J. LaViola, A.S. Forsberg, J. Huffman & A. Bragdon. *Poster : Effects of Head Tracking and Stereo on Non-Isomorphic 3D Rotation*. In 2008 IEEE Symposium on 3D User Interfaces, pages 155–156, March 2008.
- [Lécuyer 04] A. Lécuyer, J. Burkhardt & L. Etienne. *Feeling bumps and holes without a haptic interface : the perception of pseudo-haptic textures*. In CHI '04 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 239–246, New York, NY, USA, 2004. ACM.

- [Legault 07] I. Legault, R. Allard & J. Faubert. *Normal Aging and the Perception of Curvature Shapes*. Optometry and Vision Science, vol. 84, no. 12, pages 1087–1092, December 2007.
- [Liang 94] J. Liang & M. Green. *JDCAD : a highly interactive 3D modeling system*. Computer & Graphics, vol. 18, no. 4, pages 499–506, 1994.
- [Likert 32] R. Likert. *A technique for the measurement of attitudes*. Archives of Psychology, vol. 22, no. 140, pages 1–55, 1932.
- [Liu 10] L. Liu, J.B. Martens & R. van Liere. *Revisiting Path Steering for 3D Manipulation Tasks*. In 3DUI 2010 : Symposium for 3D User Interfaces, 2010.
- [MacKenzie 94] S. MacKenzie & S. Riddersma. *Effects of output display and control-display gain on human performance in interactive systems*. Behaviour and Information Technologie, vol. 1, no. 3, pages 328–337, 1994.
- [MacKenzie 01] I. A. MacKenzie, T. Kauppinen & M. Silfverberg. *Accuracy measures for evaluating computer pointing devices*. In CHI '01 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 9–16, New York, NY, USA, 2001. ACM.
- [MacKenzie 03] I. S. MacKenzie. *Motor behaviour models for human-computer interaction*, 2003.
- [Martinet 09] Anthony Martinet, Géry Casiez & Laurent Grisoni. *3D positioning techniques for multi-touch displays*. In VRST '09 : Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology, pages 227–228, New York, NY, USA, 2009. ACM.
- [Martinet 10] Anthony Martinet, Géry Casiez & Laurent Grisoni. *The Design and Evaluation of 3D Positioning Techniques for Multi-touch Displays*. In 3DUI 2010 : Proceedings of the IEEE Symposium on 3D User Interfaces, pages 115–118, New York, NY, USA, 2010. ACM.
- [Masliah 00] M. R. Masliah & P. Milgram. *Measuring the allocation of control in a 6 degree-of-freedom docking experiment*. In CHI '00 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 25–32, New York, NY, USA, 2000. ACM.
- [Meyer 88] D. E. Meyer, R. A. Abrams, S. Kornblum, C. E. Wright & J. E. K. Smith. *Optimality in human motor performance : ideal control of rapid aimed movements*. Psychological review, vol. 95, no. 3, pages 340–370, 1988.
- [Mine 97] M. R. Mine, Jr. Brooks, P. Frederick & C. H. Sequin. *Moving objects in space : exploiting proprioception in virtual-environment interaction*. In SIGGRAPH '97 : Proceedings of the 24th annual conference

- on Computer graphics and interactive techniques, pages 19–26, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [Moritz 00] E. Moritz, F. Kuester, B. Hamann, K. I. Joy & H. Hagen. *Towards Immersive Clay Modelling : Interactive Modelling with Octrees*. In Proceedings of Stereoscopic Displays and Virtual Reality Systems VII, volume 3957, pages 414–422, 2000.
- [Nacenta 09] M. A. Nacenta, P. Baudisch, H. Benko & A. Wilson. *Separability of spatial manipulations in multi-touch interfaces*. In GI '09 : Proceedings of Graphics Interface 2009, pages 175–182, Toronto, Ont., Canada, Canada, 2009. Canadian Information Processing Society.
- [Osawa 07] N. Osawa & X. Ren. *A Study on Approximate and Fine Adjustments by Hand Motion in an Immersive Environment*. Transactions of Information Processing Society of Japan, vol. 48, no. 11, pages 3568–3576, November 2007.
- [Parsons 95] L.M. Parsons. *Inability To Reason About An Object's Orientation Using An Axis And Angle Of Rotation*. Journal of experimental psychology : Human perception and performance, vol. 21, pages 1259 – 1277, 1995.
- [Pavlovych 09] A. Pavlovych & W. Stuerzlinger. *The tradeoff between spatial jitter and latency in pointing tasks*. In EICS '09 : Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems, pages 187–196, New York, NY, USA, 2009. ACM.
- [Peytavie 09] A. Peytavie, E. Galin, S. Merillou & J. Grosjean. *Arches : a Framework for Modeling Complex Terrains*. Computer Graphics Forum (Proceedings of Eurographics), vol. 28, no. 2, pages 457–467, 2009.
- [Pierce 99] J. S. Pierce, B. C. Stearns & R. Pausch. *Voodoo dolls : seamless interaction at multiple scales in virtual environments*. In I3D '99 : Proceedings of the 1999 symposium on Interactive 3D graphics, pages 141–145, New York, NY, USA, 1999. ACM.
- [Poupyrev 96] I. Poupyrev, M. Billinghurst, S. Weghorst & T. Ichikawa. *The go-go interaction technique : non-linear mapping for direct manipulation in VR*. In UIST '96 : Proceedings of the 9th annual ACM symposium on User interface software and technology, pages 79–80, New York, NY, USA, 1996. ACM.
- [Poupyrev 99] I. Poupyrev, S. Weghorst, T. Otsuka & T. Ichikawa. *Amplifying spatial rotations in 3D interfaces*. In CHI '99 : CHI '99 extended abstracts on Human factors in computing systems, pages 256–257, New York, NY, USA, 1999. ACM.

- [Poupyrev 00] I. Poupyrev, S. Weghorst & S. Fels. *Non-isomorphic 3D rotational techniques*. In CHI '00 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 540–547, New York, NY, USA, 2000. ACM.
- [Ryu 05] J. Ryu, N. Hashimoto & M. Sato. *Influence of Resolution Degradation on Distance Estimation in Virtual Space Displaying Static and Dynamic Image*. In CW '05 : Proceedings of the 2005 International Conference on Cyberworlds, pages 43–50, Washington, DC, USA, 2005. IEEE Computer Society.
- [Sachs 91] E. Sachs, A. Roberts & D. Stoops. *3-Draw : A Tool for Designing 3D Shapes*. IEEE Comput. Graph. Appl., vol. 11, no. 6, pages 18–26, 1991.
- [Schkolne 01] S. Schkolne, M. Pruett & P. Schröder. *Surface drawing : creating organic 3D shapes with the hand and tangible tools*. In CHI '01 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 261–268, New York, NY, USA, 2001. ACM.
- [Schmandt 83] C. Schmandt. *Spatial input/display correspondence in a stereoscopic computer graphic work station*. SIGGRAPH Comput. Graph., vol. 17, no. 3, pages 253–261, 1983.
- [Shaw 94] C. Shaw & M. Green. *Two-handed polygonal surface design*. In UIST '94 : Proceedings of the 7th annual ACM symposium on User interface software and technology, pages 205–212, New York, NY, USA, 1994. ACM.
- [Sheffer 06] A. Sheffer, E. Praun & K. Rose. *Mesh parameterization methods and their applications*. Found. Trends. Comput. Graph. Vis., vol. 2, no. 2, pages 105–171, 2006.
- [Shneiderman 87] B. Shneiderman. *Direct manipulation : A step beyond programming languages*. Human-computer interaction : a multidisciplinary approach, pages 461–467, 1987.
- [Shneiderman 97] B. Shneiderman. *Direct manipulation for comprehensible, predictable and controllable user interfaces*. In IUI '97 : Proceedings of the 2nd international conference on Intelligent user interfaces, pages 33–39, New York, NY, USA, 1997. ACM.
- [Stoakley 95] R. Stoakley, M. J. Conway & R. Pausch. *Virtual reality on a WIM : interactive worlds in miniature*. In CHI '95 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 265–272, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

- [Stork 97] A. Stork & M. Maidhof. *Efficient and precise solid modelling using a 3D input device*. In SMA '97 : Proceedings of the fourth ACM symposium on Solid modeling and applications, pages 181–194, New York, NY, USA, 1997. ACM.
- [Sturman 89] D. J. Sturman, D. Zeltzer & S. Pieper. *Hands-on interaction with virtual environments*. In UIST '89 : Proceedings of the 2nd annual ACM SIGGRAPH symposium on User interface software and technology, pages 19–24, New York, NY, USA, 1989. ACM.
- [Ueda 05] E. Ueda, Y. Matsumoto & T. Ogasawara. *Virtual Clay Modeling System Using Multi-Viewpoint Images*. In 3DIM '05 : Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling, pages 134–141, Washington, DC, USA, 2005. IEEE Computer Society.
- [Wang 98] Y. Wang, C. MacKenzie, V. Summers & K. Booth. *The structure of object transportation and orientation in human-computer interaction*. In CHI '98 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 312–319, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co.
- [Wanger 92] L. C. Wanger, J. A. Ferwerda & D. P. Greenberg. *Perceiving Spatial Relationships in Computer-Generated Images*. IEEE Comput. Graph. Appl., vol. 12, no. 3, pages 44–51, 54–58, 1992.
- [Watson 97] R. W. Watson, R. D. Jones & N. B. Sharman. *Two-dimensional tracking tasks for quantification of sensory-motor dysfunction and their application to Parkinson's disease*. Medical and Biological Engineering and Computing, vol. 35, no. 2, pages 141–145, March 1997.
- [Wesche 00] G. Wesche & M. Droske. *Conceptual free-form styling on the responsive workbench*. In VRST '00 : Proceedings of the ACM symposium on Virtual reality software and technology, pages 83–91, New York, NY, USA, 2000. ACM Press.
- [Wesche 01] G. Wesche & H. P. Seidel. *FreeDrawer : a free-form sketching system on the responsive workbench*. In VRST '01 : Proceedings of the ACM symposium on Virtual reality software and technology, pages 167–174, New York, NY, USA, 2001. ACM Press.
- [Wilkes 08] C. Wilkes & D. A. Bowman. *Advantages of velocity-based scaling for distant 3D manipulation*. In VRST '08 : Proceedings of the 2008 ACM symposium on Virtual reality software and technology, pages 23–29, New York, NY, USA, 2008. ACM.

- [Worringham 98] C. J. Worringham & D. B. Beringer. *Directional stimulus-response compatibility : a test of three alternative principles*. Ergonomics, vol. 41, no. 6, pages 864–880, 1998.
- [Zhai 98] S. Zhai & P. Milgram. *Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices*. In CHI '98 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 320–327, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co.

Implémentation de CrOS

Dans le cadre de l'algorithme proposé pour la fonction de placage, nous ne faisons aucune supposition particulière sur la structure de données utilisée pour représenter les objets 3-D. Pour des questions d'optimalité, il est toutefois conseillé d'utiliser une structure de données fournissant des informations sur la topologie de l'objet. Plus particulièrement, il est recommandé de pouvoir accéder facilement, pour une face donnée, à l'ensemble des faces adjacentes.

Pour décrire formellement l'algorithme, nous utilisons les entités géométriques suivantes :

- P_c (plan de contrôle) : le plan dans lequel l'utilisateur effectue les mouvements 2-D. Dans notre cas, la surface d'interaction étant la surface définie par l'écran du Workbench se trouvant à portée de main, P_c est défini comme le plan contenant la surface de l'écran ;
- $\vec{N}_c$: la normale associée au plan P_c . Dans notre cas, $\vec{N}_c = (0, 1, 0)$;
- P_g (plan géodésique) : le plan qui est utilisé pour calculer le déplacement sur la surface de l'objet ;
- $\vec{N}_g$: la normale associée au plan P_g ;
- $\vec{D}$: le déplacement 2-D effectué par l'utilisateur dans le repère du plan de contrôle. En pratique, le déplacement sera un déplacement 3-D dont la coordonnée y sera considérée comme nulle ;
- I : l'intersection entre P_g et l'objet 3-D. L'intersection est stockée sous forme d'une ligne polygonale fermée ou ouverte contenant la liste des points d'intersection entre les arêtes et sommets du maillage et le plan.

Nous présentons maintenant la manière dont P_g doit être défini.

A.1 Définition du plan géodésique

Les contraintes permettant de définir le plan P_g sont les suivantes :

- Il doit passer par la position du curseur ;
- Il doit contenir le déplacement 2-D spécifié par l'utilisateur ;
- La normale du plan doit être contenue dans le plan P_c .

En partant de ces caractéristiques, il est possible de définir géométriquement le plan P_g . $\vec{N}_g$, la normale de P_g peut être définie de la manière suivante :

$$\vec{N}_g = \vec{D} \times \vec{N}_c$$

La normale définit trois des quatre inconnues de l'équation du plan P_g . La quatrième inconnue peut être calculée en utilisant le fait que le curseur se trouve dans le plan P_g . À ce stade, nous sommes donc capables de définir le plan P_g , qui va permettre le calcul de l'intersection. Cette intersection nous donne le chemin géodésique que le curseur va suivre.

A.2 Calcul de l'intersection

L'approche que nous proposons pour le calcul de l'intersection est **incrémentale**. Une approche naïve consisterait à calculer l'ensemble de l'intersection entre P_g et l'objet pour ensuite sélectionner la partie de l'intersection correspondant au chemin géodésique. Ce calcul, avec une structure de donnée adaptée, aurait un coût en $O(n)$ où n est le nombre d'arêtes du maillage. Nous proposons une approche où l'intersection est construite de proche en proche. Le calcul incrémental permet de réduire les temps de calcul à $O(m)$ où m est le nombre de sommets ou d'arêtes intersectés appartenant au chemin géodésique. Ce gain est appréciable notamment sur des objets volumineux. Le détail de l'algorithme est donné à la fin de cette section.

Une difficulté apparaît alors dans le calcul de l'intersection, et donc, du chemin géodésique. Considérons que le curseur se trouve sur une face F . Considérons également que le chemin géodésique est défini comme une suite de sommets formant une ligne polygonale, le premier sommet du chemin géodésique étant le curseur lui-même. L'intersection entre F et P_g est une ligne polygonale dont les sommets se trouvent sur des arêtes de F ou sont des sommets de F . La question qui se pose alors est la suivante : quel est le point qui appartient au chemin géodésique ? Une illustration est donnée en Figure A.1. En choisissant le mauvais point, nous introduisons une inversion de la relation directionnelle entre le mouvement du curseur et le mouvement de l'utilisateur. Il faut donc trouver un moyen de faire le choix approprié et de lever l'ambiguïté.

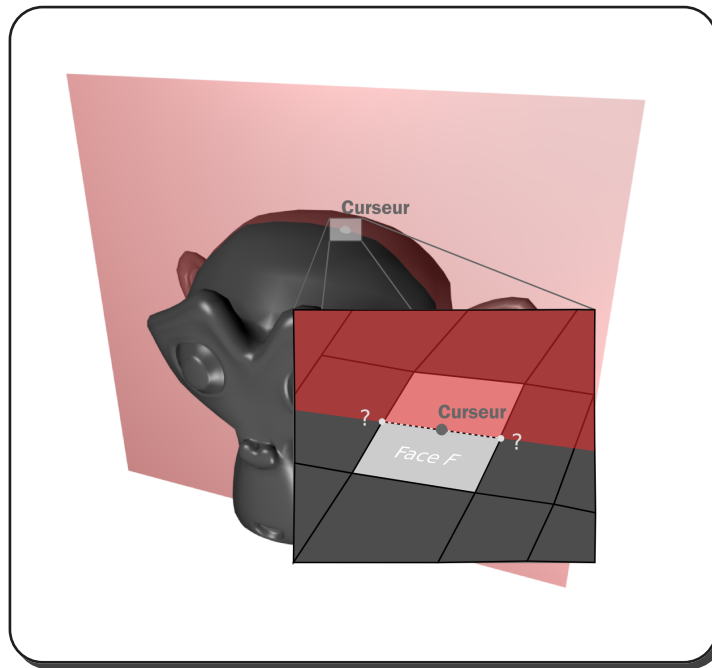


FIGURE A.1 – L'intersection entre le plan et la face F est un ensemble constitué d'au moins deux points. L'un des ces deux points se trouve sur le chemin géodésique, lequel ?

Afin de déterminer le point à choisir, considérons $\vec{V}_g$, le vecteur défini par :

$$\vec{V}_g = \text{point testé} - \text{position du curseur}$$

Il faut alors choisir le point tel que :

$$\vec{V}_g \cdot \vec{D} > 0 \quad (\text{A.1})$$

Ce test permet de savoir si le point va induire un déplacement dont la projection sur le plan P_c est dans la même direction que $\vec{D}$. Si la condition est vérifiée, le point testé se trouve bien dans le chemin géodésique. Cette condition permet de garantir une compatibilité directionnelle.

À présent, nous avons trouvé le premier sommet à ajouter au chemin géodésique. Une fois ce premier élément ajouté, la construction du reste du chemin peut se faire de manière incrémentale et sans ambiguïté. Pour cela nous utilisons l'algorithme suivant :

```
double geodesicPathLength = length(D)
Point firstIntersection = the first intersection
Edge intersectedEdge = edge on which is firstIntersection
```

```

Face currentFace = adjacent face to F and incident to intersectedEdge
list<Point> geodesicPath = {cursorPosition, firstIntersection}
double currentGeodesicPathLength = length(cursorPosition - firstIntersection)

while( currentGeodesicPathLength < geodesicPathLength ) {
    Edge currentIntersectedEdge = edge intersected by $P_g$ different of intersectedEdge
    Point currentIntersection = point of intersection between $P_g$ and currentIntersectedEdge
    currentGeodesicPathLength += length(currentIntersection - last element of the geodesicPath)
    add the intersection with currentIntersectedEdge to geodesicPath
    intersectedEdge = currentIntersectedEdge
    currentFace = adjacent face to currentFace and incident to currentIntersectedEdge
}

return geodesicPath;

```

Enfin, la position du curseur après déplacement sur le chemin géodésique est définie comme la position sur le chemin géodésique telle que la distance géodésique entre la position précédente et la position courante du curseur est égale à la longueur du déplacement effectué par l'utilisateur dans le plan de contrôle.

Questionnaire d'évaluation du suivi de trajectoires

Questionnaire d'évaluation subjective

Question 01 : La technique est intuitive.

MD:							
TOUCH:							
TOUCH+PRISM:							



Pas d'accord *D'accord*

Remarques :

.....

.....

Question 02 : La technique permet un contrôle adapté pour la tâche de suivi de trajectoires.

MD:							
TOUCH:							
TOUCH+PRISM:							



Pas d'accord *D'accord*

Remarques :

.....

.....

Question 03 : La technique permet de suivre précisément les trajectoires.

MD:							
TOUCH:							
TOUCH+PRISM:							



Pas d'accord *D'accord*

Remarques :

.....

.....

Question 04 : L'utilisation de cette technique introduit une fatigue importante.

MD:							
TOUCH:							
TOUCH+PRISM:							



Pas d'accord *D'accord*

Remarques :

.....

.....

Inter-classement / Préférence

Question 05 : J'ai préféré cette technique aux autres techniques.

MD:							
TOUCH:							
TOUCH+PRISM:							



Pas d'accord
D'accord

Remarques :

.....

.....

Questionnaire d'évaluation de DIOD

Questionnaire d'évaluation subjective

Question 01 : La technique permet de réaliser facilement des déplacements de grandes amplitudes.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 02 : La technique permet de réaliser facilement des positionnements précis.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Intuitivité

Question 03 : L'utilisation de la technique est intuitive.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 04 : L'objet déplacé réagissait conformément à mes prévisions.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

Apprentissage

Question 05 : On parvient facilement à comprendre le comportement de la technique.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 06 : On parvient facilement à maîtriser le comportement de la technique.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 07 : On parvient facilement à exploiter les caractéristiques de la technique pour réaliser la tâche.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Adéquation à la tâche

Question 08 : La technique est adaptée pour la tâche demandée.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

Question 09 : En phase finale, le comportement me paraît adapté.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 10 : En phase d'approche, le comportement de la technique me paraît adapté.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Inter-classement / Préférence

Question 11 : Pour d'autres tâches de positionnement de précision, je réutiliserais cette technique.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 12 : Pour d'autres tâches de positionnement grossier, je réutiliserais cette technique.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 13 : En général, pour des tâches de positionnement, je réutiliserais cette technique.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....

Question 14 : J'ai préféré cette technique aux autres techniques.

MD :							
PRISM :							
DIOD :							



Pas d'accord

D'accord

Remarques :

.....

.....