



"Haptic Processor Unit" : vers une Plate-Forme Transportable pour la Simulation Temps-Réel Synchrone Multisensorielle

Damien Couroussé

► To cite this version:

Damien Couroussé. "Haptic Processor Unit" : vers une Plate-Forme Transportable pour la Simulation Temps-Réel Synchrone Multisensorielle. Informatique [cs]. Institut National Polytechnique de Grenoble, 2008. Français. <NNT : >. <tel-01245110>

HAL Id: tel-01245110

<https://theses.hal.science/tel-01245110v1>

Submitted on 16 Dec 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque

--	--	--	--	--	--	--	--	--	--

THÈSE

pour obtenir le grade de

DOCTEUR de l'Institut National Polytechnique de Grenoble

Spécialité : **Ingénierie de la Cognition, de l'Interaction, de l'Apprentissage et de la Création**

préparée au **Laboratoire ICA et ACROE**

dans le cadre de l'École Doctorale **E.D.I.S.C.E**

Ecole Doctorale Ingénierie pour la Santé, la Cognition et l'Environnement

présentée et soutenue publiquement

par

Damien COUROUSSÉ

le 5 Novembre 2008

Titre:

**« HAPTIC PROCESSOR UNIT » : VERS UNE PLATE-FORME TRANSPORTABLE
POUR LA SIMULATION TEMPS-RÉEL SYNCHRONE MULTISENSORIELLE**

Directeur de thèse : **Claude CADOZ**

Co-encadrante : **Annie LUCIANI**

Jury

M ^{me} Indira THOUVENIN,	Rapporteur
M Antoine FERREIRA,	Rapporteur
M Pierre BENECH,	Examineur
M Christophe CHAILLOU,	Examineur
M Claude CADOZ,	Directeur de thèse
M ^{me} Annie LUCIANI,	Co-encadrante

ACROE (Association pour la Création et la Recherche sur les Outils d'Expression)

Laboratoire ICA (Informatique et Création Artistique)

46, av. Félix Viallet, 38031 Grenoble Cedex, FRANCE

Remerciements

Je remercie Indira THOUVENIN, maître de conférences à l'Université Technologique de Compiègne, Antoine FERREIRA, professeur à l'ENSI de Bourges, Pierre BENECH, professeur à l'INP Grenoble, et Christophe CHAILLOU, professeur à l'Université de Lille, qui m'ont fait le plaisir et l'honneur de faire partie de mon jury de thèse.

J'adresse en grand merci à Claude CADOZ, directeur de l'ACROE, pour avoir assumé la responsabilité de cette thèse, et à Annie LUCIANI, directrice de l'ICA, pour m'avoir accueilli au sein dès 2002 pour un stage de fin d'études d'ingénieur, puis pour avoir accepté de continuer l'aventure jusqu'à présent. Merci Annie également d'avoir constamment affirmé un soutien et une confiance sans réserve dans les travaux que nous avons menés ensemble.

Je tiens à remercier tout particulièrement

Jean-Loup FLORENS, ingénieur de recherche à l'ACROE et responsable de l'équipe *temps réel*, dont l'engagement a été déterminant dans la réalisation de cette thèse. Au-delà du chercheur, je remercie également la disponibilité et la générosité d'un collègue de travail.

Gunnar JANSON, professeur émérite de l'université de Uppsala en Suède, pour son investissement dans les travaux que nous avons menés sur la perception de l'élasticité, et en particulier pour son accueil chaleureux à Uppsala.

Les membres du laboratoire qui ont contribué à l'achèvement de ce travail : Ali ALLAOUI, Geoffrey CERNA, Nicolas CASTAGNÉ et Sylvain MARLIÈRE pour leurs contributions importantes aux réalisations logicielles, Daniela URMA dans le cadre du réseau d'excellence Enactive, les stagiaires Arnaud DRAZEK, Rémi CAILLETAUD, Yannis HECTOR, Maxime HOUOT et Daniela FAVARETTO.

Annick MONTANVERT, Cédric GÉROT et tous mes collègues du département Informatique de l'IUT2 de Grenoble, pour leur confiance, le plaisir que j'ai eu à travailler avec eux durant ces années et leur soutien dans les moments difficiles.

La bonne humeur et le soutien humain de toutes les personnes du laboratoire qui n'ont pas encore été citées plus haut : Aurélie, Olivier, Chimin, Maria, François, Alexandros, Julien, Kévin, Diana, Matthieu...

Ma famille, mes amis et Delphine pour leur présence à mes côtés.

Le travail présenté dans cette thèse a été financé par le réseau d'excellence européen Enactive Interfaces IST-2004-002114-ENACTIVE. Il a été soutenu par le Ministère de la Culture et de la Communication, le Ministère de l'Enseignement Supérieur et de la Recherche, et le Conseil Régional de la Région Rhône-Alpes.

L'aboutissement de ce travail a également été rendu possible grâce à un poste d'Assistant Temporaire à l'Enseignement et à la Recherche au département Informatique de l'IUT2 à l'Université Pierre Mendès France de Grenoble.

The work presented in this thesis was funded by the network of excellence Enactive Interfaces IST-2004-002114-ENACTIVE. It was supported by the french Ministry of Culture and Communication, the french Ministry of Higher Education and Research, and by the Rhône-Alpes Regional Council.

The successful outcome of this work was also made financially possible thanks to a temporary position of Assistant Professor at the department of Computer Science at the IUT2 of University Pierre Mendès France, Grenoble.

à Delphine,

Résumé

Ce travail se situe dans le domaine de l'Interaction Personne-Système, et plus particulièrement dans celui de la simulation instrumentale multisensorielle telle que conceptualisée par le groupe de recherche ACROE & ICA, qui nécessite un couplage fort homme-instrument.

La première partie de cette thèse présente les différents degrés d'intégration du geste dans l'ordinateur, puis propose une approche fonctionnelle des technologies pour le retour d'effort. Nous dégagons de cette analyse les grandes approches qui se partagent actuellement le champ de la recherche « haptique ».

Nous présentons ensuite une étude sur les différents composants matériels et logiciels nécessaires à la chaîne de simulation haptique, ainsi que les différentes approches utilisées pour connecter un système à retour d'effort à un processus de simulation en temps réel. L'analyse du rôle des composants de la chaîne de simulation et de leurs relations permet de formaliser le concept de « *Haptic Processor Unit* ». Ce composant permet en particulier de garantir les conditions de réactivité propres à la simulation multisensorielle. La nouvelle architecture de simulation multisensorielle que nous avons réalisée, ERGON_X, met en œuvre le concept de HPU. ERGON_X est un simulateur compact et transportable, et permet d'utiliser des fréquences de simulation jusqu'à 44 100 Hz.

La dernière partie présente la validation de la plate-forme de simulation ERGON_X. Elle est essentiellement orientée vers l'implantation de nouveaux modèles, utilisés dans le cadre d'un travail de recherche sur la situation instrumentale médiatisée. Le « E » est un modèle de démonstration des performances de la technologie ERGOS que la nouvelle architecture de simulation permet d'exploiter pleinement. Les modèles de tapping (percussion) et de pâtes déformables ont permis d'avancer des résultats sur l'interaction homme-objet, et valident le simulateur comme un outil pour l'expérimentation psychophysique. Les *Enactive Emblematic Scenarii* « *Ergotic Sounds* » (frottement d'archet) et « *Pebble Box* » (la boîte à cailloux) sont des illustrations du concept de l'Enaction. Elles valident l'utilisation de l'architecture de simulation comme une plate-forme pour l'expérimentation et ouvrent de nouvelles perspectives de recherche sur l'enaction et la notion de présence en simulation interactive.

Abstract

This work is related to the field of Human-Computer Interaction, and particularly to the field of multisensory instrumental simulation, as conceptualized by the research group ACROE & ICA, and which needs a strong coupling between the human and the instrument.

The first part of this thesis presents various degrees of the integration of gesture in computer uses, then develops a functional approach of force feedback technologies. This analysis elicits the mainstreams that are currently sharing the field of haptics research.

We then present a study of the hardware and software components that are used in haptic simulation, and the various approaches used to connect a force feedback device to a real time modelling system. The analysis of the role of each of the components in the simulation chain and their relationships allowed us to conceptualize the “Haptic Processor Unit”. This component guarantees in particular the conditions of reactivity that are required for multisensory simulation. The new simulation architecture that we designed in this work, named ERGON_X, implements the concept of HPU. ERGON_X is a compact and transportable simulator, and handles simulation frequencies up to 44 100 Hz.

The third part presents the validation of the simulation platform ERGON_X. It mainly focuses on the design of new models, which were used in the framework of the research carried on by ACROE & ICA about instrumental interaction. The “E” is a model demonstrating the capabilities of the ERGOS technology, which is now fully exploitable thanks to this new simulation architecture. The models of tapping and of deformable paste allowed us to bring new results on human-object interaction, and validate the simulator as a tool for psychophysical experimentation. The *Enactive Emblematic Scenarii* “*Ergotic Sounds*” and “*Pebble Box*” illustrate the conception of Enaction. They validate the use of our simulation architecture as an experimental platform and lead us to a paradigm shift from “instrumental interaction” to “enactive interaction”.

Table des matières

Remerciements	iii
Résumé	ix
Abstract	xi
Table des matières	xiii
Table des figures	xvii
Liste des tableaux	xxi
Introduction Générale	1
1 Le Geste	5
1 Introduction	5
2 Typologie du geste	6
2.1 Le geste comme moyen d'action	6
2.2 Le geste comme moyen de communication	7
2.3 Le geste instrumental	7
3 Le geste et l'ordinateur	9
3.1 Les origines	9
3.2 Les contrôleurs gestuels, le mapping	11
3.3 Le retour d'effort et la réalité virtuelle	13
3.4 Le geste instrumental	16
4 Conclusion	18
2 Analyse Fonctionnelle des TGR	21
1 Introduction	22
2 La controverse du terme « système haptique »	22
3 Le choix de l'approche fonctionnelle	23
4 Méthodologie	24
5 Classification des approches fonctionnelles	26
5.1 Approche « objet »	26
5.2 Approche « homme »	31
5.3 Approche « interaction »	34
5.4 Approche « tâche »	36
6 Conclusion	46

3	Architectures pour la Simulation Haptique	47
1	Introduction	48
2	Les composants	49
2.1	Schéma général	49
2.2	Processus simulation, processus superviseur	50
3	Approches asynchrones	51
3.1	Le compromis entre puissance de calcul et réactivité	51
3.2	Haptic rendering	52
3.3	Réalité Virtuelle	55
3.4	Performances des architectures de simulation actuelles	57
4	Le contrôleur haptique	58
4.1	Introduction	58
4.2	Exemples d'implantations du contrôleur haptique	61
4.3	LHifAM – CEIT	65
5	Systèmes commerciaux	71
5.1	Sensable et les bras haptiques <i>Phantom</i>	71
5.2	Force Dimension – systèmes <i>Delta</i> et <i>Omega</i>	72
6	Approches synchrones	73
6.1	Introduction	73
6.2	Réalisations de simulateurs synchrones	74
6.3	Les simulateurs de l'ACROE	75
6.4	TELLURIS	76
7	Conclusion	79
4	Le HPU — <i>Haptic Processor Unit</i>	83
1	Introduction	83
2	Composants fonctionnels	84
3	Définition du HPU	87
3.1	Le choix du terme HPU	87
3.2	Rôle du HPU	88
3.3	Place du processus simulation	89
3.4	Interface « geste » du HPU	90
3.5	Interface « simulateur » du HPU	97
3.6	Le processeur du HPU	100
4	Conclusion	101

5	ERGON_X : un Simulateur Multisensoriel Compact et Transportable	103
1	Introduction	103
2	Architecture matérielle du simulateur	105
2.1	La carte TORO de Innovative Integration	105
2.2	Organisation fonctionnelle	106
2.3	Composition d'un simulateur-type ERGON_X	108
3	Architecture logicielle	109
3.1	Pilote de périphérique — la bibliothèque libtd	109
3.2	Développement d'une API — libErgonSim	110
3.3	Exécution d'un pas de simulation et chaîne de transmission des signaux	111
3.4	Communication hôte-DSP	113
3.5	Synchronisation du calcul de la simulation	114
3.6	Le calcul d'un pas de simulation	115
3.7	Automate d'états du simulateur	115
4	Conclusion	116
6	Tâches et Scenarii Emblématiques pour l'Interaction Instrumentale	119
1	Introduction	120
2	Le « E » : un démonstrateur pour ERGON_X et ERGOS	121
2.1	Un modèle-type pour la simulation haptique	121
2.2	Description du modèle	122
2.3	Transformation cinématique	124
2.4	Choix de la fréquence de simulation : 40 kHz	124
2.5	Raideur simulée	125
3	Simulation multisensorielle pour la psychophysique	125
3.1	Conditions nécessaires pour une démarche expérimentale	125
3.2	Perception de matériaux déformables	127
3.3	Le tapping	133
4	Scénarios Emblématiques de l'Enaction	136
4.1	Le Scénario Emblématique « Pebble Box »	136
4.2	Le Scénario Emblématique « Ergotic Sounds »	144
5	Conclusion	152
	Conclusion	155
	Bibliographie	173
	Annexes	
A	La Technologie ERGOS	A-1
B	Données pour le Scénario « PebbleBox »	B-1
1	Retours de visite	B-1
1.1	Retours de visite pour l'expérience 1 (PebbleBox de QUB)	B-1
1.2	Retours de visite pour l'expérience 2 (PebbleBox ULUND)	B-4
1.3	Retours de visite pour l'expérience 3 (PebbleBox ACROE)	B-4
2	Captures d'écran du scénario	B-6

C	Données pour le Scénario « Ergotic Sounds »	C-1
1	Statistiques sur le nombre de visites et les temps de visites	C-1
2	Analyse des résultats	C-2
3	Retours des visites	C-3
4	Captures d'écran du scénario	C-15
D	Le Formalisme CORDIS-ANIMA	D-1
1	Introduction	D-1
2	Topologie	D-2
3	Module	D-3
4	T-Simulation	D-4
5	Interface simulateur/extérieur	D-5
5.1	Transducteur gestuel rétroactif	D-5
5.2	Sortie sonore	D-6
5.3	Sortie visuelle	D-6
6	Séquence d'un pas de simulation	D-7
7	Les modules	D-9
7.1	Modules élémentaires	D-9
7.2	Modules intégrés	D-10
8	Conditions initiales	D-11

Table des figures

1.1	Raymond Goertz et le télémanipulateur E1, 1954 – extrait de [Goertz et Thompson, 1954]	10
1.2	Le transducteur gestuel rétroactif (TGR®) développé par Jean-Loup Florens en 1978 [Florens, 1978]	11
1.3	Illustration des concepts de <i>primary</i> et de <i>secondary</i> feedback dans la stratégie de mapping (extrait de [Wanderley et Depalle, 1999])	11
1.4	Structure d'un instrument composé ou virtuel – extrait de [Wanderley et Depalle, 1999]	12
1.5	Implantation de la situation de mapping	13
1.6	Représentation synthétique d'une architecture classique pour la Réalité Virtuelle .	15
1.7	Architecture pour la simulation du geste instrumental	17
2.1	Schémas d'interaction homme-objet dans la situation naturelle (à gauche) et médiatisée (à droite)	25
2.2	L'impédance mécanique comme modèle de l'objet	26
2.3	L'interface haptique vue comme un quadripôle dans un système de téléopération — extrait de [Lawrence, 1993]	27
2.4	Le musée PURE FORM développé dans le cadre du projet européen PURE FORM (www.pureform.org)	29
2.5	Schéma du « nonportable dextrous master » breveté par Zarudiansky [Zarudiansky, 1981]	30
2.6	Les gants Rutgers Master II (à gauche) et Rutgers Master II-New Design (à droite) – University of Rutgers	31
2.7	Le gant à retour d'effort CyberGrasp (à gauche), et le gant tactile CyberGlove (à droite) – Immersion Inc.	31
2.8	Un environnement de type « CAVE » – EPFL (source : Deliverable RD2.2 Enactive Interfaces [ENACTIVE, 2007b])	32
2.9	Cas exemplaires d'approche fonctionnelle de type « interaction » : le doigt sur le verre (à gauche) et la simulation de la corde frottée du violon (à droite)	35
2.10	Position de la ligne de mobilité (en pointillés) dans le cas de l'objet manipulé (à gauche) et dans le cas de l'outil (à droite)	37
2.11	Limite de l'Environnement Virtuel « restreint » (en gris)	37
2.12	Limite de l'Environnement Virtuel « élargi » (en gris)	38
2.13	La plateforme de locomotion Torus Threadmill — Iwata et al.	39
2.14	Le Haptic Master — FCS Control Systems	41

2.15	Le volant à retour d'effort Senso-Wheel commercialisé par SensoDrive	42
2.16	Le bouton rotatif programmable PR-1000 de Immersion Corporation	42
2.17	Le Pantograph — Université de McGill, Montreal	43
2.18	Deux réalisations du FEELEX : le FEELEX 1 (1997) avec une surface d'interaction de 24×24 cm (à gauche) ; le FEELEX 2 (2001), avec une surface de 5×5 cm (à droite)	44
3.1	Schéma synoptique du simulateur multisensoriel	49
3.2	Découpage d'un cycle de simulation type	51
3.3	Approche <i>haptic rendering</i> : deux boucles asynchrones permettent une meilleure répartition des ressources de calcul du simulateur	53
3.4	Organisation d'une architecture traditionnelle pour la simulation haptique – extrait de [Salisbury <i>et al.</i> , 2004]	54
3.5	Architecture de simulation pour le Rutgers Portable Master – extrait de [Burdea, 1996]	56
3.6	Algorithme de contrôle de la boucle de commande (adapté de [Hollis et Salcudean, 1992])	60
3.7	Composants logiciels de la boucle lente (workstation software) et de la boucle de commande (I/O device software) – extrait de [Hollis et Salcudean, 1992]	60
3.8	Diagramme fonctionnel du bloc « contrôleur haptique » [Bouzit <i>et al.</i> , 2002b]	62
3.9	Détail - Diagramme de la boucle de servo-contrôle [Bouzit <i>et al.</i> , 2002b]	62
3.10	Architecture pour le contrôle haptique de média (MacLean <i>et al.</i>). L'architecture de simulation haptique est synchrone, mais une boucle lente (context management) peut modifier la boucle haptique 1 kHz – extrait de [MacLean <i>et al.</i> , 1999]	74
3.11	Architecture de la plateforme de simulation temps-réel TELLURIS	77
4.1	Schéma général d'une plateforme de simulation multisensorielle : chaque transducteur est connecté au processeur principal par le biais de périphériques spécialisés.	85
4.2	Composants fonctionnels de la boucle de simulation haptique	87
4.3	Les interfaces « geste » et « simulateur » du HPU	88
4.4	Configuration 1 : le HPU héberge le processus simulation	89
4.5	Configuration 2 : le HPU n'héberge pas le processus simulation	90
4.6	Le processus simulation est partagé sur le HPU et sur la machine hôte	90
5.1	La carte TORO (Innovative Integration)	106
5.2	Schéma synoptique de la carte TORO	107
5.3	Architecture de la plateforme de simulation ERGON_X et répartition des différents processus sur les unités de calcul	109
5.4	La chaîne des signaux mise en œuvre entre les entrées/sorties du TGR ERGOSet la simulation du modèle CORDIS	112
5.5	Zone mémoire de communication entre le PC hôte et le DSP	114
5.6	Déroulement d'un pas de simulation	115
5.7	Automate d'états du programme principal exécuté sur le DSP	116

6.1	Démonstration du modèle lors de l'exposition grand public « Toucher le Futur » de la conférence Enactive/07 (Novembre 2007)	122
6.2	Représentation schématique d'un modèle conventionnel « corner-in-the-box ». L'utilisateur déplace le solide blanc dans une boîte tridimensionnelle, et peut expérimenter la qualité de restitution du mouvement libre et des contacts.	122
6.3	Capture d'écran de la simulation du « E » ; le cube jaune en bas à droite de la grille du « E » représente le point manipulé par l'utilisateur.	123
6.4	Modélisation de la forme géométrique du « E » à partir de deux formes rectangulaires élémentaires convexes et concaves	123
6.5	Comparaison entre les <LIA> RECT2D et les <LIA> BUT et BUL	124
6.6	Topologie du modèle CORDIS « E »	124
6.7	Représentation schématique du modèle simulé	128
6.8	Captures d'écran de la simulation : la pâte passe entre les obstacles de gauche (a) à droite (d) (raideur faible)	129
6.9	Courbes force d'interaction en fonction du déplacement obtenues à partir de l'enregistrement de traces expérimentales ; de haut en bas respectivement pour K_1 , K_4 et K_7	130
6.10	Schéma fonctionnel du modèle de pâtes	131
6.11	Echelle subjective moyenne attribuée par les sujets en fonction de la raideur	132
6.12	Le modèle CORDIS utilisé pour l'expérience de tapping	134
6.13	Photographie de l'expérience de tapping ; seuls le poignet et la main participent aux mouvements verticaux de percussion, le coude est au repos sur la table	135
6.14	Schéma fonctionnel de la PebbleBox développé par O'Modhain et Essl (Media Lab Europe, 2004)	137
6.15	Diagramme fonctionnel de la PebbleBox développée par Charlotte Magnusson (ULUND)	138
6.16	Schéma fonctionnel de la PebbleBox ACROE	139
6.17	Topologie du modèle de PebbleBox ; le modèle est représenté pour $n = 2$ cailloux (MAT 2D) seulement ; la visualisation est créée à partir des positions des modules MAS 2D.	140
6.18	Le Scénario Emblématique « Pebble Box »	143
6.19	A gauche, la cinématique du stick 2D ERGOS monté sur le clavier ; à droite, une photo du clavier ERGOS équipé d'un stick 2D	145
6.20	Le modèle CORDIS de frottement d'archet	145
6.21	Diagramme fonctionnel du modèle de frottement d'archet – version 2001	146
6.22	Diagramme fonctionnel du modèle de frottement d'archet – version 2007	146
6.23	Le scénario emblématique « Ergotic Sounds », lors de l'exposition « Toucher le Futur » de la conférence Enactive/07 (Novembre 2007)	149
6.24	Observation objective de la tâche d'inversion du mouvement d'archet en maintenant la continuité du son. Les signaux sont extraits du modèle (unités : UL_v Unité de Longueur virtuelle, UF_v Unité de Force virtuelle) L'oscillation de la corde est maintenue pendant l'inversion du mouvement de l'archet. On peut observer que la pression appliquée ne s'annule pas pendant l'inversion, et que la force de friction change de signe au moment de l'inversion du mouvement.	151

A.1	Trois systèmes ERGOS équipés de différents habillages mécaniques : à gauche, un stick 6D, au milieu un stick 3D, à droite un archet	A-3
D.1	Représentation schématique des points de communication M et L	D-2
D.2	Règles de connexion des points de communication CORDIS	D-3
D.3	Un exemple de réseau CORDIS-ANIMA construit à partir des deux types de modules élémentaires.	D-3
D.4	Modules <MAT> et <LIA>	D-4
D.5	Connexion d'un TGR en mode impédance au modèle CORDIS	D-6
D.6	Connexion d'un TGR en mode admittance au modèle CORDIS	D-6
D.7	Connexion d'une sortie sonore à un modèle CORDIS	D-6
D.8	Séquence des phases <LIA> et <MAT> avec un TGR en mode impédance	D-8
D.10	Séquence d'un pas de simulation CORDIS	D-8
D.9	Séquence des phases <MAT> et <LIA> avec un TGR en mode admittance	D-9

Liste des tableaux

2.1	Représentation schématique des quatre approches fonctionnelles de l'interaction homme-objet	33
2.2	Les quatre catégories de l'approche fonctionnelle « tâche »	45
3.1	Synthèse des principales caractéristiques des systèmes présentés	67
6.1	Stratégies d'exploration et contenu de la description subjective de la scène en fonction de la relation geste-son	144
A.1	Caractéristiques de la base ERGOS	A-2
A.2	Caractéristiques des habillages mécaniques	A-2
C.1	Nombre de visites	C-1
C.2	Durée totale des visites (h :min :sec)	C-1
C.3	Nombre de visites de durée supérieure à 30 s par expérience	C-2
C.4	Durée moyenne des visites de durée supérieure à 30 s par expérience	C-2
C.5	Dans quel cas avez-vous le mieux réussi la tâche demandée ? (Cas 1 : friction faible ; Cas 2 : friction normale ; Cas 3 : friction exagérée)	C-2
C.6	Quel était, selon vous, le cas le plus agréable à jouer ? (Cas 1 : friction faible ; Cas 2 : friction normale ; Cas 3 : friction exagérée)	C-3

Introduction Générale

Le travail présenté dans cette thèse entre dans le cadre des recherches effectuées par l'ACROE¹ et le laboratoire ICA², que nous appellerons dans ce document le groupe de recherche ACROE & ICA. Ce travail a été soutenu par le réseau d'excellence européen Enactive Interfaces³. Il s'inscrit dans les domaines de l'informatique appliquée à la création artistique (informatique musicale, image animée) et de la simulation instrumentale multisensorielle telle que conceptualisée par le laboratoire et qui implique un couplage fort homme-instrument.

* * *

Les domaines de la Réalité Virtuelle, de la simulation instrumentale par ordinateur et de la technologie du retour d'effort entrent dans une période de forte croissance. Chaque année voit la création de nouvelles sociétés spécialisées dans le développement d'interfaces gestuelles, d'architectures matérielles, dans le développement d'outils logiciels de simulation, de conception assistée par ordinateur, de prototypage, etc.

La loi de Gordon Moore prédit depuis 1965 le doublement de la puissance de calcul des microprocesseurs tous les 18 mois. Cette loi se vérifie encore aujourd'hui, même si nous arrivons à un stade où l'augmentation de la puissance des processeurs ne peut plus croître aussi rapidement parce que nous approchons de certaines limites physiques⁴. Dans le domaine des Réalités Virtuelles et de la simulation par ordinateur, la puissance des architectures est encore un facteur limitant, mais l'augmentation rapide de la puissance des microprocesseurs fut un facteur de développement important, permettant aux technologies de Réalité Virtuelle et de la simulation multisensorielle d'aborder de nouveaux champs d'application jusque là inabordables en raison de la complexité calculatoire nécessaire.

La croissance de la puissance de calcul des microprocesseurs conjuguée à une miniaturisation importante et à une diminution de la consommation électrique a permis d'équiper de nombreux

¹Association pour la Création et la Recherche sur les Outils d'Expression. Centre de recherche conventionné du ministère de la Culture.

²Informatique et Création Artistique. Equipe d'accueil EA 2934 du Ministère de l'Enseignement Supérieur et de la Recherche

³IST-2004-002114-ENACTIVE. Le programme d'activité du réseau d'excellence Enactive Interfaces s'est déroulé de Janvier 2004 à Mars 2008.

⁴La loi de Moore devrait en principe rester valide jusqu'en 2015, avant que la technologie des processeurs ne soit confrontée aux effets quantiques. Certains problèmes technologiques peuvent se poser d'ici là, en particulier parce que la taille des transistors est de plus en plus réduite (bientôt quelques dizaines d'atomes seulement par transistor) et que les procédés de fabrication sont extrêmement coûteux.

objets électroniques de microprocesseurs. Le super ordinateur des années 1950, puis le micro-ordinateur des années 1980 se diversifient dans de nombreuses solutions informatiques.

Les technologies du multimédia ont également connu une phase de développement sans précédent ces dernières années. L'écran cathodique traditionnel se conjugue aujourd'hui en écran mural ultraplat de grandes dimensions, projecteur numérique avec une résolution suffisante pour la projection sur de grands écrans des salles de cinéma, mais aussi en afficheurs miniaturisés à faible consommation électrique pour systèmes personnels intelligents : téléphone portable, agenda électronique, et « stations » multimédia de poche. Le domaine du son est également un domaine porté par une forte innovation technologique, sous l'effet de l'explosion des technologies multimédia et de leur démocratisation.

* * *

La technologie pour la simulation instrumentale multisensorielle se situe à l'intersection des domaines technologiques que nous venons d'évoquer : technologie des microprocesseurs et des architectures de calcul, technologie du multimédia pour le retour visuel et le retour sonore, génie logiciel et techniques de simulation. Un des composants essentiels d'une plateforme d'interaction multisensorielle que nous n'avons pas encore cité est l'interface gestuelle, grâce à laquelle un utilisateur peut interagir avec l'objet ou l'environnement virtuel simulé. Les technologies pour l'interaction gestuelle sont apparues dès les années 1950 avec les premiers télémanipulateurs électriques, puis avec l'invention de la souris en 1970. L'utilisation de systèmes gestuels en boucle ouverte tels que la souris ou le clavier s'est rapidement développée dès les années 1970 et standardisée avec l'utilisation du micro-ordinateur dans les années 1980, mais le retour d'effort est resté confiné au domaine de la recherche jusque dans les années 1990. En 1994, la société américaine Sensable commercialise le Phantom. Quelques systèmes à retour d'effort étaient déjà commercialisés de manière marginale, mais ce système fut le premier commercialisé à grande échelle⁵. L'apparition de systèmes commerciaux a contribué à populariser la technologie du retour d'effort et son utilisation dans de nombreux domaines. Le premier système à retour d'effort pour la manipulation dans un espace 3D et avec un prix de commercialisation adapté au grand public est apparu en 2007 : le Falcon, produit par la société Novint. On trouve aujourd'hui des applications du retour d'effort dans de nombreux domaines : téléopération, robotique industrielle, aérospatiale, chirurgie assistée par ordinateur, apprentissage, prototypage industriel, création artistique, etc. Tous ces domaines d'application sont un terrain fertile pour le développement de la technologie à retour d'effort et de nouveaux besoins.

* * *

Le travail que nous présentons dans cette thèse se situe dans le cadre de ce vaste domaine, et est orienté sur la technologie du retour d'effort en situation instrumentale multisensorielle. Ce travail est centré autour d'une des hypothèses fondatrices du groupe de recherche ACROE & ICA selon laquelle l'interaction instrumentale, telle que définie dans ce programme de recherches, nécessite un couplage fort entre l'homme et l'objet manipulé. Il s'agit à la fois d'un travail d'analyse sur les

⁵Dans le domaine de la technologie pour le retour d'effort, « grande échelle » signifie encore à l'heure actuelle une production de quelques milliers d'unités par an seulement.

orientations, les tendances et les grands principes technologiques, d'un travail de développement d'une nouvelle plate-forme de simulation destinée spécialement aux situations instrumentales multisensorielles exigeant une grande réactivité de la part du simulateur, et enfin d'un travail sur la validation et les usages de la plate-forme développée.

Dans le chapitre 1, nous présentons les concepts généraux associés à la notion de geste, ainsi que les trois principaux paradigmes actuels utilisés pour l'intégration du geste dans l'ordinateur : le mapping, le « haptic rendering » et le geste instrumental. Le paradigme du geste instrumental est l'axe central de ce travail, et c'est lui qui en guidera les choix conceptuels et technologiques. Ce travail de recherche s'appuie également sur les travaux de F. Varela sur la théorie de l'enaction. Le paradigme du geste instrumental, en plaçant au centre de l'interaction homme-objet les principes de couplage mécanique et de couplage énergétique, est une situation exemplaire de l'approche enactive. La fin de ce chapitre mettra en avant quelques uns des liens entre enaction et geste instrumental.

Les chapitres 2 et 3 présentent respectivement les grandes tendances conceptuelles et technologiques qui se partagent le champ de l'interaction homme-objet dans le cadre des Réalités Virtuelles. Nous en tirons en particulier les grands principes théoriques et techniques à même de supporter l'approche enactive pour le geste instrumental : le paradigme de l'enaction et du geste instrumental sont relatifs à l'approche fonctionnelle "interaction" présentée dans le cadre du chapitre 2, et c'est l'utilisation d'architectures de simulation synchrones présentées à la fin du chapitre 3 qui permet de les implanter dans une architecture de simulation.

Une fois ces concepts théoriques et technologiques clairement posés, les chapitres 4 et 5 présentent leur mise en oeuvre dans le cadre de cette thèse. Le chapitre 4 présente la formalisation du concept de *HPU (Haptic Processor Unit)*. De la même manière que la plupart des architectures informatiques sont aujourd'hui équipées d'unités spécialisées dans les calculs graphiques ou pour le traitement des signaux sonores, nous présentons ici le HPU comme le composant d'une architecture pour la simulation instrumentale multisensorielle destinée à interfacer un Transducteur Gestuel Rétroactif au reste de l'architecture de simulation. Il s'agit du composant d'une architecture de simulation qui garantit les principes de synchronisation et de multisensorialité à l'interface du TGR. Le chapitre 5 présente alors plus en détails la réalisation d'une nouvelle plate-forme de simulation multisensorielle réalisée selon ces principes. Nous appelons cette plate-forme *ERGON_X*, et sa réalisation est centrée sur l'implantation concrète du concept de HPU présenté dans le chapitre 4.

Dans le chapitre 6, nous présentons les modèles pour l'interaction instrumentale qui ont été réalisés pour le simulateur *ERGON_X*. Ces modèles ont permis de valider la plateforme de simulation et d'avancer des résultats nouveaux sur la perception multisensorielle dans la communication instrumentale.

Chapitre 1

Le Geste

Sommaire du Chapitre

1	Introduction	5
2	Typologie du geste	6
2.1	Le geste comme moyen d'action	6
2.2	Le geste comme moyen de communication	7
2.3	Le geste instrumental	7
3	Le geste et l'ordinateur	9
3.1	Les origines	9
3.2	Les contrôleurs gestuels, le mapping	11
3.3	Le retour d'effort et la réalité virtuelle	13
3.4	Le geste instrumental	16
4	Conclusion	18

1 Introduction

Le thème central de ce mémoire est le geste dans le cadre de l'interaction homme-ordinateur. Dans ce chapitre, nous présentons dans un premier temps la notion de geste dans son sens le plus général, ainsi que les termes associés « action » et « mouvement ». Nous présentons ensuite la typologie du geste introduite par Cadoz en 1994 [Cadoz, 1994] qui distingue les trois fonctions possibles du geste : sémiotique, épistémique et ergotique, et qui introduit le geste instrumental, comme une catégorie particulière de gestes où ces trois fonctions sont réunies. Dans un deuxième temps, nous présentons un rapide historique de l'introduction du geste dans l'utilisation de l'ordinateur, puis les trois principaux paradigmes actuels de l'implantation du geste dans l'ordinateur. En particulier, nous montrons les conditions nécessaires pour retrouver la situation de geste instrumental dans la relation homme-ordinateur.

2 Typologie du geste

2.1 Le geste comme moyen d'action

« Action » et « mouvement » sont des termes fréquemment employés lorsqu'il s'agit de décrire le geste utilisé pour agir. On peut dire, sans trop de conteste, que le mouvement est le déplacement dans l'espace d'une partie ou de la totalité d'un système (un être humain, un objet...). L'action ou la tâche sont le résultat des mouvements. Ces concepts décrivent le résultat de mouvements successifs ou la manière d'effectuer une suite de gestes : je décris une action lorsque je dis que je tape sur le clavier de mon ordinateur, je décris une suite de mouvements lorsque je dis que mon index a frappé de telle manière la touche 'e' du clavier, puis s'est déplacé ainsi vers un autre endroit du clavier, etc.

Pour Smyth et Wing, psychologues du comportement, il y a deux manières de décrire le mouvement : une est phénoménologique, l'autre résume l'action ou la suite de mouvements à un événement.

« L'action diffère du mouvement un peu de la même façon que la perception diffère de la sensation. Lorsqu'on décrit une suite de mouvements, on dit 'J'ai pris le crayon', mais on ne dit pas 'j'ai bougé ma main suivant cette trajectoire', ou 'j'ai contracté mes muscles dans cet ordre'. C'est que, quand nous pensons à l'action, définie en termes de but, nous ne pensons pas aux mouvements particuliers par lesquels elle a été mise en œuvre. L'utilisation du terme action implique une description d'un plus haut niveau de ce que nous faisons. »

Mary M. Smyth and Alan M. Wing. The psychology of Human Movement, chapter 1. Movement, action and skill, pages 1–15. Academic Press, 1984. [Smyth et Wing, 1984]

Smyth et Wing distinguent trois niveaux d'observation d'une action :

Niveau 1. L'action proprement dite, ou la tâche, qui est la réalisation d'un événement : atteindre une cible, boire un verre, tomber à terre...

Niveau 2. Le mouvement dans son unicité, sa configuration spatiale et temporelle, c'est-à-dire la manière dont une action est effectuée, par exemple l'ensemble des trajectoires, les muscles mis en jeu...

Niveau 3. Les éléments de ce mouvement qui en font un item catégorisable, la structure spatiale et temporelle qui reste constante entre deux mouvements, deux réalisations d'une même action (on parle aussi parfois de patterns).

On peut aussi appliquer cette classification, effectuée pour l'analyse des mouvements humains, pour l'analyse de tout type de mouvement. Ainsi, dans le cas de la chute d'un corps : l'action serait le fait de tomber, le mouvement serait la trajectoire réellement effectuée, l'évolution de la vitesse de la chute, sa durée... Les éléments permettant de catégoriser ce mouvement seraient par exemple la vitesse du mouvement, propre à un corps en chute, les frottements du corps en chute libre dans l'air...

2.2 Le geste comme moyen de communication

Le geste est un moyen d'action de l'homme sur son environnement, mais aussi un moyen de communication. Communication et action sont deux fonctions qui sont intimement mêlées au sein du même geste.

Cadoz utilise le terme de « canal gestuel » pour regrouper toutes les fonctions du geste en tant que moyen de communication [Cadoz, 1994]. À l'inverse des autres canaux de communication humains que sont par exemple l'ouïe ou la vue, qui permettent uniquement à l'homme de recevoir des informations de l'environnement, le canal gestuel est bidirectionnel, c'est-à-dire qu'il permet de recevoir des informations de l'environnement, mais aussi d'en émettre. Cadoz décrit les trois fonctions du canal gestuel : les fonctions sémiotique, épistémique et ergotique que nous rappelons ici très brièvement :

- La fonction *épistémique* du geste est la fonction qui permet d'acquérir la connaissance de notre environnement, de recevoir l'information. Par le toucher, le sens kinesthésique (informations renvoyées par les muscles) et des mouvements exploratoires, il est possible de prendre connaissance des propriétés mécaniques, rhéologiques et géométriques d'un objet manipulé, telles que l'élasticité, l'état de surface, le poids, la température, etc. Pour décrire de manière complète la fonction épistémique du geste, Cadoz parle de sens tactilo-proprio-kinesthésique [Cadoz, 1994].
- La fonction *ergotique* du geste a trait à l'action matérielle de l'homme sur l'environnement. Ceci ne peut être rendu possible que par l'interaction mécanique : la fonction ergotique nécessite donc une mise en contact physique, et un échange d'énergie entre le corps humain et l'objet manipulé.
- La fonction *sémiotique* du geste est la fonction par laquelle l'homme peut envoyer des informations à destination de l'environnement. Cette fonction peut s'exercer sans contact physique avec l'environnement, comme dans le geste accompagnant la parole ou le geste du chef d'orchestre.

Cette typologie décrit les trois fonctions essentielles du canal de communication gestuel : émettre de l'information, en recevoir, et interagir avec l'environnement. Notons que ces trois fonctions sont exclusives, mais qu'elles se trouvent intimement mêlées dans la plupart des utilisations du geste : par exemple, prendre un objet en main c'est interagir avec l'environnement, échanger de l'énergie en donnant de l'énergie potentielle ou cinétique à cet objet (fonction ergotique), éventuellement le déformer (fonction ergotique), mais aussi percevoir son poids (fonction épistémique).

2.3 Le geste instrumental

Cadoz définit le *geste instrumental* comme une catégorie particulière de gestes dont l'origine se trouve dans une interaction mécanique avec un objet (l'instrument) mais dont le but est la production d'un phénomène de type informationnel.

Le geste instrumental a de particulier qu'il regroupe les trois fonctions du geste :

1. Il est sémiotique parce qu'il s'adresse à un auditeur, au récepteur de l'information,
2. Il est épistémique parce que la « réaction », le comportement de l'instrument en situation de jeu est un paramètre important dans la réalisation du jeu musical,

3. Il est éminemment ergotique, parce qu'il ne peut y avoir de production sonore sans que l'instrumentiste n'apporte d'énergie à son instrument pour le mettre en vibration¹ :

« Le geste instrumental musical n'est performant dans sa fonction informationnelle que parce qu'il s'effectue avec une certaine dépense d'énergie. Il n'est efficacement sémiotique que parce qu'il est ergotique. [...] Ce n'est que parce qu'il y a une interaction et une dépense d'énergie qu'il y a efficacité communicationnelle. » [Cadoz, 1994]

La définition du geste instrumental de Cadoz est la suivante :

1. Le geste instrumental s'applique à un objet matériel, et il y a interaction physique avec celui-ci.
2. Dans le cadre de cette interaction se produisent des phénomènes physiques différenciés dont les formes et les évolutions dynamiques peuvent être maîtrisées par l'instrumentiste.
3. Ces phénomènes peuvent alors devenir les supports de messages communicationnels.

Ajoutons que le geste instrumental, en tant que geste de production, est un geste dont l'objet est la structure vibrante de l'instrument. Pour qu'il y ait communication instrumentale effective, il faut que la source de la production sonore soit en relation mécanique directe avec l'instrumentiste. En d'autres termes, il faut que la dépense d'énergie mette en relation l'instrumentiste et la structure vibrante de l'instrument. Nous le verrons dans la suite de ce document, cette condition aura des conséquences technologiques importantes pour l'implantation du geste instrumental avec l'ordinateur.

Le geste instrumental est un type de geste particulier, dans le sens où il implique à la fois l'action de l'homme sur son environnement², mais aussi les facultés de l'homme-instrumentiste à recevoir des informations en provenance de l'environnement : perception mécanique directe de l'instrument, retours visuels et sonores de l'instrument (justesse de la hauteur du son, position du corps par rapport au clavier d'un piano...) ou de l'environnement (jeu des autres instrumentistes par exemple). Dans le geste instrumental, 'action' de l'homme vers l'instrument et 'rétroaction' de l'instrument vers l'homme sont profondément liées et participent étroitement à la naissance du jeu instrumental.

¹Dans les instruments non électrifiés tout du moins. Le cas de l'orgue est particulier parce que le principe de son fonctionnement met en jeu un mécanisme de relais énergétique. Le mécanisme des claviers est donc totalement découplé énergétiquement et mécaniquement de la partie de l'orgue supportant la vibration sonore (le tuyau), et il est souvent prétendu — à tort ! — que la fonction ergotique du geste n'est pas nécessaire au jeu de l'organiste. Si l'organiste n'est pas en relation mécanique directe avec le phénomène de production sonore, les subtilités de jeu de cet instrument nécessitent néanmoins que l'organiste soit en contact mécanique direct avec le mécanisme de transmission de l'orgue. À partir du début du XX^e siècle, les nouveaux instruments construits ont pris des dimensions imposantes. La transmission mécanique de l'orgue devenait alors trop complexe et trop lourde pour être seulement mue par la force des doigts de l'organiste. Divers moyens de relais énergétiques furent inventés (par exemple, la machine Barker), puis la transmission mécanique fut remplacée par des interrupteurs associés à des électro-aimants permettant d'actionner les soupapes sous les tuyaux. D'abord considéré comme un gain en confort permettant de jouer de très grands instruments, le découplage du jeu des claviers de la transmission mécanique fut sévèrement critiqué par une grande partie des interprètes et des compositeurs. Depuis la fin des années 1970, la facture instrumentale moderne est revenue à la construction d'instruments entièrement basés sur une transmission mécanique, preuve s'il en est que l'orgue n'est pas un instrument pouvant se passer de la fonction ergotique du geste !

²En fait, sur un objet particulier de cet environnement : l'instrument.

3 Le geste et l'ordinateur

3.1 Les origines

La fin des années 1950, puis les années 1960 sont un tournant dans l'histoire de l'informatique. C'est à cette période en effet que l'ancêtre de l'ordinateur évolue du statut de « supercalculateur » que l'on peut programmer à l'aide de commutateurs et de multiples branchements de câbles, à la forme primitive de l'ordinateur que nous connaissons maintenant. Ces premiers ordinateurs se programment alors à l'aide de cartes perforées, et les premiers claviers sont utilisés pour la réalisation des cartes perforées dans une machine semblable à la machine à écrire.

Ce n'est qu'à partir des années 1960 que le clavier apparaît pour une utilisation telle que nous la connaissons maintenant. Les entrées clavier sont tout d'abord répercutées sur une sortie papier (machine PDP-1, construite par Digital Equipment Corporation en 1960), puis sur une sortie vidéo dès 1964, date de l'apparition des *terminaux*³ : une unité composée d'un écran vidéo associé à un clavier. La souris naît avec l'apparition des premiers terminaux graphiques. Dès 1964, D. Engelbart crée le premier prototype d'interface gestuelle de type pointeur. Le concept est breveté en 1970 [Engelbart, 1970].

L'apparition des premières interfaces gestuelles pour l'ordinateur est née du besoin de pouvoir interagir plus facilement avec cette nouvelle machine, à l'époque surtout utilisée comme un supercalculateur programmable. Contrôler ce type de machine signifie alors surtout pouvoir la programmer. Les premiers claviers permettent donc d'interagir avec l'ordinateur à un niveau symbolique, au niveau du langage : les informations entrées dans l'ordinateur sont des caractères, des mots. Cependant dès les années 1960, avec l'apparition des premiers terminaux graphiques, l'ordinateur est non plus seulement envisagé comme un supercalculateur, mais comme un outil de représentation [Lecarme, 1971]. Il s'agit alors de créer les nouvelles interfaces permettant l'interaction ces nouvelles représentations. L'article de 1965 de Sutherland [Sutherland, 1965], décrivant les futures interfaces gestuelles qui permettraient à l'homme d'interagir avec ces nouveaux mondes représentés par l'ordinateur, est représentatif des nouvelles manières de penser l'ordinateur qui apparaissent alors.

Avec l'émergence de l'informatique « moderne », on développe en parallèle aux interfaces gestuelles pour la manipulation des systèmes intelligents pouvant être commandés soit par commande vocale, soit par communication gestuelle de type *sémiotique*. « Put That Here » [Schmandt et Hulteen, 1982] est emblématique des travaux de recherche qui furent développés dans ce but : il s'agit d'un système implémenté au MIT par Schmandt et Hulteen, permettant à l'utilisateur de contrôler la machine par le biais de commandes vocales et gestuelles (désignation) : l'utilisateur pouvait déplacer un objet virtuel en proférant : « Put » (« mets ») « That » (« ça », en désignant un objet du doigt) « Here » (« là », en désignant un lieu). Ces travaux concernent aujourd'hui la capture de mouvement, la reconnaissance de forme et la reconnaissance vocale.

À partir des années 1940, on construit des télémanipulateurs afin de protéger les opérateurs des rayonnements causés par les matières radioactives manipulées. Les premiers télémanipulateurs sont entièrement mécaniques, mais de conception relativement complexe, puisqu'il faut fournir à l'utilisateur un espace de manipulation suffisamment grand avec plusieurs degrés de liberté, tout

³Système MULTICS, créé par MIT, Bell Laboratories et General Electric.

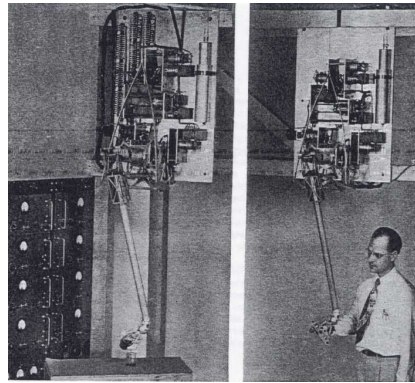


Figure 1.1. Raymond Goertz et le télémanipulateur E1, 1954 – extrait de [Goertz et Thompson, 1954]

en limitant au maximum le jeu de transmission mécanique ou les frottements. En 1952, R. Goertz conçoit au Argonne National Laboratory un télémanipulateur à couplage électrique [Goertz, 1952, Goertz et Thompson, 1954]. La véritable innovation apportée par Goertz lorsqu’il conçoit le télémanipulateur E1 est de remplacer le couplage mécanique assurant la transmission des mouvements entre les deux sites distants par des signaux électriques, permettant ainsi de s’affranchir du couplage mécanique dues à l’éloignement (figure 1.1). Afin de restituer à l’opérateur la sensation des efforts exercés sur l’objet distant manipulé, chaque site est équipé d’actionneurs et de capteurs. R. Goertz crée le concept d’*interface pour le geste à retour d’effort*.

Le premier télémanipulateur à couplage électrique conçu par Goertz en 1954 est souvent considéré comme l’ancêtre des systèmes haptiques modernes. À partir des années 1960 jusqu’au début des années 1980, ce nouveau concept d’interface à retour d’effort sera décliné dans les champs de la robotique [Jacobsen *et al.*, 1986], de la téléopération [Vertut et Coiffet, 1984], de l’informatique musicale [Cadoz *et al.*, 1982] ou des HCI [Sutherland, 1965].

En 1971, James Batter et Frederick Brooks utilisent le télémanipulateur GROPE-I, développé en 1968 [Steelman, 1968], dans une application de simulation d’objets virtuels [Batter et Brooks, 1971]. Deux des sept degrés de liberté du manipulateur à retour d’effort sont utilisés pour permettre à l’utilisateur d’expérimenter divers modèles de champs exercés sur une particule soumise à un champ de force. Il s’agit de la première utilisation de l’ordinateur pour la représentation d’objets rendus accessibles à un utilisateur par le moyen de l’interaction.

En 1978 Jean-Loup Florens met au point à l’ACROE le transducteur gestuel rétroactif [Florens, 1978] : il s’agit de la première interface gestuelle spécialement conçue pour la simulation d’objets virtuels (figure 1.2). Les algorithmes de simulation sont alors implantés sur des cartes électroniques analogiques, et ce n’est qu’au début des années 1980 que seront développées les premières interfaces haptiques contrôlées par ordinateur [Cadoz *et al.*, 1982, Cadoz *et al.*, 1984].

Au début des années 1980, le développement des systèmes haptiques prend un nouveau tournant, pour évoluer de prototypes expérimentaux au statut prospectif vers l’utilisation de systèmes plus matures dont l’utilisation est appliquée à des cas concrets. Les années 1990 voient le développement des premiers systèmes commercialisés avec les premiers gants à retour d’effort (VPL Research) et le bras haptique Phantom [Massie et Salisbury, 1994]. De la même manière

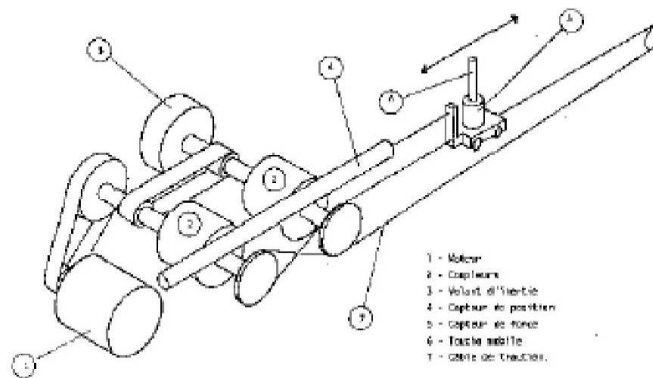


Figure 1.2. Le transducteur gestuel rétroactif (TGR®) développé par Jean-Loup Florens en 1978 [Florens, 1978]

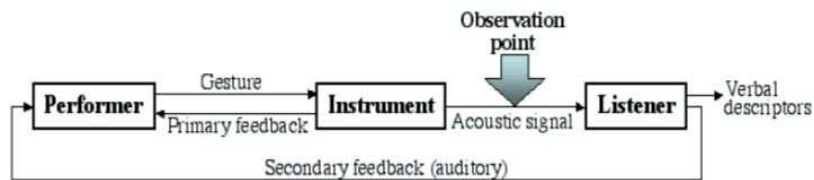


Figure 1.3. Illustration des concepts de primary et de secondary feedback dans la stratégie de mapping (extrait de [Wanderley et Depalle, 1999])

que les premières interfaces haptiques sont apparues dans des domaines divers, les premiers systèmes commerciaux sont destinés à des domaines aussi divers que l'infographie, le prototypage industriel, les arts numériques, etc. [sen, 2007].

3.2 Les contrôleurs gestuels, le mapping

Dans la situation instrumentale classique (i.e. pas dans le sens cadozien), Wanderley et Depalle considèrent que l'instrumentiste reçoit deux types de retour instrumental [Wanderley et Depalle, 1999] (figure 1.3) : (1) la *rétroaction primaire*, constituée des retours visuel, tactilo-kinesthésique et auditif primaire de l'instrument ; (2) la *rétroaction secondaire* est constituée par le *retour du signal sonore de l'instrument* aux oreilles de l'instrumentiste.

Dans la stratégie de mapping, la boucle de rétroaction primaire n'englobe plus la partie acoustique de l'instrument en raison de la rupture introduite par la couche signal, mais on fait l'hypothèse que la boucle secondaire est suffisante pour le jeu instrumental. Ainsi, la stratégie de mapping fait l'usage de 'contrôleurs gestuels' : il s'agit de capteurs (de mouvement, de pression, etc.) permettant d'envoyer un flux d'information unidirectionnel de l'utilisateur vers l'ordinateur. Ce flux d'information peut être de type événementiel (le clic sur un bouton de souris, la pression d'une touche sur un clavier) ou de type signal (l'évolution des coordonnées X et Y de la souris au cours du temps). Les retours informationnels à l'utilisateur sont principalement de types auditif et visuel. On appelle ces interfaces des 'contrôleurs gestuels' parce que le paradigme de manipulation implicite est celui du contrôle : étant donné que le flux d'information gestuelle va seulement de

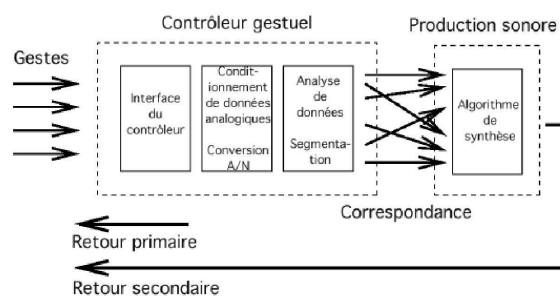


Figure 1.4. Structure d'un instrument composé ou virtuel – extrait de [Wanderley et Depalle, 1999]

l'utilisateur vers l'ordinateur, il ne s'agit pas à proprement parler d'interaction (bien que si on considère la situation multisensorielle dans sa globalité il y a bien une interaction possible par le biais des retours sonores et visuels).

En comparaison avec la situation instrumentale traditionnelle (i.e. non électrifiée), cette situation introduit une rupture forte entre le composant en relation avec l'instrumentiste (le contrôleur gestuel), et le composant responsable de la production du phénomène sonore (le processus de synthèse). La relation entre ces deux composants est supportée par une couche signal unidirectionnelle (figure 1.4) : ainsi, l'« instrument mécanique » et l'« instrument acoustique » sont dissociés.

Le terme anglais de *mapping* signifie dans ce contexte « association avec », « mise en relation ». La situation de *mapping* exploite la rupture introduite par la couche signal : les paramètres ou les variables traités en sortie des contrôleurs gestuels sont attribués artificiellement aux paramètres d'entrée des processus de synthèse (figure 1.4). La dissociation introduite par la couche signal entre le contrôleur gestuel (l'instrument mécanique) et le processus de synthèse (l'instrument acoustique) permet d'ouvrir un nouveau champ de possibilités instrumentales, par la combinaison de différents contrôleurs gestuels, par l'extension d'un instrument traditionnel avec de nouveaux contrôleurs gestuels, ou encore en plaçant l'instrumentiste à un meta-niveau de contrôle du phénomène musical. L'instrumentiste se retrouve face à un panel immense de possibilités d'association d'un geste à un événement sonore, en comparaison de la situation instrumentale classique où le musicien est dans une situation « one-gesture-to-one-acoustic-event » [Wessel et Wright, 2002]. Le niveau d'expressivité nécessaire aux interfaces pour la création musicale est alors recréé par un ajustement fin des relations entre les flux de données venant des contrôleurs gestuels et les paramètres d'entrée des processus [Hunt *et al.*, 2002].

La stratégie de mapping est donc essentiellement centrée sur la fonction sémiotique du geste, les retours sensoriels étant assurés par l'audition ou la vision. Le geste ergotique est présent, de par l'interaction mécanique entre l'utilisateur et les contrôleurs ou les capteurs, mais l'ergotécité du geste n'est pas ici une composante fonctionnelle du jeu dans la stratégie de mapping.

Les instruments de musique électronique ne sont pas les seuls à implémenter ce que nous pouvons appeler de manière générale des stratégies de 'mapping'. Dans toutes les situations où l'utilisateur se trouve en interaction avec un ordinateur par le biais de contrôleurs gestuels, c'est-à-dire de capteurs, le schéma fonctionnel est relativement le même (figure 1.5).

- La bande passante du flux d'information entre le capteur et le processus de simulation est très variable suivant les applications. L'information peut être de type phénoménologique (mesure de

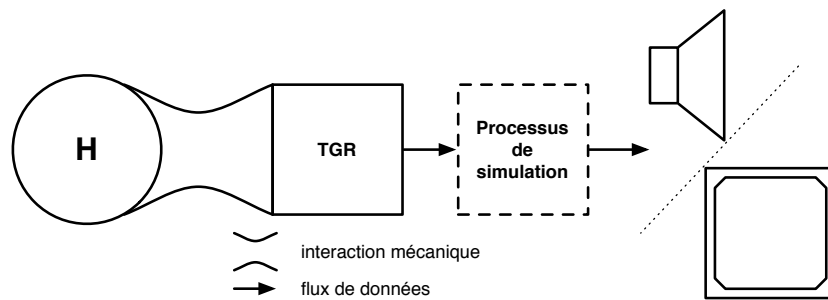


Figure 1.5. Implantation de la situation de mapping

positions par exemple), ou bien événementielle (envoi d'une commande, par exemple de type MIDI).

- Le processus de simulation est exécuté à une fréquence dépendant des transducteurs utilisés en sortie du simulateur ; s'il y a une sortie sonore, le processus doit être capable de générer les flux d'échantillons avec une bande passante de 44,1 kHz au minimum. S'il y a une sortie visuelle, la fréquence des flux d'information à destination de l'écran est très variable [3 0– 100 Hz] et dépend du dispositif d'affichage.
- L'ensemble des processus qui composent la chaîne de mapping travaille de manière totalement asynchrone. La seule contrainte temporelle se situe au niveau de l'utilisateur : il faut que le délai éventuellement généré par les différents processus ne soit pas perceptible de l'utilisateur, ou bien qu'il apparaisse comme une des composantes fonctionnelles de l'instrument.

Cependant, si le panel de possibilités d'associations d'un geste à un événement sonore est considérablement élargi, la situation de mapping ne permet plus d'obtenir le haut degré d'expressivité existant dans la situation instrumentale classique :

« Most traditional acoustical musical instruments are not easy to play at first but afford the development of a high degree of musicality. On the other hand, many simple-to-use computer interfaces proposed for musical control seem — after even a brief period of use — to have a toy-like character. »

David Wessel and Matthew Wright. Problems and prospects for intimate musical control of computers. Computer Music Journal, 26(3) :11–22, Fall 2002. [Wessel et Wright, 2002]

En particulier, l'utilisation d'interfaces gestuelles de type contrôleur ne permet pas d'implanter dans l'ordinateur la rétroaction primaire tactilo-kinesthésique, qui nécessite l'utilisation du retour d'effort.

3.3 Le retour d'effort et la réalité virtuelle

Plusieurs domaines de recherche intéressés par l'interaction entre l'homme et l'ordinateur, et plus généralement l'interaction entre l'homme et la machine, ont tenté de savoir comment il était possible d'améliorer les performances gestuelles de l'utilisateur en interaction avec un ordinateur

ou une machine. La première solution, dite de mapping, consiste à *augmenter* les possibilités de contrôle gestuel de l'utilisateur par l'extension des commandes qu'il a à sa disposition, c'est-à-dire à augmenter les capacités sémiotiques du geste dans la situation d'interaction avec l'ordinateur. La seconde solution est d'ajouter dans la boucle d'interaction la fonction épistémique du geste, soit par l'ajout de retour tactile, soit par l'ajout de ce que l'on appelle le « retour d'effort » [Lin et Salisbury, 2004].

Dans la plupart des applications de Réalité Virtuelle en effet, la boucle haptique est considérée comme un moyen pour l'utilisateur de connaître son environnement. Ce moyen d'en obtenir l'information est envisagé soit de manière exclusive, soit (le plus souvent) en complément des retours visuel et sonore.

Dans le domaine des interfaces homme-machine (HCI), Rosenberg et Brave ont montré que l'utilisation du retour d'effort peut améliorer les tâches de pointage, de sélection ou de positionnement d'objets dans des interfaces graphiques [Rosenberg et Brave, 1996]. Les boutons et fenêtres de l'environnement graphique étaient augmentés de champs de force attractifs ou répulsifs. Les résultats d'une étude expérimentale sur 5 sujets ont montré que les temps de réalisation d'une tâche étaient écourtés de 10 à 50 % à l'aide de retour d'effort.

Dans le domaine des logiciels pour l'édition de son, la gestion d'une interface conviviale et efficace pour l'utilisateur est un domaine d'étude typique des HCI. Chu présente l'intégration de boutons à retour d'efforts (haptic knobs) dans un logiciel d'édition de son [Chu, 2002], afin d'ouvrir de nouvelles techniques de navigation et d'édition. Dans ces deux exemples, nous pouvons remarquer que le retour d'effort est utilisé sans que la notion d'objet physique simulé ne soit présente, mais seulement afin d'implémenter ce que nous appelons la fonction épistémique du geste.

Dans le domaine de l'infographie, le retour haptique est généralement considéré comme un moyen donné à l'utilisateur de percevoir la *forme* ou les contours des objets manipulés⁴. De même que les objets graphiques sont envisagés comme des volumes géométriques ayant des propriétés visuelles de surface, de forme, de transparence ou de réflexion, les objets manipulés sont considérés comme des volumes géométriques dont il faut rendre à l'utilisateur les propriétés de surface (élasticité d'interaction, friction, texture) par le biais d'une interface haptique [Ruspini *et al.*, 1997]. L'architecture logicielle et matérielle utilisée reflète le parallélisme conceptuel entre représentation visuelle et représentation haptique. Un moteur de simulation est chargé de calculer le comportement de l'objet ou de l'environnement virtuel en temps réel, puis des couches logicielles spécifiques se chargent de prendre en charge les spécificités de chaque modalité : génération de la représentation graphique, le plus souvent sur une unité de calcul dédiée (GPU) pour la visualisation, et gestion de l'interaction mécanique entre l'objet simulé et l'utilisateur par un processus dédié (figure 1.6).

Le schéma fonctionnel général des architectures de Réalités Virtuelles de type 'haptic rendering' est représenté figure 1.6. Nous avons choisi dans notre représentation le fonctionnement du

⁴L'utilisation du terme haptic rendering est révélatrice du positionnement de ce domaine de recherche quant à l'utilisation des systèmes haptiques (encore appelés haptic displays) : le système haptique est utilisé pour représenter un objet pour la main, comme un écran CRT représente un objet pour les yeux.

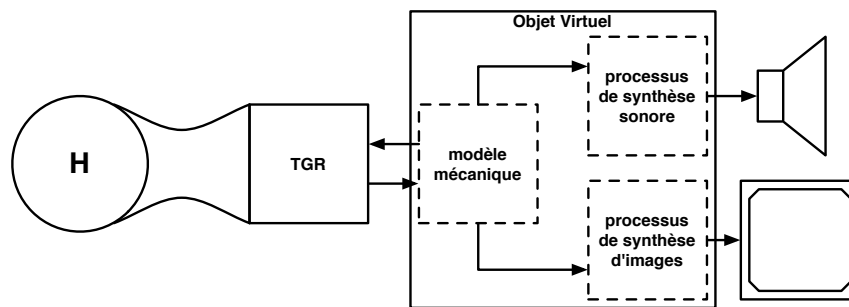


Figure 1.6. Représentation synthétique d'une architecture classique pour la Réalité Virtuelle

système haptique en mode impédance⁵, cas le plus fréquemment rencontré dans les domaines d'application concernés.

1. Le modèle mécanique assure la connexion entre le système haptique et le modèle virtuel calculé par le moteur de simulation. Il calcule l'effort que le système haptique doit retourner en fonction de sa position courante. Généralement, le modèle mécanique est exécuté à une fréquence de calcul entre 1 et 3 kHz, et fonctionne de manière synchrone avec les flux d'entrée et de sortie du système haptique.
2. Le moteur de simulation procède au calcul du comportement de tout le modèle virtuel, excepté l'avatar. Le modèle de simulation est basé sur un modèle physique, et peut-être calculé à des fréquences très variables suivant le type de modèle (entre 30 et 1 000 Hz, suivant la complexité du modèle et les contraintes dynamiques). Ce processus est soumis à des contraintes de temps-réel mou le plus souvent, et n'est pas synchronisé sur le processus calculant le modèle mécanique.
3. Des processus spécifiques sont utilisés pour les modalités visuelles et auditives. Ils ne sont pas synchronisés avec le moteur de simulation, mais doivent fournir un débit d'information adapté à chaque modalité utilisée dans l'application : le processus de synthèse graphique fonctionne généralement à une fréquence de 30 Hz ou plus, le processus de synthèse sonore à quelques centaines de Hertz [Sinclair et Wanderley, 2007]. De même que dans le cas de l'implémentation de type mapping, les informations envoyées par le moteur de simulation à destination de ces flux peut être de nature phénoménologique ou événementielle. En particulier, étant donné la puissance de calcul demandée pour pouvoir générer (moteur de simulation) puis traiter (processus de synthèse sonore) un flux de données à la fréquence audio (44,1 kHz), les processus de synthèse sonore reçoivent des données de type événementiel, puis ces données sont utilisées pour générer un signal sonore. Par exemple, le moteur de simulation extrait d'un contact entre deux objets des informations sur l'instant de la collision et la vitesse de collision. Le processus de synthèse sonore utilise ces informations soit en entrée d'un modèle physique de percussion, soit pour le déclenchement d'échantillons

⁵En mode impédance, à chaque pas de calcul le système haptique envoie un échantillon de position par degré de liberté au processus de simulation ; les valeurs de position sont obtenues des capteurs de mouvement du système haptique. À la fin du calcul du pas de simulation, le processus de simulation renvoie un échantillon force par degré de liberté au système haptique ; les valeurs de force sont fournies en entrée des actionneurs du système haptique (cf. annexe D).

En mode admittance, le système haptique génère un signal de force et reçoit un signal de position.

pré-enregistrés (l'information de vitesse étant alors utilisée pour moduler l'amplitude ou le temps d'amortissement du son) [de Götzen *et al.*, 2007, de Götzen et Marogna, 2007].

Nous présenterons de manière plus approfondie cette architecture dans le chapitre 3. Observons simplement pour le moment que ces différents composants fonctionnent de manière asynchrone les uns par rapport aux autres, exception faite du modèle mécanique, généralement synchronisé sur les entrées/sorties du système haptique.

Cette implémentation présente plusieurs avantages :

- Chaque processus travaille à une fréquence de calcul adaptée à la modalité concernée. On considère généralement qu'une fréquence de rafraîchissement entre 30 et 50 Hz est suffisante pour le rafraîchissement des informations visuelles, et que le délai dans la boucle geste-son n'est pas perçu s'il est inférieur à quelques dizaines de ms (soit une fréquence de fonctionnement de quelque centaines de Hertz si les délais de communication entre les processus sont relativement courts).
- Cette structure relativement modulaire permet d'adapter les besoins de la simulation à la puissance de calcul du simulateur : il est par exemple possible de réduire la fréquence de fonctionnement d'un processus de calcul (avec les conséquences que cela implique) au profit d'un autre processus, généralement le moteur de simulation, jugé plus important ou plus gourmand en ressources.
- Elle permet une intégration aisée dans les systèmes d'exploitation courants (Windows XP, Mac OSX, Linux), qui ne sont pas conçus pour supporter les contraintes spécifiques des applications en temps réel dur [Castagné *et al.*, 2005].

Mais présente également des inconvénients majeurs :

- Le paradigme implicite à ce type d'architecture de simulation est orienté vers la superposition ou la combinaison de différentes modalités plus que vers la considération de l'objet simulé comme une seule et même entité phénoménologique. L'objet virtuel est construit par la superposition de plusieurs processus de simulation, chacun étant adapté à une modalité sensorielle.
- En outre, la frontière entre les différents processus impliqués dans la simulation n'est pas clairement définie et n'implémente pas systématiquement des flux d'information bidirectionnels entre chaque bloc fonctionnel. Cette architecture ne permet donc pas de respecter la continuité énergétique dans la chaîne instrumentale, particulièrement si les informations échangées entre processus sont réduites à des informations de type événementiel [Couroussé, 2002]. *Cette architecture ne permet donc pas de restituer la fonction ergotique du geste dans la relation homme-ordinateur.*

3.4 Le geste instrumental

L'architecture de calcul présentée figure 1.6 n'est pas satisfaisante pour le jeu instrumental étant donné que la chaîne énergétique ne permet pas l'implémentation d'une ergotique *fonctionnelle* dans l'interaction homme-ordinateur. En effet, la situation précédente demande qu'une interaction gestuelle ergotique s'établisse entre l'homme et l'interface gestuelle, puisqu'un échange d'énergie est nécessaire ne serait-ce que pour actionner les périphériques gestuels, mais nous considérons que la fonction ergotique n'y est pas fonctionnelle puisqu'elle n'est pas en relation avec l'objet virtuel qui se trouve objet de l'interaction.

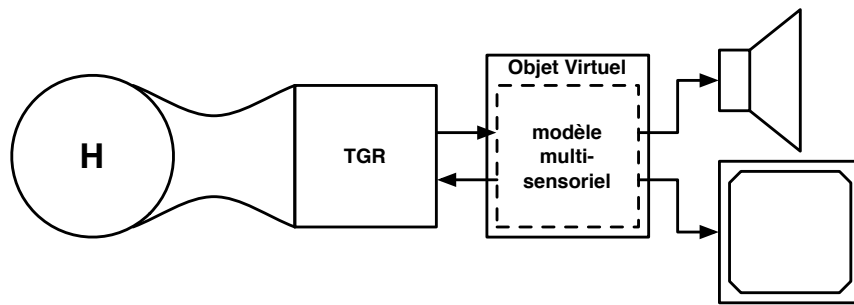


Figure 1.7. Architecture pour la simulation du geste instrumental

Il est des situations où le couplage mécanique fort entre l'homme-instrumentiste et l'objet constitue l'essence même du phénomène qui émerge de l'interaction. Dans le cas du jeu du violon par exemple, le son si particulier de la corde frottée naît du couplage intime entre l'archet tenu dans la main de l'instrumentiste, et de la corde qui vibre [Florens, 2003]. Afin de représenter dans l'ordinateur des situations instrumentales où le geste ergotique est fonctionnel, il faut donc que la représentation de l'objet simulé permette la continuité énergétique entre l'homme et la partie mécanique de l'objet qui sera source du phénomène instrumental.

Pour pouvoir représenter correctement ces situations dans l'ordinateur, nous ne pouvons faire autrement que d'assurer un couplage fort entre chacun des constituants de la chaîne instrumentale, et nous faisons l'hypothèse que ceci n'est réalisable que par l'utilisation d'un seul modèle intégrant l'ensemble de la chaîne instrumentale.

Dans la situation instrumentale, c'est la dynamique du système physique représenté qui est prépondérante. Du point de vue du processus de simulation, cela implique des contraintes au niveau de la fréquence de calcul et de la latence des échanges de données. Les travaux de l'ACROE se trouvent au cœur de cette approche : [Luciani *et al.*, 2005a] présente les critères techniques pour l'implémentation de la chaîne instrumentale sonore dans l'ordinateur, qui est le cas le plus exigeant techniquement pour l'implémentation de la chaîne instrumentale. Nous pouvons étendre ces critères à toutes les situations instrumentales :

- La fréquence de simulation doit permettre de représenter l'intégralité des phénomènes physiques se trouvant dans la bande passante du phénomène considéré. En particulier, si l'objet simulé permet la production de phénomènes vibratoires dans le domaine de fréquence audio, la fréquence de simulation doit permettre la génération du signal sonore. Elle doit donc se trouver entre 10 et 50 kHz dans ce cas.
- La plage dynamique des variables physiques utilisées pour la simulation doit englober la dynamique du phénomène physique simulé. La plage dynamique des objets sonores en particulier est très importante, puisqu'elle comprend les gestes du musicien (de l'ordre de grandeur du mètre) et les mouvements de la structure vibrante (de l'ordre de grandeur de quelques μm), soit environ 1 000 dB.
- Le processus de simulation doit être complètement synchronisé avec les différents périphériques mis en jeu dans la simulation interactive, en particulier les transducteurs (système haptique, écran, haut-parleur). Le déterminisme strict de chaque processus impliqué dans la simulation est la seule manière d'assurer que ces contraintes de latence sur les entrées/sorties seront

respectées : en particulier, le temps d'exécution d'un pas de simulation doit être connu de manière déterministe dans la fenêtre temporelle imposée par les conditions de simulation.

Ce dernier critère signifie qu'il est nécessaire d'implanter la plateforme de simulation sur un système d'exploitation supportant le temps réel dur, afin de respecter les critères de déterminisme et de synchronisation.

4 Conclusion

L'interaction instrumentale est un type particulier d'interaction de l'homme avec son environnement. Elle regroupe les trois fonctions sémiotique, épistémique et ergotique. Dans le cadre de l'interaction gestuelle avec un ordinateur, nous avons présenté plusieurs approches, qui permettent la réintégration de la relation instrumentale naturelle à différents degrés. La stratégie de *mapping* met en jeu la fonction sémiotique, mais la fonction épistémique n'y est pas présente et la fonction ergotique n'est pas en relation avec le processus de synthèse sonore. Pour réintroduire la fonction ergotique dans la relation homme-ordinateur, il est nécessaire d'utiliser le retour d'effort. Dans les techniques de simulation classiques, le choix est fait d'utiliser un modèle différent pour chaque modalité adressée par la simulation. Le retour d'effort permet donc l'interaction avec un objet mécanique simulé, mais il existe toujours une séparation entre la représentation mécanique de l'objet virtuel et sa représentation sonore ou visuelle en raison du parti-pris de la séparation fonctionnelle des sensorialités impliquées dans la simulation. Dans ce cas, la communication entre les composantes mécaniques et sonores d'un même objet virtuel est assurée par le biais d'un support de type signal. Nous faisons alors l'hypothèse que ce n'est que par l'usage d'un modèle unique prenant en compte toutes les modalités sensorielles à adresser qu'il est possible d'implanter de manière complète la relation ergotique multisensorielle telle qu'on la trouve dans la relation entre un homme et son environnement naturel non informatisé. Cette hypothèse pose la fonction ergotique du geste comme un composant nécessaire de la relation instrumentale.

Par ailleurs, la relation ergotique et multisensorielle entre l'homme et son environnement est conceptuellement proche des théories écologiques de la perception ainsi que de la théorie de l'énaction, développées dans les sciences de la cognition. Ces théories ont d'ailleurs été le cadre conceptuel du réseau d'excellence Enactive Interfaces.

La théorie de l'énaction fut proposée par Varela en 1991 [Varela, 1991]. Le mot *enaction* vient du verbe anglais *enact*, c'est-à-dire mettre en pratique (mais aussi être sur scène, ou encore mettre en application une loi). La théorie enactive met en avant le rôle de l'action dans la cognition : la cognition devient un processus où l'être vivant *enacte* le monde dans lequel il vit [Stewart, 2007]. Plutôt qu'une relation duale entre l'être vivant et le monde, l'énaction met l'accent sur un couplage structurel qui co-détermine le système cognitif et son monde :

«La cognition dans son sens le plus vaste consiste plutôt en l'énaction ou le faire-émerger d'un monde par le biais d'une histoire viable de couplage structurel.» Varela, l'inscription corporelle de l'esprit, 1993 [Varela, 1993]

Le monde pour l'organisme n'est alors rien d'autre que les conséquences perceptives de ses actions : non seulement l'approche enactive suppose un couplage structurel fort entre l'organisme et le monde, mais suppose aussi une imbrication étroite entre action et perception :

«Nous voici à présent en mesure de proposer une formulation préliminaire de ce que nous entendons par enaction. En bref, cette approche se compose de deux points : (1) la perception consiste en une action guidée par la perception ; (2) les structures cognitives émergent des schèmes sensori-moteurs récurrents qui permettent à l'action d'être guidée par la perception.» Varela, *l'inscription corporelle de l'esprit*, 1993 [Varela, 1993]

La thèse que nous présentons dans ce document est construite autour de ces deux hypothèses principales fondatrices de l'enaction. Ainsi, les analyses sur les systèmes technologiques actuels que nous avons proposées dans le chapitre 3 pointent leurs difficultés à rendre compte des propriétés enactives des systèmes et des situations. Puis les technologies que nous avons développées, décrites dans les chapitres 4 et 5 l'ont été pour précisément tenter de les atteindre. Enfin, nous verrons, dans le chapitre 6 que ces outils technologiques nous ont permis de mettre en évidence expérimentalement plusieurs de ces propriétés, émergence, présence, organicité de l'interaction, apprentissage, adaptivité, etc.

Chapitre 2

Analyse Fonctionnelle des Transducteurs Gestuels Rétroactifs

Sommaire du Chapitre

1	Introduction	22
2	La controverse du terme « système haptique »	22
3	Le choix de l'approche fonctionnelle	23
4	Méthodologie	24
5	Classification des approches fonctionnelles	26
5.1	Approche « objet »	26
5.1.1	Définition	26
5.1.2	Le TGR générateur d'impédance mécanique	26
5.1.3	Transparence	27
5.1.4	Z-Width et passivité	27
5.1.5	Téléopération immersive	28
5.1.6	Bras exosquelettes	28
5.1.7	Gants exosquelettes	29
5.1.8	CAVEs	30
5.2	Approche « homme »	31
5.3	Approche « interaction »	34
5.3.1	Définition	34
5.3.2	Enaction, « interface enactive »	35
5.4	Approche « tâche »	36
5.4.1	Introduction	36
5.4.2	Ligne de mobilité	36
5.4.3	Limite de l'environnement virtuel	37
5.4.4	Situation #a. Interaction générale avec un environnement	38
5.4.5	Situation #b. Outil	39
5.4.6	Situation #c. Commande	40
5.4.7	Situation #d. Objet	42
6	Conclusion	46

1 Introduction

L'objectif de ce chapitre est double : il s'agit en premier lieu de dresser un état des lieux des positions théoriques et de l'état de l'art pour l'interaction gestuelle. Mais il s'agit aussi dans un même temps de mettre en évidence le fait que la plupart des travaux de recherche effectués dans ce domaine ne permettent pas de supporter l'approche enactive. En effet, nous verrons que l'approche enactive est supportée par la catégorie fonctionnelle que nous appelons « interaction », alors la grande majorité des travaux appartiennent aux catégories fonctionnelles « objet » et « homme », qui sont centrées sur un seul des deux acteurs en présence dans l'interaction homme-objet.

Ce chapitre constitue un état de l'art de la technologie pour le retour d'effort. Nous avons choisi un parcours original, basé sur l'analyse fonctionnelle de ces systèmes. À partir de cette analyse, nous identifions quatre catégories fonctionnelles : approche « objet », approche « homme », approche « interaction » et approche « tâche ». Chaque catégorie fonctionnelle est illustrée par la présentation succincte de systèmes exemplaires.

L'analyse fonctionnelle que nous proposons dans ce chapitre constitue aussi une méthode d'analyse du couplage homme-objet dans la situation naturelle — i.e. interaction avec le monde réel — et dans la situation médiatisée — i.e. interaction avec un objet ou un environnement virtuel par l'intermédiaire d'un système à retour d'effort. Ce parcours se veut complet du point de vue de l'analyse fonctionnelle qui y est présentée et de la méthode, mais ne présente pas un contenu exhaustif des travaux effectués dans le domaine de l'haptique aujourd'hui. Pour obtenir une vue plus étoffée de l'état de l'art, on pourra se reporter avec profit au livrable D3.1 : « Report on the State of the Art on existing sensors/actuators technologies for haptic interfaces » [Jean-Loup Florens, 2004] produit par le réseau Enactive Interfaces.

En particulier, notre analyse met en avant le découplage fonctionnel qui a lieu dans la plupart des approches pour l'interaction gestuelle, alors que dans le monde physique, la manipulation d'un objet passe par un couplage mécanique fort, qui, nous le verrons plus loin, peut conduire à la modification des propriétés physiques des deux systèmes en jeu (d'un côté, l'humain, de l'autre, l'objet manipulé). Seule l'approche fonctionnelle « interaction » que nous présentons dans ce chapitre est centrée sur la mise en oeuvre du couplage. C'est cette approche fonctionnelle qui nous intéressera plus particulièrement dans le cadre de cette thèse.

2 La controverse du terme « système haptique »

Dans la suite de ce document, nous utiliserons le terme TGR (Transducteur Gestuel Rétroactif), employé dès les années 1980 par les chercheurs de l'ACROE [Cadoz *et al.*, 1988], au lieu du terme courant « système haptique » ou encore « interface haptique ».

Le terme « haptique » est en effet couramment employé avec des significations diverses et parfois contradictoires. Le plus souvent, « haptique » est synonyme du sens du toucher au sens large. Si nous nous en référons au contenu de la page de Wikipedia en version française (juin 2008) : « haptique désigne la science du toucher [...] (provient du grec "haptomai" qui signifie "je touche"). Au sens strict, l'haptique englobe le toucher et les phénomènes kinesthésiques, i.e. la

perception du corps dans l'environnement. » mais nous pouvons également lire plus loin : « *Il est classique de différencier deux types de perception tactile manuelle [...] : la perception cutanée et la perception haptique.* » puis encore : « *La perception tactilo-kinesthésique ou haptique [...] ou active résulte de la stimulation de la peau résultant des mouvements actifs d'exploration de la main entrant en contact avec des objets.* ». Il semble donc que le terme haptique désigne la perception kinesthésique, et que ce terme englobe parfois la perception tactile, parfois non.

De même, le terme « système » est général et possède une signification floue de l'objet que nous souhaitons désigner. Le terme « interface » semblerait plus approprié puisqu'il rend explicite le fait que le TGR se situe entre deux entités : à savoir d'un côté l'homme, de l'autre l'ordinateur.

Pour ces raisons, nous préférons aux termes « système haptique » ou « interface haptique » le terme « TGR », acronyme de « transducteur gestuel rétroactif » qui a le mérite de rendre explicite l'objet de notre propos. Le terme « transducteur » désigne un objet dont la fonction première est la transformation de signaux en phénomènes sensibles et inversement (un écran ou un haut-parleur sont donc des transducteurs). Deuxièmement ce transducteur est adapté au geste, nous le qualifions donc de « gestuel ». Enfin, ce transducteur utilise la technologie du « retour d'effort », qui a de particulier d'être le lieu d'une boucle de rétroaction alors que les transducteurs-afficheurs pour la vision et les transducteurs-haut-parleurs pour l'audition sont utilisés en boucle ouverte. Nous le qualifions donc de rétroactif.

3 Le choix de l'approche fonctionnelle

Tenter de proposer une catégorisation des systèmes conçus pour l'interaction homme-machine est une tâche complexe pour plusieurs raisons. D'autre part, le geste humain est éminemment versatile [Burdea, 1996, Luciani *et al.*, 2006a], et les utilisations du geste dans la vie courante sont extrêmement variées : depuis les mouvements impliquant tout le corps comme les activités sportives aux tâches de précision micrométrique comme l'orfèvrerie. Cette versatilité du geste humain ainsi que la grande variété des utilisations du geste dans les activités humaines ne peuvent pas être réunies dans la conception d'un seul TGR générique : il existe actuellement tout un ensemble de systèmes de conceptions diverses afin de répondre aux spécificités de certains gestes, ou de certaines applications. D'autre part, au regard du nombre de nouvelles applications des TGR et le nombre de nouveaux prototypes apparaissant chaque année, il apparaît que le domaine de la recherche pour les réalités virtuelles et le retour d'effort est actuellement en pleine ébullition. Aussi, si l'on considère la richesse des origines historiques des TGR ainsi que la versatilité du geste humain et l'extrême variété des applications possibles ou existant, il apparaît que la catégorisation des systèmes haptiques est une tâche relativement complexe.

On trouve aujourd'hui de nombreuses références proposant un état de l'art des systèmes haptiques. La méthode de classification est généralement construite sur deux axes orthogonaux. Le premier axe de classification s'appuie sur les caractéristiques techniques des TGR (de table, exosquelettes, technologie pour l'actuation et la capture de mouvement, etc.). Le second axe de classification s'appuie sur les champs d'application du retour d'effort : téléopération, prototypage industriel, création artistique et arts numériques, etc.

Les états de l'art généralistes sont construits sur ces deux axes : technologique et applicatif. Citons par exemple l'état de l'art de Grigore Burdea [Burdea et Coiffet, 1993, Burdea, 1996].

Certains états de l'art sont spécifiques à un champ d'application particulier : robotique [Burdea, 1999], téléopération [Jean-Loup Florens, 2004, Sheridan, 1992], infographie [Laycock et Day, 2003], informatique musicale et instruments électroniques [O'Modhrain, 2000], etc.

La classification des systèmes haptiques est parfois mise en relation avec les caractéristiques du geste humain, par exemple pour la manipulation experte [Jones, 1997], ou bien à partir de considérations historiques [Stone, 2000].

Hayward et Astley [Hayward et Astley, 1996] proposent un schéma d'évaluation des performances des systèmes haptiques selon plusieurs critères : nombre de degrés de liberté, espace de travail, inertie du point manipulé, forces de friction, capacité en force des actionneurs, bande passante, etc.

Nous partons du principe que le TGR joue le rôle d'une interface pour un être humain interagissant avec un objet ou plus généralement avec son environnement, que ce dernier soit réel ou virtuel. Suivant l'application visée ou les choix de conception, le TGR peut occuper une fonction différente dans l'interaction homme-objet. Inversement, les critères de conception d'un nouveau système ou de mise en œuvre d'une nouvelle application sont relativement différents suivant le rôle fonctionnel qu'est censé jouer le TGR dans la relation homme-objet.

Cependant, dans les différentes classifications des TGR et les états de l'art du domaine que nous avons pu consulter, les aspects fonctionnels des TGR et leur utilisation sont rarement considérés. À notre connaissance, seule la présentation du FEELEX par Iwata et al. en 1997 [Iwata *et al.*, 2001] présente une ébauche de classification des TGR sur la base d'aspects fonctionnels. Trois catégories fonctionnelles sont présentées :

- Les exosquelettes
- Les TGR basés sur la métaphore de l'outil
- Les TGR basés sur la métaphore de l'objet ; c'est dans cette catégorie qu'est classé le FEELEX
- En outre, plusieurs TGR ne sont pas inclus dans cette catégorisation, comme les systèmes tactiles, ou les systèmes passifs basés sur des capteurs de force.

Ce chapitre présente donc un aperçu de la variété des TGR, des positionnements théoriques et des applications qui y sont attachés sur la base d'une approche fonctionnelle.

4 Méthodologie

Il est possible d'aborder les aspects fonctionnels des TGR à différents niveaux :

1. Au niveau théorique
2. Dans la conception de systèmes particuliers
3. Au niveau applicatif

Dans notre analyse, nous avons tenté d'identifier les aspects fonctionnels explicites ou implicites dans les travaux du domaine haptique à ces différents niveaux.

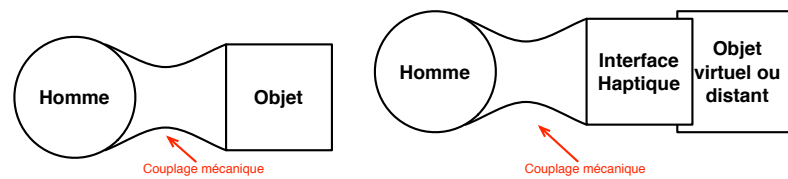


Figure 2.1. Schémas d'interaction homme-objet dans la situation naturelle (à gauche) et médiatisée (à droite)

Nous considérons alors deux cas types de l'interaction homme-objet¹ : la situation naturelle et la situation médiatisée. La situation naturelle est la situation de l'interaction de l'homme avec son environnement naturel, c'est-à-dire le monde physique de tous les jours. Dans cette situation, l'humain interagit avec un objet ou un environnement par l'intermédiaire du couplage mécanique. La situation médiatisée fait intervenir l'ordinateur, jouant le rôle de medium entre l'homme et l'objet manipulé. Cet objet peut ne plus exister que dans l'ordinateur, sans autre preuve matérielle de son existence que des électrons s'agitant dans les circuits imprimés ! Dans la situation médiatisée, c'est l'interface haptique qui permet la réintroduction du couplage mécanique entre l'utilisateur et l'objet virtuel simulé² (figure 2.1).

Notre méthodologie se base sur la comparaison du schéma d'interaction homme-objet dans la situation naturelle avec la situation médiatisée. La situation médiatisée est la situation mettant en jeu l'humain en situation d'interaction, mais où le TGR joue le rôle de medium avec l'objet de l'interaction.

Nous pouvons considérer deux types de situations médiatisées :

1. L'objet manipulé est un objet virtuel.
2. Dans le cas de la téléopération, l'objet manipulé existe réellement, mais se trouve hors de portée de l'homme (soit en raison d'une distance trop grande, soit en raison d'une barrière physique empêchant la manipulation directe, soit en raison d'un changement d'échelle spatiale).

Du point de vue du couplage entre l'homme et l'objet manipulé, cette situation est identique à la manipulation d'un objet virtuel : dans les deux cas c'est le TGR qui joue le rôle d'interface pour la manipulation.

¹Le terme d'objet comprend ici toutes les entités pouvant se trouver en situation d'interaction avec un humain, depuis ce qui est manipulé dans les tâches de manipulation fine (le sens premier d'objet) jusqu'à l'environnement dans son sens le plus général.

²Dans la situation médiatisée, C. Cadoz parle plus volontiers de *représentation* du couplage mécanique [Cadoz, 1994].

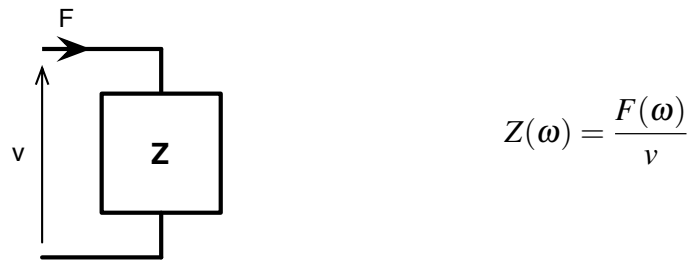


Figure 2.2. L'impédance mécanique comme modèle de l'objet

5 Classification des approches fonctionnelles

5.1 Approche « objet »

5.1.1 Définition

Dans l'approche objet, l'observation se porte sur les propriétés de l'objet naturel ou simulé indépendamment des propriétés de l'humain impliqué dans le processus d'interaction. L'hypothèse est faite que le couplage mécanique entre l'homme et l'objet peut être coupé afin d'observer indépendamment l'objet. En conséquence, l'objet est considéré comme une entité séparable de l'homme, et il est observé à son point de couplage (tableau 2.1 #1, page 33).

Dans la situation médiatisée, l'interface haptique est considérée la plupart du temps comme partie intégrante de l'objet. Le point d'observation se situe donc en deçà, au point de couplage mécanique avec l'homme.

5.1.2 Le TGR générateur d'impédance mécanique

L'approche fonctionnelle « objet » part du principe que l'objet de l'interaction peut être observé indépendamment de l'homme. Dans ce cas, un modèle couramment pratiqué est le modèle d'impédance : du point d'observation choisi, l'objet observé est réduit à un dipôle mécanique (F, v) caractérisé par son impédance Z (figure 2.2). Le TGR est alors considéré comme un générateur d'impédances mécaniques :

« A haptic interface may be thought of as a device which generates mechanical impedances. Impedance, here, should be understood to represent a dynamic (history-dependent) relationship between velocity and force » Colgate and Brown, 1994 [Colgate et Brown, 1994]

Dans cette catégorie fonctionnelle, nous trouvons la plupart des analyses qui tentent de délimiter les conditions d'un fonctionnement acceptable pour l'interface haptique ou pour le simulateur haptique (c'est-à-dire, l'interface haptique couplée au système de calcul des objets virtuels).

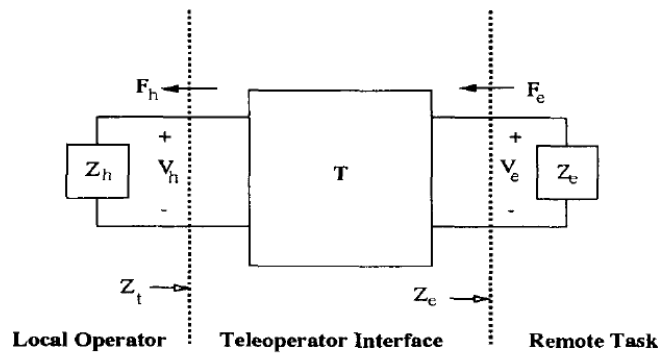


Figure 2.3. L'interface haptique vue comme un quadripôle dans un système de téléopération — extrait de [Lawrence, 1993]

5.1.3 Transparence

La théorie de la transparence de D. Lawrence s'applique essentiellement aux systèmes de téléopération [Lawrence, 1993]. Dans la chaîne de téléopération, l'impédance mécanique de l'objet distant, Z_e , est transmise à l'opérateur par l'interface de téléopération, c'est-à-dire l'interface haptique. Au travers du téléopérateur, l'impédance de l'objet distant devient Z_t , impédance mécanique de toute la chaîne de téléopération (Figure 2.3). Z_t dépend des caractéristiques mécaniques de l'objet distant (donc de son impédance Z_e), mais aussi des caractéristiques propres de l'interface haptique. Lawrence définit le critère de transparence de la manière suivante :

“The notion of haptic transparency is used to quantify the fidelity with which virtual object properties are presented to, and perceived by, the human operator”
Lawrence et al., 1996 [Lawrence et al., 1996]

Ce critère revient à dire que lorsque la condition de transparence est parfaitement réalisée, l'objet mécanique distant est perçu au travers de l'interface haptique de la même manière que dans la situation d'interaction directe, c'est-à-dire : $Z_t = Z_e$. Le critère de transparence peut aussi s'appliquer à la simulation d'objets ou d'environnement virtuels : il revient à définir les conditions pour lesquelles l'interface haptique déforme le moins possible la représentation de l'objet virtuel calculée par le simulateur.

5.1.4 Z-Width et passivité

L'approche du paradigme du générateur d'impédance proposée par Colgate est relativement différente de l'approche de la transparence proposée par Lawrence : l'un des problèmes fondamentaux des systèmes haptiques réside dans le fait de pouvoir générer une large plage d'impédances. En effet, le TGR idéal doit pouvoir générer à la fois (1) des impédances nulles, correspondant au mouvement libre, et (2) des impédances infinies, correspondant à un objet infiniment rigide. Colgate et Brown définissent alors le terme « Z-Width », correspondant à l'amplitude de la plage d'impédances que l'on peut effectivement obtenir avec un système donné [Colgate et Brown, 1994].

En outre, E. Colgate fait l'observation que, à l'exception des êtres vivants, tous les objets réels sont passifs. En conséquence, l'interface haptique doit elle aussi présenter un comportement inconditionnellement passif³. L'approche de Colgate et Brown est alors de concevoir des systèmes haptiques présentant la meilleure plage d'impédance possible, avec la condition que l'interface haptique se comporte de manière inconditionnellement passive. Colgate et Brown se basent sur l'implémentation d'impédances mécaniques présentant un comportement passif, avec des critères d'observation statiques. Ils montrent en outre que l'augmentation de la viscosité mécanique propre au TGR permet d'augmenter l'impédance mécanique maximale atteignable, tout en conservant la passivité des contacts ; afin de conserver une impédance faible en mouvement libre, une impédance négative vient compenser la viscosité mécanique du TGR.

5.1.5 Téléopération immersive

Historiquement, les premières limitations des applications de la robotique furent liées à la conception des effecteurs. Les terminaisons sont le plus souvent des pinces constituées de deux parties à mouvement parallèle présentant un faible intérêt pour la préhension d'objets, la manipulation dextre ou pour obtenir des informations rhéologiques sur l'objet manipulé. La première amélioration apportée fut de doter les pinces de plusieurs doigts, à l'image de la main humaine (multi-fingered robotics hands) [Shimoga, 1992]. Cependant, lorsque le modèle de l'environnement de la tâche est peu précis, lorsqu'il évolue rapidement ou lorsque la tâche demande une manipulation délicate ou des mouvements complexes (par exemple en chirurgie), l'opérateur humain doit être maintenu dans la chaîne de la tâche robotique [Burdea, 1999].

En téléopération immersive, l'opérateur humain est complètement immergé dans l'environnement de la tâche [Shimoga, 1992, Shimoga et Khosla, 1994]. Dans ce cas, un retour d'effort est appliqué à tous les membres de l'opérateur à l'aide d'un exosquelette, afin de réduire les temps de complétion de la tâche, la consommation d'énergie, le nombre d'erreurs et l'amplitude des forces de contact [Shimoga, 1992].

5.1.6 Bras exosquelettes

Arm Exoskeleton est un exosquelette développé par Bergamasco et al. au laboratoire PERCRO [Bergamasco et al., 1994], dans le cadre du projet européen PURE FORM. Le but de ce projet était de permettre à des aveugles ou à des personnes visuellement déficientes d'interagir avec un environnement virtuel via le toucher. Le résultat principal de ce projet fut la réalisation d'un « musée virtuel », où les visiteurs peuvent toucher les statues présentées grâce à des exosquelettes et un système de vision 3D. Cet exosquelette procure 4 DDL et est conçu pour les mouvements du bras de l'utilisateur. Le système fut conçu pour être combiné avec *Hand-Exos*, un exosquelette conçu pour deux doigts de la main. *Hand-Exos* procure 2×3 DDL. Ces deux systèmes reposent sur le sol, ce qui évite de faire porter le poids des actionneurs à l'opérateur du système.

Le *MasterArm Pneumatic Haptic Interface* est un exosquelette développé au laboratoire Systems Laboratory, Southern Methodist University. La structure de l'exosquelette, en aluminium, est fixée sur un fauteuil et se termine par une poignée tenue en main par l'opérateur. Le système mesure les

³Le critère de passivité est plus contraignant que le critère de stabilité.



Figure 2.4. Le musée PURE FORM développé dans le cadre du projet européen PURE FORM (www.pureform.org)

positions de l'épaule et du coude droits, et des pistons pneumatiques permettent d'actionner les 4 degrés de liberté [Hurmuzlu *et al.*, 1998].

5.1.7 Gants exosquelettes

À partir des années 1980, plusieurs systèmes furent conçus pour immerger la main entière de l'opérateur dans l'environnement de la tâche : on parle communément de *gants*. Le premier brevet de gant conçu pour un usage immersif en téléopération fut déposé par Zarudianski en 1981 [Zarudiansky, 1981]. Ce système était spécialement conçu pour des applications de téléopération. Une structure extérieure rigide maintenait un gant souple dans lequel l'utilisateur insérait sa main. Les actionneurs étaient prévus pour être placés dans la structure de l'exosquelette, procurant un retour d'effort indépendant pour chaque doigt, la paume et le poignet. Cependant, le brevet est resté à l'état de projet et n'a pas donné lieu à réalisation.

Les premiers gants immersifs furent réalisés par Iwata et al. à l'université de Tsukuba, puis par Bouzit et Coiffet au LRP⁴ [Burdea, 1996]. Le gant de Iwata procurait un retour d'effort pour deux doigts. Ce système utilisait une structure d'exosquelette rigide, les actionneurs, des moteurs DC, étant situés sur l'avant-bras, et reliés au mouvement des doigts par des câbles. Le *LRP Hand Master* (1993) était conçu pour fournir un retour d'effort aux cinq doigts de la main à 14 emplacements différents. Le système utilisait aussi une structure rigide d'exosquelette et la transmission de la cinématique des actionneurs était assurée par des câbles. Les mesures de position étaient effectuées par des potentiomètres placés sur les arbres des moteurs DC.

Un des problèmes majeurs dans la conception d'exosquelettes concerne l'actuation. En effet, les actionneurs sont les composants les plus lourds d'un TGR, et il est nécessaire qu'ils soient situés à proximité de la zone sur laquelle doivent s'exercer les efforts. D'un côté, la distance entre les actionneurs et la zone où doivent s'exercer les efforts ne doit pas être trop importante afin de minimiser les frictions et le jeu générés par le système de transmission de mouvement qui sont proportionnels à la distance à parcourir. Inversement, placer les actionneurs à proximité de la zone

⁴Laboratoire de Robotique de Paris

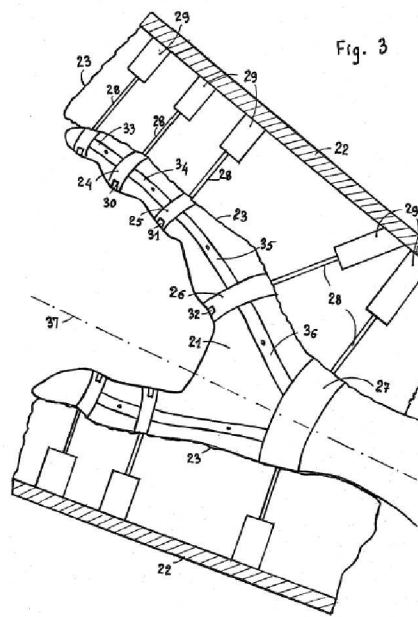


Figure 2.5. Schéma du « nonportable dextrous master » breveté par Zarudiansky [Zarudiansky, 1981]

où les efforts doivent s'exercer signifie que le poids des actionneurs sera supporté par l'utilisateur (sur l'avant-bras ou sur les épaules). La conception d'un exosquelette consiste donc à trouver un compromis entre la charge de la structure qui sera supportée par l'utilisateur, et la distance nécessaire pour la transmission du mouvement.

Le "Rutgers Master" est un gant exosquelette qui fut conçu pour résoudre les problèmes liés au poids des actionneurs et aux problèmes de transmission de mouvement. Le système d'actuation est constitué de pistons pneumatiques, présentant un rapport poids/puissance intéressant, alimentés en air sous pression par le biais de petits câbles flexibles. Les actionneurs sont placés à l'intérieur de la main de l'utilisateur, en contact direct avec la partie à actionner. L'ensemble du système pèse seulement 105 g, et permet d'appliquer jusqu'à 16 N pour chaque doigt (figure 2.6).

Immersion Incorporation [imm, 2008] a commencé en 1993 le développement d'interfaces haptiques destinées à la commercialisation. Le CyberGrasp est un exosquelette pour la main. Il est composé d'une armature rigide légère (350 g), et la transmission de mouvement s'effectue par l'intermédiaire de câbles, les actionneurs étant placés à la base du système, en dehors de la structure portée par l'utilisateur. Le retour d'effort est appliqué aux cinq doigts de la main indépendamment, jusqu'à 12 N, et la capture des mouvements de la main s'effectue sur 18 à 22 points de mesure en fonction des versions du système. Ce gant à retour d'effort est conçu pour être utilisé en combinaison avec le gant tactile CyberGrasp (Figure 2.7).

5.1.8 CAVEs

Les systèmes appelés « CAVEs » ne sont pas des interfaces haptiques. Il s'agit de systèmes destinés à immerger visuellement et auditivement un utilisateur dans un environnement virtuel (figure 2.8). Ces plateformes virtuelles sont cependant parfois utilisées en combinaison avec des interfaces



Figure 2.6. Les gants Rutgers Master II (à gauche) et Rutgers Master II-New Design (à droite) – University of Rutgers



Figure 2.7. Le gant à retour d'effort CyberGrasp (à gauche), et le gant tactile CyberGlove (à droite) – Immersion Inc.

haptiques. D'un point de vue fonctionnel, les CAVes sont conçus pour immerger l'utilisateur dans un environnement virtuel 3D.

5.2 Approche « homme »

Dans cette approche fonctionnelle, l'observation se porte sur les propriétés de l'homme dans la situation d'interaction avec un objet ou un environnement (tableau 2.1 #2, page 33). Cette approche est le symétrique de l'approche objet, et de la même manière l'hypothèse faite est que le couplage mécanique homme-objet peut être séparé pour observer l'homme indépendamment de l'objet.

Dans la situation médiatisée, la fonction de l'interface haptique est de transmettre les actions de l'homme à l'objet.

Comme dans l'approche objet, le point d'observation se trouve à la frontière entre l'homme et l'objet. Dans l'approche objet, l'observation est essentiellement basée sur l'extraction d'invariants tels que la raideur, la viscosité ou la masse de l'objet. De la même manière, dans l'approche fonctionnelle « homme », il est possible d'observer l'homme sur ces mêmes critères, mais l'homme est un objet d'observation beaucoup plus complexe parce qu'il est composé de nombreux degrés de liberté, et qu'il est actif. Cette différence fondamentale avec les objets communs rend



Figure 2.8. Un environnement de type « CAVE » – EPFL (source : Deliverable RD2.2 Enactive Interfaces [ENACTIVE, 2007b])

l'étude beaucoup plus complexe, et rend l'extraction d'invariants similaires à la masse ou à la raideur extrêmement difficile. En conséquence, l'approche « homme » porte essentiellement sur des données événementielles ou phénoménologiques. Cette approche fonctionnelle s'apparente aux champs de recherche dont l'objet d'étude est l'homme : par exemple les sciences cognitives, la psychophysique, etc.

Dans cette approche fonctionnelle, on peut distinguer deux positions :

1. La situation naturelle homme-objet sert de référence pour l'étude de l'homme, et la situation médiatisée est considérée comme un outil pour obtenir de nouveaux stimuli expérimentaux qui seraient difficiles voire impossibles à obtenir avec des objets réels. La plupart des études psychophysiques sur la perception humaine entrent dans cette catégorie. On citera à titre d'exemple les travaux de Tan et al. sur la perception manuelle des propriétés rhéologiques des objets grâce à un TGR à un seul DDL permettant de simuler différentes valeurs de raideur ou de viscosité [Tan *et al.*, 1992, Tan *et al.*, 1993, Tan *et al.*, 1994, Tan *et al.*, 1995], ou encore les études de la posture par Smart, Stoffregen et Bardy, utilisant une plateforme mobile afin de soumettre les sujets de l'expérience à des stimuli proprioceptifs particuliers [Smart *et al.*, 2002].
2. La situation médiatisée est une nouvelle situation d'interaction homme-objet, et cette situation devient un objet d'étude à part entière : dans ce cas, l'homme est étudié de la perspective d'une interaction médiatisée, sans référence nécessaire à la situation naturelle. On peut citer par exemple les travaux de Lécuyer sur les « illusions sensorielles » et le retour « pseudo-haptique » [Lécuyer *et al.*, 2001, Lécuyer, 2001, Lécuyer *et al.*, 2003], ou encore les études sur la sensibilité de la perception aux délais dans les boucles sensori-motrices [Ellis *et al.*, 2004, Ellis *et al.*, 1999, Mania *et al.*, 2004, Mäki-Patola et Hämäläinen, 2004].

La conception de TGR spécifiques reste relativement marginale à cette approche fonctionnelle malgré le fait que les travaux de recherche sur la perception humaine utilisent de plus en plus les TGR. Dans cette approche, on se place en utilisateur des systèmes et on a recours à des

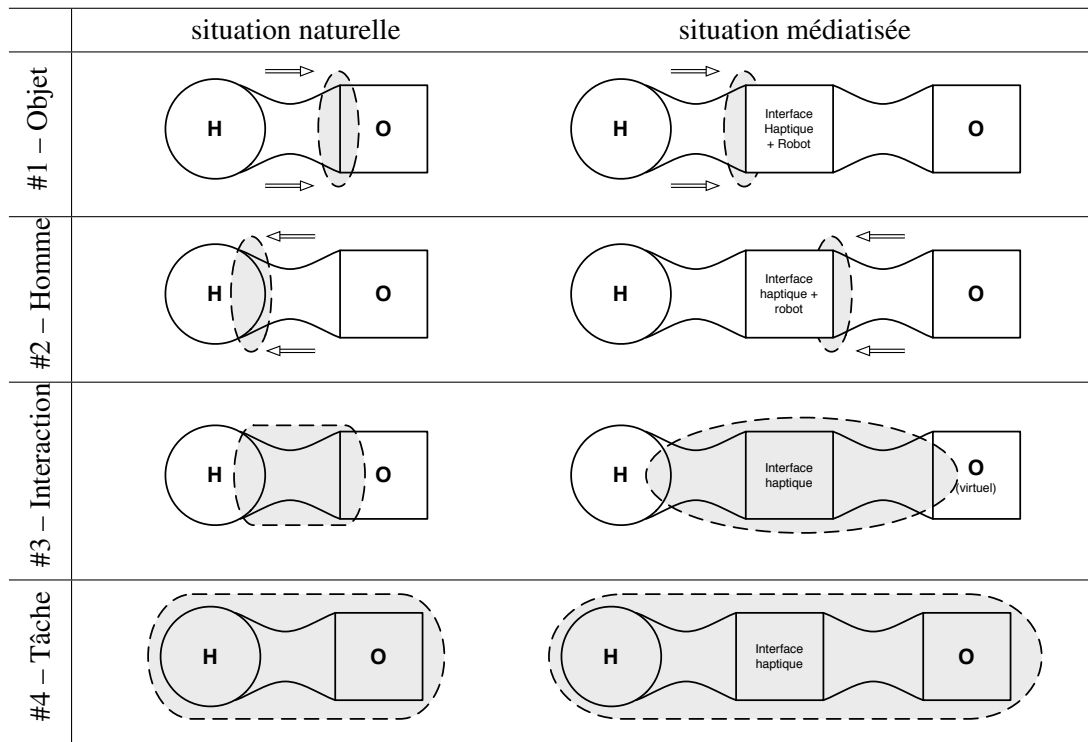


Tableau 2.1. Représentation schématique des quatre approches fonctionnelles de l'interaction homme-objet

systèmes commerciaux, ou à des prototypes dont la destination première est souvent détournée. Certaines équipes de recherche en psychophysique ont parfois eu recours au développement de périphériques gestuels spécialisés afin de mener certaines expériences sur la perception haptique. Nous citerons les travaux de Hong Tan et de son équipe : de nombreuses expériences ont été menées de 1991 à 1995 [Tan *et al.*, 1995] sur la perception des propriétés rhéologiques à partir d'un système électromagnétique, le « hand function tester » [Pang *et al.*, 1991]. Le système était équipé de deux plaques en aluminium destinées à être pincées par le sujet entre le pouce et l'index, actionnées par un moteur DC. La force exercée par le sujet était mesurée à l'aide d'une jauge de contrainte, la position de la partie mobile par un FLDT, et la vitesse par un capteur à induction.

Les systèmes de capture de mouvement⁵ sont aussi largement utilisés dans les études de l'homme. Ces systèmes permettent simplement la mesure des mouvements du corps humain par l'intermédiaire de capteurs, les *marqueurs*, fixés sur le corps. L'utilisation des systèmes de capture est le plus souvent destinée à l'observation de l'homme en situation d'interaction sans contact avec un objet, ou bien en situation d'interaction naturelle. Dans ce cas, le système de capture de mouvement a la fonction d'un système de mesure conventionnel, au lieu de s'interposer entre l'homme et l'objet comme dans le cas d'une interface haptique.

⁵On les appelle le plus souvent des systèmes de *Motion Capture* ou *MoCap*.

5.3 Approche « interaction »

5.3.1 Définition

Dans cette approche, c'est le couplage homme-objet qui est l'objet de l'observation. À la différence des deux approches précédentes, le système couplé est considéré comme un système unique et indissociable (tableau 2.1 #3, page 33), et est observé du point de vue de l'observation physique ou phénoménologique et de sa modélisation. Cette approche fonctionnelle s'attache à l'étude des phénomènes propres au couplage homme-objet, qui ne peuvent apparaître ni dans l'étude de l'homme ni dans l'étude de l'objet considérés indépendamment. Il ne s'agit donc pas de l'observation du système couplé homme-objet dans son ensemble (que nous présentons plus loin dans l'approche « tâche »), mais plus particulièrement de l'observation des phénomènes spécifiques naissant dans le couplage mécanique entre les deux systèmes homme et objet.

Dans certaines situations, le couplage mécanique entre l'homme (ou par extension l'outil manipulé par l'homme) et l'objet en interaction donne lieu à des phénomènes physiques présentant un large spectre vibratoire [Bongers, 1994, Florens, 2003, Campion et Hayward, 2005] : doigt sur la vitre, craie contre le tableau... D'une manière générale, ces phénomènes apparaissent dans de nombreuses interactions mécaniques comme les collisions, les frottements, etc., et particulier dans les instruments de musique où le geste d'excitation est couplé à la structure vibrante, ce qui est le cas du frottement d'archet.

L'approche fonctionnelle « objet » considère l'objet comme un système ouvert, que l'on peut modéliser par un dipôle mécanique, et qui peut être dissocié de l'homme dans l'observation de l'interaction homme-objet. L'approche fonctionnelle « homme » est son pendant, s'attachant à l'observation de l'homme. Au contraire, l'approche fonctionnelle « interaction » considère le couplage mécanique homme-objet comme un nouveau système mécanique à part entière, que l'on ne peut rompre sous peine de modifier les phénomènes qui émergent de cette association particulière. Dans la situation d'interaction naturelle, le « doigt sur le verre de cristal » est un cas exemplaire de couplage mécanique relevant de l'observation fonctionnelle de type « interaction » : la manière si particulière qu'a le doigt de faire sonner la paroi du verre émerge du couplage entre le doigt et le bord de la paroi de verre et du frottement qui en résulte (figure 2.9). Dans ce cas, il n'est pas possible d'étudier indépendamment le doigt puis le verre de cristal pour étudier le phénomène vibratoire. Dans la situation médiatisée, le TGR se retrouve au centre du couplage homme-objet. L'introduction du TGR donne lieu à des modifications des propriétés physiques de l'objet médiatisé, mais aussi à des perturbations des lois physiques contenues dans le système de modélisation de l'objet virtuel simulé [Florens et Urma, 2006]. Cette approche concerne la simulation de situations artificielles où la qualité de l'interaction mécanique homme-objet joue un rôle fondamental : plusieurs simulations de la corde frottée dans le jeu de violon ont été implémentées à l'ACROE [Florens, 2003]. L'hypothèse générale qui a donné lieu à ces implémentations est que le geste instrumental, et en particulier un geste d'excitation comme le frottement d'archet, ne peut pas être réduit à un simple modèle de contrôle. Dans la simulation du frottement d'archet, un unique modèle adresse simultanément l'interaction haptique et les caractéristiques sonores de l'objet simulé. Au contraire d'autres simulations du frottement d'archet, cette implémentation présente la plupart des caractéristiques du jeu de violon propres à l'interaction colophanée entre l'archet et la corde (figure 2.9).

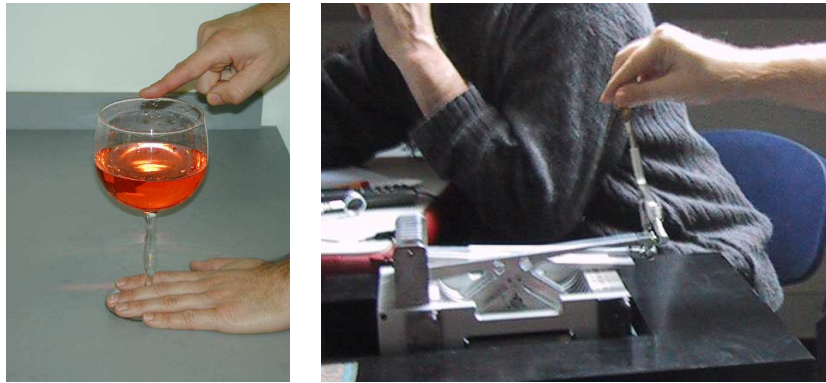


Figure 2.9. Cas exemplaires d'approche fonctionnelle de type « interaction » : le doigt sur le verre (à gauche) et la simulation de la corde frottée du violon (à droite)

5.3.2 Enaction, « interface enactive »

Le concept d'enaction fut introduit en 1960 par Jérôme Bruner, chercheur en psychologie cognitive spécialisé dans les théories de l'apprentissage. Ce concept fut repris et développé par Varela [Varela *et al.*, 1993], qui apparaît aujourd'hui comme le père de la théorie de l'enaction. À l'origine, la théorie de Bruner organise la connaissance en trois catégories :

- les représentations *iconiques*, basées sur une représentation graphique
- les représentations *symboliques*, associées aux représentations
- les représentations *enactives*, associées à l'action et au mouvement

L'hypothèse principale de Bruner est que toutes les représentations humaines du monde ne peuvent être décrites seulement à l'aide des représentations iconiques ou symboliques. Selon cette théorie, il serait en effet difficile d'exprimer des connaissances telles que faire du ski, jouer du violon par des représentations iconiques ou symboliques.

Le concept d'enaction propose une voie moyenne entre les théories cognitives objectivistes, selon lesquelles le monde réel et ses propriétés seraient projetées dans le système cognitif humain, et les théories subjectivistes, selon lesquelles les représentations du monde externe construites par la cognition humaine ne sont que le reflet de cette dernière. La théorie enactive de Varela propose une approche intermédiaire en mettant en avant le concept de couplage structurel entre le monde et son environnement externe. Ainsi, la théorie de l'enaction suppose que [ENACTIVE, 2004] :

- la perception est guidée par l'action,
- la capacité de l'être humain à « percevoir » est déterminée par sa capacité sensorimotrice.

Nous avons introduit précédemment la situation médiatisée et nous avons montré qu'elle supposait l'introduction d'un artefact technologique entre l'humain et le monde virtuel représenté par l'ordinateur : le TGR. Utiliser l'ordinateur comme un outil pour la représentation enactive suppose alors que l'ordinateur puisse supporter cette représentation enactive, et que le medium homme-ordinateur puisse également transmettre « l'enactivité » de cette représentation.

La théorie de l'enaction, en mettant l'accent sur le couplage structurel entre l'homme et son environnement, est particulièrement représentative de l'approche « interaction ».

5.4 Approche « tâche »

5.4.1 Introduction

Cette approche fonctionnelle considère le système homme-objet dans son intégralité (tableau 2.2 #4, page 45). Il est alors possible d'envisager plusieurs niveaux d'observation différents : topologie des couplages mécaniques, observation des mouvements de l'homme, commentaires de l'homme soit pendant l'interaction avec l'objet, ou bien dans un entretien à la fin de la tâche, évolution des caractéristiques physiques de l'objet avant, pendant et après la tâche, etc.

Ce niveau d'interaction est le plus général ; il comprend toutes les approches fonctionnelles qui ne sont pas comprises par les trois approches précédentes. La méthodologie consiste toujours à mettre en relation la situation naturelle et la situation médiatisée, mais, étant donné le niveau de généralité de cette approche fonctionnelle, nous devons envisager un second niveau de classification fonctionnelle. En effet, puisque nous considérons la relation homme-objet dans son niveau le plus global, la topologie des interactions mécaniques peut varier suivant les situations considérées mais aussi au cours d'une même interaction homme-objet. Dans cette approche, la topologie des relations homme-objet sera donc un second axe de classification de notre classification fonctionnelle. Nous reviendrons plus en détail sur ce point, mais nous pouvons déjà remarquer que si les relations topologiques sont amenées à varier au cours du temps, l'interaction considérée pourra passer d'une catégorie fonctionnelle de cette approche à une autre.

5.4.2 Ligne de mobilité

Nous supposons qu'au cours de chaque tâche, il existe nécessairement une frontière le long de laquelle les contacts mécaniques ne sont pas permanents. C'est, entre autres, ce qui permet de faire la différenciation physique entre l'homme et l'objet de l'interaction. Nous appellerons cette frontière la *ligne de mobilité*. D'un point de vue général, nous pouvons dire en effet que le corps humain présente de fait des liens mécaniques avec son environnement, mais que ces liens mécaniques ne sont pas permanents. Ainsi, des lignes de mobilité peuvent être définies relativement à la position de ces liens mécaniques sur le corps de l'homme et sur l'objet en interaction. Suivant la tâche considérée, il se peut que cette ligne de mobilité dépasse le corps de l'homme pour englober des objets en contact permanent, comme les outils. Dans un tel cas, la ligne de mobilité est déplacée entre l'objet-outil, et un autre objet ou l'environnement.

Afin d'illustrer cette notion de ligne de mobilité, prenons un exemple simple : la manipulation du tournevis. Dans un magasin au rayon bricolage, le tournevis a, d'un point de vue fonctionnel, le statut d'*objet* : je peux le saisir, le manipuler, le retourner pour l'ausculter sous toutes ses formes afin d'évaluer son poids, l'ergonomie de sa poignée, etc. Dans ce cas, les contacts mécaniques entre ma main et l'objet ne sont pas permanents et la ligne de mobilité se situe entre le tournevis et ma main. Dans l'atelier, je peux ensuite utiliser ce tournevis comme un *outil*, par exemple afin de visser des vis dans un morceau de bois. Dans ce cas, il y a une liaison mécanique permanente entre ma main et le tournevis devenu outil, et il y a une liaison mécanique intermittente entre le tournevis et les vis. Dans le cas du tournevis devenu outil, la ligne de mobilité est déplacée depuis la main jusqu'à la pointe du tournevis. D'une manière générale, c'est la ligne de mobilité qui nous permettra de créer une dissociation nette entre les différentes catégories fonctionnelles de tâches.

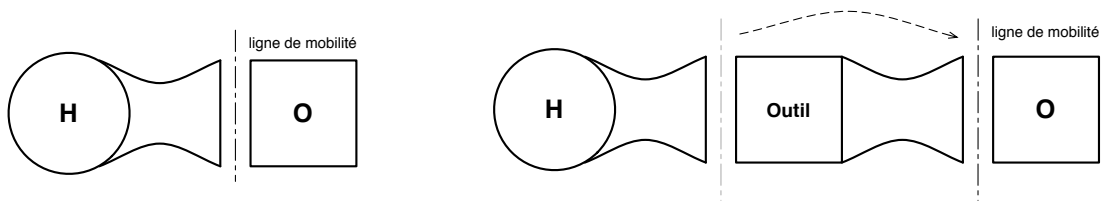


Figure 2.10. Position de la ligne de mobilité (en pointillés) dans le cas de l'objet manipulé (à gauche) et dans le cas de l'outil (à droite)

Nous introduisons la ligne de mobilité en prenant en compte deux aspects :

- La nécessité d'une ligne de mobilité autour du corps humain
- Le fait que la position de cette ligne de mobilité est déterminée par le type de tâche considérée.

5.4.3 Limite de l'environnement virtuel

Dans le cas de l'interaction homme-objet médiatisée, nous devons considérer une autre frontière : la limite de l'environnement virtuel. En effet, le statut de l'objet médiatisé navigue entre deux positions extrêmes [Cadoz, 1994] :

- Il peut être intégré partiellement ou complètement au corps humain (cas de l'embodiment)
- Il peut faire partie de l'environnement

Nous considérons alors les deux cas extrêmes, quant à la position de la limite de l'environnement virtuel (VE).

L'environnement virtuel « restreint » L'environnement virtuel est complètement défini par le calcul effectué par le simulateur (figure 2.11). Dans ce cas, l'interface haptique a le statut d'interface mécanique dans son sens habituel : elle rend accessible à l'utilisateur humain l'objet calculé dans le simulateur. Cette définition de l'environnement virtuel restreint convient à la plupart des situations non immersives.

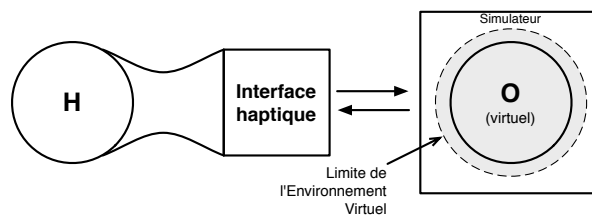


Figure 2.11. Limite de l'Environnement Virtuel « restreint » (en gris)

L'environnement virtuel « élargi » L'environnement virtuel englobe la frontière entre l'opérateur humain et l'objet en interaction (figure 2.12). Cette approche est particulièrement utilisée dans les applications immersives ; l'environnement virtuel contient alors la partie du corps humain comprise dans la tâche, voire tout le corps humain (immersion totale).

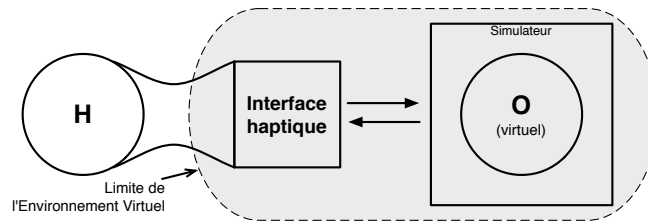


Figure 2.12. Limite de l'Environnement Virtuel « élargi » (en gris)

5.4.4 Situation #a. Interaction générale avec un environnement

Définition Cette situation concerne toutes les tâches où le lien entre l'homme et l'environnement n'est pas spécifié dans une relation avec un type particulier d'objet. L'environnement (E) est considéré dans son sens le plus général et peut contenir des liens avec le sol. Dans ce cas, la ligne de mobilité est nécessairement située à la limite du corps humain (H) (Tableau 1.2 #a—gauche 2.2, page 45).

Toutes les tâches de navigation sont comprises dans cette catégorie : la locomotion, mais aussi des tâches de navigation plus abstraites, par exemple trier ou ranger des livres dans une étagère.

La situation artificielle est médiatisée par l'intermédiaire d'un exosquelette, consistant en un lien mécanique permanent avec l'opérateur *H* (Tableau 1.2 #a — droite). L'exosquelette permet de placer la partie du corps humain qui lui est attachée dans une situation immersive. L'intérêt des systèmes exosquelettes est leur généralité théorique : idéalement l'exosquelette doit pouvoir tenir compte de tous les mouvements effectués par la partie du corps de l'utilisateur incluse dans la structure de l'exosquelette ; aussi, l'environnement de l'utilisateur est complètement déterminé pendant toute la durée de la tâche.

La ligne de mobilité de la tâche est située à la limite du corps humain. Étant donné que la partie du corps humain concernée par la tâche se trouve immergée par l'exosquelette dans l'environnement virtuel, la ligne de mobilité est située à l'intérieur de l'environnement virtuel.

La ligne de mobilité de la tâche est située à la limite immédiate du corps, et se trouve à l'intérieur de l'environnement virtuel. Cela signifie que l'interface haptique doit permettre en particulier des mouvements libres sur chacun de ses degrés de liberté. Cette situation immersive ayant la vocation d'être la plus générale possible et de placer l'utilisateur humain dans une situation de mouvement qui soit naturelle, l'exosquelette se doit d'intégrer le plus grand nombre possible de degrés de liberté des mouvements de l'utilisateur, et une grande palettes de mouvement différents (saisie, gestes sémiotiques — pointer du doigt —, déplacements, etc.). Cela signifie que les exosquelettes doivent répondre à des critères de transparence importants (en particulier pour compenser les effets mécaniques passifs de la structure de l'exosquelette, comme l'inertie ou la friction), et qu'ils sont confrontés à de sérieuses difficultés techniques de réalisation, parce que le critère de transparence est couplé avec la réalisation de cinématiques complexes afin d'imiter la cinématique des mouvements humains.

Locomotion Outre les exosquelettes, nous pouvons placer dans cette catégorie fonctionnelle les plateformes de locomotion. Ces systèmes sont des interfaces permettant à l'utilisateur de se



Figure 2.13. La plateforme de locomotion *Torus Threadmill* — Iwata et al.

déplacer dans un environnement virtuel par le biais de la locomotion, comme il le ferait dans la situation naturelle de référence.

La plateforme de locomotion *Torus Threadmill* a été développée à l'université de Tsukuba par Iwata et al. (figure 2.13).

5.4.5 Situation #b. Outil

Définition Dans cette situation, la ligne de mobilité principale se situe entre l'outil (T) et l'objet ou la partie de l'environnement sur lequel porte la tâche (E) (Tableau 2.2 #b, page 45). Le lien mécanique entre la main de l'opérateur (H) et l'outil (T) est permanent tout au long de la tâche outillée. Dans l'interaction outil-objet, il est nécessaire de considérer deux types de mouvements le long de la ligne de mobilité :

- Le geste de sélection. Ce mouvement consiste dans l'évolution de l'outil (T) d'une absence de contact à une situation de contact avec (E).
- Le geste ergotique, pendant lequel l'outil (T) agit sur l'objet de la tâche (E).

Dans les cas les plus courants de la situation d'interaction naturelle, les outils sont des systèmes artificiels constitués de un ou plusieurs éléments rigides. Ainsi, un outil possède une cinématique propre relativement simple, et n'est capable de fournir que quelques degrés de liberté. Par exemple, la pince commune possède 7 degrés de liberté (les 6 degrés de liberté d'un objet rigide dans l'espace, plus l'axe de rotation des deux pinces).

Dans la situation médiatisée, l'interface haptique se situe à cheval sur la limite de l'environnement virtuel. La partie tenue en main par l'utilisateur se trouve dans le monde réel, tandis que la partie en contact avec l'objet de la tâche se trouve dans l'environnement virtuel. Dans ce cas, l'environnement virtuel est de type *restreint*.

Alors que les interfaces haptiques immersives doivent présenter un degré de transparence suffisant pour permettre le mouvement libre, la métaphore de l'outil ne demande pas d'obtenir une transparence importante. En effet, l'outil en situation naturelle présente des caractéristiques mécaniques d'inertie, éventuellement de friction, qui lui sont propres. Dans la situation médiatisée, les

propriétés mécaniques de l'interface elle-même comme l'inertie ou la friction, perçues comme des inconvénients majeurs dans la situation immersive, seront ici attribuées aux propriétés mécaniques de l'outil. Cette caractéristique de l'interaction outillée, combinée avec le nombre limité de degrés de liberté exigés par ce type d'interaction, rendent la réalisation d'interfaces haptiques pour l'interaction outillée plus simples.

En contrepartie, dans la situation médiatisée, le système dynamique créé par le couplage mécanique outil-objet est complètement supporté par le système de simulation. Le simulateur doit donc pouvoir correctement représenter des comportements d'interactions complexes, notamment les phases de mise en contact.

Interfaces haptiques de table Les bras haptiques de table, encore qualifiés de « desktop », constituent une famille de systèmes haptiques fortement représentée actuellement. Le plus souvent ces interfaces sont conçues sur le principe de l'interaction outillée ; elles présentent une poignée similaire à un outil, permettant la préhension, et offrent en général entre 3 et 6 degrés de liberté.

La société Sensable commercialise les interfaces haptiques PHANTOM. Le premier système PHANTOM fut commercialisé dès 1994 [Massie et Salisbury, 1994]. Ce fut également le premier système commercialisé à grande échelle : on trouve aujourd'hui au moins un système PHANTOM dans pratiquement tous les laboratoires dont les recherches sont liées au domaine de l'haptique. Ces systèmes possèdent de simples moteurs DC alimentés par un signal PWM, situés dans la base du système. La transmission de mouvement est effectuée par la combinaison de systèmes pantographes et de transmission par câbles. Le PHANTOM « desktop » est le plus petit de la gamme. Il possède 3 DDL pour un espace de travail de $16 \times 13 \times 13$ cm, et la raideur maximale atteignable est de 3 100 N/m. Le système « Premium 3.0 » est le haut de la gamme. Il possède 6 DDL pour un espace de travail de $41 \times 59 \times 84$ cm, et la raideur maximale atteignable est de 1 000 N/m.

Le Haptic Master est commercialisé par Moog FCS Control Systems (figure 2.14). C'est un système à 3 degrés de liberté dont les dimensions sont légèrement supérieures aux autres interfaces haptiques de type « outil » (env. $1 \times 1 \times 1$ m). Le système fonctionne en admittance, ce qui permet de présenter de faibles forces de friction en mouvement libre tout en permettant la simulation de raideurs relativement importantes (5. 104 N/m). En raison de sa taille, le Haptic Master système procure un grand espace de travail, mais possède aussi une inertie propre non négligeable (env. 2 kgs).

Parfois les propriétés spatiales et dynamiques de l'interface haptique sont caractérisées précisément puis corrigées pour satisfaire le modèle d'interaction outillée utilisé comme référence. Par exemple, la correction d'inertie est appliquée sur le système Lhifam conçu à CEIT [Savall *et al.*, 2002].

5.4.6 Situation #c. Commande

Définition Cette catégorie correspond à la plupart des systèmes artificiels conçus comme des interfaces (au sens général) à la main de l'homme : leviers, boutons, manettes, etc. Dans cette situation, la ligne de mobilité est située entre l'utilisateur (H) et la commande (C) (tableau 2.2 #c, page 45). La différence avec la situation précédente, l'interaction outillée, réside dans le fait

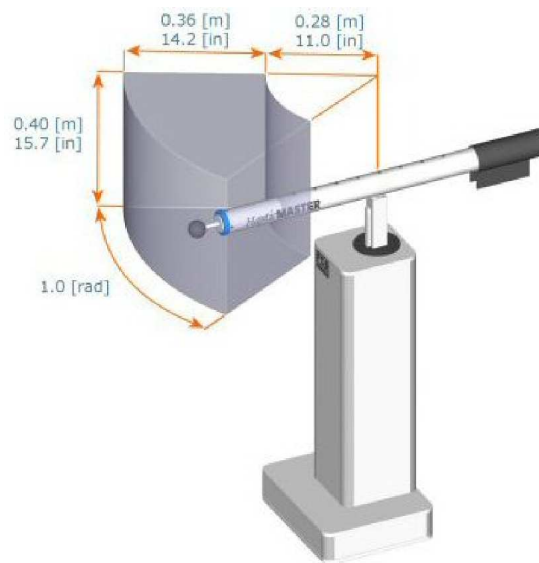


Figure 2.14. Le Haptic Master — FCS Control Systems

que la ligne de mobilité est déplacée entre la main (H) et la commande (C). D'un point de vue fonctionnel, seul le geste de sélection est pertinent pour la fonction de commande.

La manipulation du levier de vitesse illustre parfaitement cette situation : la ligne de mobilité est située entre la main et le levier, et il y a des contacts mécaniques permanents entre tous les composants de la chaîne de commande, depuis le levier de vitesse jusqu'au sol. Le clavier de piano est un autre exemple de commande : l'interaction entre la main et le clavier est faite de phases de contact et de phases de non-contact.

Dans la situation médiatisée, la ligne de mobilité n'est pas nécessairement comprise dans l'environnement virtuel. Cette situation est proche de l'interaction outillée du point de vue des contraintes de réalisation des systèmes haptiques : le nombre de DDL est généralement inférieur ou égal à 6 et une transparence importante n'est pas nécessaire.

Interfaces haptique pour la commande S'il existe une différence fonctionnelle entre les interfaces de type « outil » et les interfaces de type « commande », toutes les interfaces de type « desktop » peuvent être utilisées dans des tâches de commande. Outre les interfaces de type « desktop », nous pouvons citer quelques interfaces haptiques spécifiquement conçues pour des applications de commande.

Le Senso-Wheel SD-17C (figure 2.15) est un volant à retour d'effort commercialisé par SensoDrive, une société créée pour commercialiser les produits développés au DLR, l'institut aérospatial allemand. Ce système fut conçu pour être utilisé dans les simulateurs à retour d'effort. Il s'agit d'un volant à retour d'effort possédant un DDL en rotation, actionné par un moteur brushless DC. Le couple maximum est de 26 Nm. La société SensoDrive travaille sur un projet d'intégration de ce volant dans des véhicules automobiles de marque allemande.

Immersion Corporation développe des boutons rotatifs programmables [imm, 2008]. Il s'agit de composants de petites tailles destinés à remplacer les boutons rotatifs de sélection que l'on



Figure 2.15. Le volant à retour d'effort Senso-Wheel commercialisé par SensoDrive



Figure 2.16. Le bouton rotatif programmable PR-1000 de Immersion Corporation

trouve par exemple habituellement dans les voitures (figure 2.16). Ces systèmes sont purement passifs, intégrant un frein électromagnétique permettant de programmer une force de résistance au mouvement en fonction de l'angle de rotation du bouton.

Le « Pantograph » fut développé à l'université de McGill par V. Hayward et al. [Hayward *et al.*, 1994]. Ce système a été conçu en particulier pour la manipulation digitale : il s'agit d'un système à 2 degrés de liberté, possédant un espace de travail de 10×16 cm. Les mouvements de deux moteurs DC sont combinés par un système de pantographes mécaniques en aluminium. Ce système est surtout utilisé pour des applications de type HCI [Ramstein et Hayward, 1994], dont la manipulation s'apparente à la manipulation d'une souris d'ordinateur (figure 2.17).

5.4.7 Situation #d. Objet

Définition Cette catégorie de tâches fonctionnelles concerne en particulier l'objet en situation de manipulation. Dans cette situation l'objet est en interaction mécanique directe avec la main de l'opérateur (H) ; la ligne de mobilité se situe donc entre l'opérateur et l'objet manipulé (O) (tableau 2.2#d, page 45).

Alors que la situation #a (interaction directe avec un environnement) est basée essentiellement sur des tâches centrées sur la vision ou l'audition telles que la navigation, cette situation, proche de la situation #a du point de vue de la topologie des interactions, est centrée sur la manipulation, et la

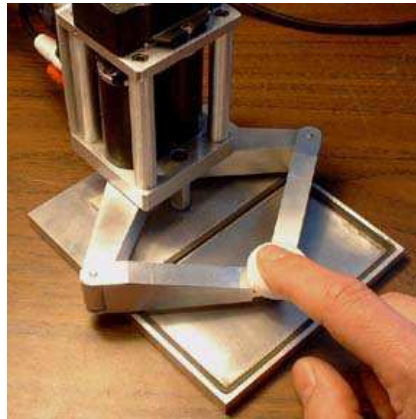


Figure 2.17. Le Pantograph — Université de McGill, Montreal

modalité visuelle est reléguée à un rôle de contrôle de l'action : A. Luciani parle de situation de vis-à-vis [Luciani, 2004]⁶.

De la même manière que dans l'interaction outillée (situation #b), nous pouvons différencier deux types de gestes :

- le geste de sélection, comprenant les gestes d'approche main-objet (mouvement libre sans interaction),
- le geste ergotique, associé à la mise en contact, où a lieu un échange d'énergie dans l'interaction mécanique.

Cette différence entre les situations naturelles d'interaction avec un environnement (de navigation) et de vis-à-vis avec un objet donne lieu à une différence fondamentale dans le cas de la situation médiatisée : la situation #a, orientée sur les tâches de navigation, devient immersive, alors que la situation #d, orientée sur les tâches de manipulation, peut conserver le schéma de vis-à-vis dans la relation homme-objet. Cependant, l'interface haptique doit présenter des propriétés particulières parce que la ligne de mobilité se trouve *entre* la main et l'objet. L'interface fait donc partie de l'objet médiatisé, et la frontière de l'environnement virtuel coïncide avec la limite physique de l'objet (tableau 2.2#d, page 45).

Systèmes tactiles La plupart des interfaces tactiles sont des systèmes vibrotactiles. Le principe est le suivant : l'utilisateur touche une surface déformable constituée d'une matrice de pointes mobiles. Chaque pointe peut se déplacer individuellement à la verticale sur une course de plusieurs millimètres, actionnée grâce à des piezzos.

Summers et al. ont développé à l'université d'Exeter une interface tactile par stimulation vibrotactile [Summers *et al.*, 2005]. Le système est une matrice de 11 cm, composé de 100 pointes

⁶Le vis-à-vis tel qu'il est présenté par A. Luciani [Luciani, 2004] correspond à nos trois catégories fonctionnelles des tâches outillées, de commande, et de manipulation d'un objet. La situation de vis-à-vis s'oppose à la situation immersive, et réfère aux objets se trouvant dans l'espace proche de l'homme, i.e. à portée de main. Les concepts de vis-à-vis et d'immersion sont considérés comme des points extrêmes sur l'axe mesurant la proximité homme-objet (1– au loin, 2– à proximité du corps, 3– en contact). Cette notion est importante dans la définition des environnements virtuels puisqu'elle définit la nécessité ou non d'implémenter des métaphores de navigation. En outre, l'objet se trouvant dans la situation de vis-à-vis peut devenir prothèse, c'est-à-dire devenir partie du corps humain, par le processus d'*embodiment*.



Figure 2.18. Deux réalisations du FEELEX : le FEELEX 1 (1997) avec une surface d'interaction de 24×24 cm (à gauche) ; le FEELEX 2 (2001), avec une surface de 5×5 cm (à droite)

mobiles disposées en grille 10×10 , entourées par une rangée de 44 pointes fixes, permettant ainsi que chaque pointe mobile soit complètement entourée par d'autres pointes. Les pointes sont actionnées individuellement par des piezzoélectriques pouvant les mettre en vibration dans la plage de fréquences [20 ; 400] Hz.

FEELEX Le FEELEX est un système particulier dans le paysage de la recherche haptique. Ce système fut initialement conçu par Iwata et al. à l'université de Tsukuba pour combler le manque d'interfaces haptiques permettant la manipulation directe d'un objet virtuel [Iwata *et al.*, 2001]. Ce système fut conçu avec l'idée de procurer à ses utilisateurs une interaction avec une forme spatialement continue (i.e. une surface), plutôt qu'une interaction ponctuelle, comme c'est le cas pour toutes les interfaces haptiques de type « desktop » (cf. interfaces de table, paragraphe 5.4.5 dans ce chapitre).

Le FEELEX 1 (figure 2.18, à gauche), développé en 1997, est composé d'une matrice de 36 tiges métalliques mobiles verticalement sur plusieurs centimètres, sur laquelle repose une surface déformable en contact avec l'utilisateur, et d'un vidéoprojecteur projetant des images directement sur cette surface. L'objet simulé est représenté mécaniquement par l'intermédiaire de la surface déformable, et représenté visuellement par la projection d'images directement sur la surface déformable.

Les applications ciblées par ce système sont la simulation de gestes de palpation pour la médecine, la modélisation de formes tridimensionnelles, les écrans tactiles, ou encore les sculptures interactives (« toucher l'image » de la sculpture). Hormis les applications basées sur le concept d'écran tactile, qui relèvent de la catégorie « commande » dans l'approche fonctionnelle de type « tâche », toutes les autres applications relèvent de la catégorie « objet » dans l'approche fonctionnelle « tâche ».

Le FEELEX 2 fut ensuite développé en 2001 (figure 2.18, à droite) dans le dessein d'améliorer la résolution spatiale de la surface déformable. La nouvelle résolution spatiale (8 mm) fut déterminée en prenant pour situation de référence la palpation de tumeurs, dont le diagnostic s'effectue en comparaison avec la distance entre deux doigts. Le FEELEX 2 est constitué de 23 pointes, permettant une surface de palpation de 50×50 mm.

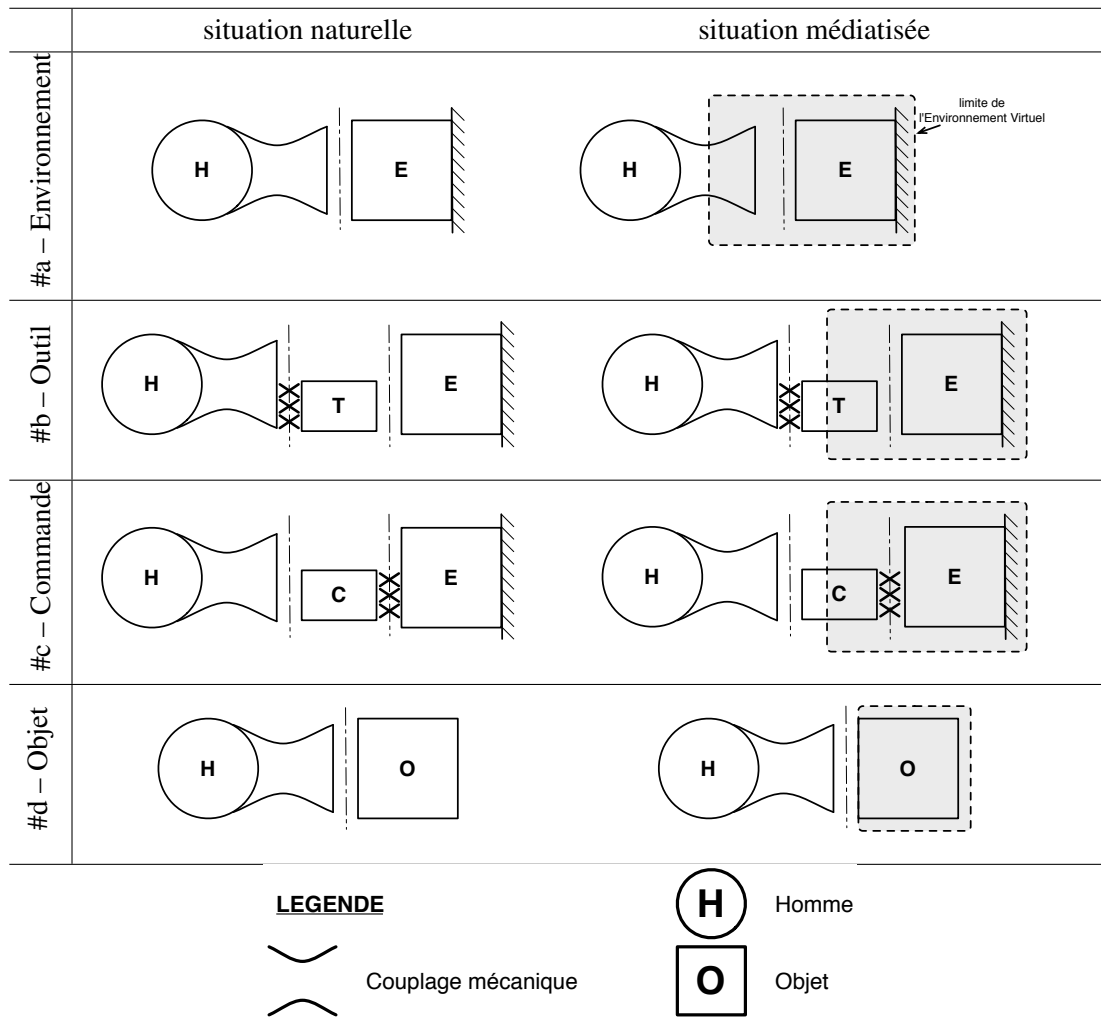


Tableau 2.2. Les quatre catégories de l'approche fonctionnelle « tâche »

Interfaces rhéologiques Les fluides rhéologiques ont la particularité de présenter une viscosité variable en fonction du champ électrique auquel ils sont soumis. Les fluides électro-rhéologiques sont sensibles aux champs électriques, alors que les fluides magneto-rhéologiques et les ferro-fluides sont sensibles aux champs magnétiques.

Les fluides rhéologiques constituent une nouvelle technologie pour l'actuation : on trouvera un exemple d'application dans [Mavroidis *et al.*, 2000a, Mavroidis *et al.*, 2000b] ; [[Fisch *et al.*, 2003] dans des dissipateurs actifs pour la simulation de la gravité ; [THSV96] dans des interfaces tactiles.

Scilingo et al. utilisent les propriétés de viscosité variable des fluides rhéologiques pour représenter différents états de la matière en contact direct avec la main. La ligne directrice de ces travaux est la réalisation d'interfaces haptiques permettant aux utilisateurs de manipuler directement l'objet virtuel. Le fluide rhéologique est excité à l'aide d'un champ électrique, et l'utilisateur peut manipuler le fluide au travers d'une enveloppe protectrice. Cette technique a été utilisée pour simuler la raideur des tissus biologiques dans des applications de chirurgie peu invasive, d'abord pour la manipulation pouce-index [Scilingo *et al.*, 2000], puis pour la manipulation à l'aide de

toute la main [Scilingo *et al.*, 2003b, Scilingo *et al.*, 2003a].

À l'heure actuelle, cette technique est immersive, étant donné que l'utilisateur doit plonger la main dans le liquide rhéologique, mais cette immersion n'est pas fonctionnelle, à l'inverse des exosquelettes ; d'un point de vue fonctionnel, la situation d'interaction obtenue est bien celle du schéma homme-objet.

6 Conclusion

Ce chapitre présente un parcours synthétique de l'état de l'art sur la technologie du retour d'effort. En orientant notre observation uniquement sur les propriétés fonctionnelles des systèmes étudiés, nous proposons une approche complémentaire des classifications basées sur les principes technologiques.

Nous avons effectué ce parcours du point de vue fonctionnel en considérant à la fois les théories, la conception d'interfaces haptiques et leurs applications. Quatre catégories fonctionnelles apparaissent :

- L'approche « objet » concerne exclusivement l'observation de l'objet en situation d'interaction. Elle comporte les méthodes de représentation d'objets virtuels, et les travaux s'intéressant au comportement de l'interface haptique observée depuis le côté « homme ».
- L'approche « homme » est le symétrique de l'approche objet : elle concerne l'observation de l'homme en situation d'interaction. Cette approche est pratiquée essentiellement par les disciplines ayant trait à l'étude de l'homme : psychophysique, neurosciences, etc.
- L'approche « interaction » considère le système constitué par le couplage de l'homme et de l'objet dans la situation d'interaction. Elle considère les phénomènes spécifiques émergeant du couplage mécanique entre les deux entités homme et objet.
- L'approche « tâche » considère le système homme-objet dans sa globalité, à différents niveaux d'observation. En observant les relations topologiques entre l'homme et l'objet, et les limites de l'environnement virtuel dans la situation médiatisée, quatre catégories de tâches émergent : l'interaction de type « environnement », outillée, de type commande, et objet.

Nous avons vu que c'est l'approche « interaction » qui supporte la théorie de l'enaction. En particulier, cette approche supporte le paradigme du geste instrumental, qui, tel que conceptualisé dans les travaux de Claude Cadoz et de l'ACROE, ne peut se passer du couplage fort entre l'homme et l'objet. Seule une approche théorique fondée sur ce paradigme et portant une attention particulière à la ré-introduction d'un couplage fort entre l'homme et l'objet-instrument virtuel sera à même de restituer les qualités intrinsèques au geste instrumental dans la situation naturelle, et que nous souhaitons réintroduire dans la situation médiatisée.

Chapitre 3

Architectures pour la Simulation Haptique

Sommaire du Chapitre

1	Introduction	48
2	Les composants	49
2.1	Schéma général	49
2.2	Processus simulation, processus superviseur	50
3	Approches asynchrones	51
3.1	Le compromis entre puissance de calcul et réactivité	51
3.2	Haptic rendering	52
3.2.1	Boucle de commande	53
3.2.2	Boucle lente	54
3.2.3	Communication entre la boucle de commande et la boucle lente	55
3.2.4	Approches multi-fréquences et stabilité	55
3.3	Réalité Virtuelle	55
3.4	Performances des architectures de simulation actuelles	57
4	Le contrôleur haptique	58
4.1	Introduction	58
4.2	Exemples d'implantations du contrôleur haptique	61
4.2.1	Rutgers Master II-ND – Rutgers University of New Jersey	61
4.2.2	MagLev – Robotic Institute – Carnegie Mellon University	62
4.2.3	Delta Haptic Device – ForceDimension	63
4.2.4	High Performance Haptic Interface – Queen's University, Kingston	63
4.2.5	L-Exos – PERCRO	64
4.2.6	Haptic Master – Moog FCS	64
4.2.7	SPIDAR-G – Tokyo Institute of Technology	65
4.3	LHifAM – CEIT	65
5	Systèmes commerciaux	71
5.1	Sensible et les bras haptiques <i>Phantom</i>	71
5.2	Force Dimension – systèmes <i>Delta</i> et <i>Omega</i>	72

6	Approches synchrones	73
6.1	Introduction	73
6.2	Réalisations de simulateurs synchrones	74
6.3	Les simulateurs de l'ACROE	75
6.4	TELLURIS	76
6.4.1	Architecture de la plateforme	76
6.4.2	Le compromis entre généricité et efficacité	78
7	Conclusion	79

1 Introduction

Le chapitre 2 mettait l'accent sur la relation homme-objet d'un point de vue fonctionnel. Nous avons vu que la théorie de l'enaction, supportée par l'approche « interaction », est encore peu représentée dans les travaux pour la simulation gestuelle. Ce chapitre propose une analyse complémentaire du chapitre précédent, en proposant l'étude des caractéristiques techniques et des choix conceptuels utilisés dans les architectures matérielles et logicielles pour la simulation multisensorielle. En particulier, si nous faisons l'hypothèse que la restitution d'un couplage fort entre l'homme et l'objet est nécessaire dans la situation instrumentale médiatisée, et que nous souhaitons prendre les dispositions techniques nécessaires pour garantir ce couplage, nous devons alors porter une attention particulière aux conditions techniques qui permettent sa réalisation.

Ce chapitre s'attache donc à la description des composants matériels et logiciels que comprend la chaîne de simulation haptique, en mettant un accent particulier sur les conditions de réalisation du couplage entre l'homme et l'objet simulé. En effet, c'est de la réalisation du couplage instrumental dans la relation de l'homme avec un objet médiatisé que dépend la réalisation de ce que nous pourrions appeler une « situation enactive ».

Nous porterons également dans ce chapitre une attention particulière aux conditions de synchronisation entre les éléments qui interviennent dans la chaîne de simulation. Les conditions de synchronisation déterminent les conditions de réalisation satisfaisantes pour l'implantation du couplage mécanique à même de supporter le geste instrumental. La première approche, asynchrone, consiste à supposer que l'exécution de manière pseudo-parallèle¹ de tous les composants logiciels de la chaîne de simulation haptique permet de s'affranchir des contraintes de synchronisation. La seconde approche assume les contraintes temporelles liées au synchronisme. Nous montrons que l'approche synchrone est nécessaire, par exemple dans les situations où la dynamique des modèles à simuler est une caractéristique importante.

¹Nous utilisons le terme « pseudo-parallèle » car l'exécution de ces différents processus n'apparaît parallèle qu'à un niveau d'observation relativement macroscopique, ou peu détaillé. La plupart du temps en effet, tous les composants logiciels sont exécutés sur une architecture mono-processeur, ce qui suppose, si nous plaçons notre observation au niveau de l'ordonnanceur du système d'exploitation, une exécution nécessairement séquentielle.

2 Les composants

2.1 Schéma général

Décrivons l'architecture générale d'un simulateur multisensoriel interactif.

Le cœur du simulateur est le ordinateur, qui traite toutes les opérations nécessaires au bon déroulement du processus en charge de la simulation. Ce processus de simulation ne peut être rendu sensible à l'utilisateur que par le biais de transducteurs. Le simulateur comporte autant de types de transducteurs que le nombre de sensorialités que l'on souhaite adresser. Typiquement, un simulateur multisensoriel comprendra :

- Une transducteur gestuel rétro-actif (le TGR)
- Un écran
- Des haut-parleurs

Sur les architectures modernes, les transducteurs sont connectés au ordinateur par l'intermédiaire de périphériques de contrôle spécifiques, dont la principale fonction est d'assurer la traduction des signaux échangés dans un format qui puisse être traité d'un côté par le ordinateur, de l'autre par le transducteur.

La figure 3.1 présente de manière synthétique les différents composants d'un simulateur multisensoriel. Nous allons par la suite détailler ces composants et leurs relations.

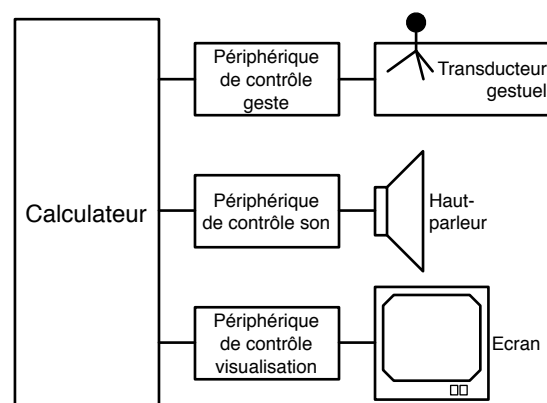


Figure 3.1. Schéma synoptique du simulateur multisensoriel

Ce chapitre s'attache à l'étude des différentes architectures pour la simulation haptique que l'on peut trouver dans la littérature. L'objet de cette partie comprend à la fois l'architecture pour le calcul du processus de simulation proprement dit, mais aussi l'architecture qui permet d'interfacer ce processus de simulation avec un TGR. Les limites de cette étude sont :

- D'une part, les entrées/sorties du TGR. Nous considérerons le TGR comme une boîte noire, dont seules sont connues ses caractéristiques pertinentes pour l'utilisateur (espace de travail, nombre de degrés de liberté) et le format de ses entrées/sorties (signaux analogiques ou numériques, codage des signaux numériques, etc.).

- De l'autre, les entrées/sorties du processus de simulation, qui sont toujours dans un format numérique².

Il s'agit donc en quelque sorte de mettre en relation les interfaces de communication (1) du TGR et (2) du processus de simulation, et ce à plusieurs niveaux :

- Physique : par la mise en relation de flux d'information qui ne sont pas nécessairement disponibles sur un même support physique ou sur des architectures de calcul identiques ou seulement compatibles
- Temporel : le cadencement des flux de communication de ces deux interfaces, comme nous le verrons par la suite, est le plus souvent diamétralement différent
- Informationnel : il s'agit parfois de mettre en relation des flux d'information qui ne peuvent pas être immédiatement associés, soit pour des raisons de codage de l'information, soit par la nature même des informations à communiquer³.

2.2 Processus simulation, processus superviseur

Dressons un schéma général et simplifié du processus de simulation : le processus de simulation comporte une boucle de calcul durant laquelle, à chaque pas de simulation, sont mises à jour les variables d'état du système simulé en fonction de ses points d'entrée. Le calcul d'un pas de simulation commence donc par prendre connaissance des nouvelles valeurs des variables d'entrées, met à jour l'état du modèle simulé, puis communique aux autres processus impliqués dans l'application globale de simulation le nouvel état de la simulation en cours. Chaque pas de simulation sera donc dissocié en trois phases consécutives [Uhl, 1996] :

- (a) Le début de la phase (a) marque le début d'un pas de simulation. Durant cette phase, le processus simulation se met en attente des données d'entrée du modèle simulé. Typiquement, ces données proviennent du module geste : les données en entrée du TGR sont échantillonnées et traitées (calibrage), puis envoyées au processus simulation. Simultanément, la conversion numérique analogique est effectuée : les données reçues du simulateur sont converties pour être fournies aux moteurs du TGR.
- (b) La phase (b) est la phase de calcul du modèle physique simulé : au cours de cette étape, on rappelle que le calculateur va exécuter tous les algorithmes d'interaction (calcul des forces) du modèle puis tous les algorithmes de masse (calcul des positions).
- (c) La phase (c) comporte le transfert des données de sortie du modèle simulé à destination des autres processus en attente. Typiquement, les données de sortie du modèle sont émises par le processus simulation à destination des processus geste, son et image. Les données transmises du processus simulation au processus geste sont prêtes à être transférées (conversion numérique analogique) au TGR durant la phase (a) du prochain cycle de simulation.

²Si le processus de simulation est aujourd'hui toujours exécuté sur un support informatique numérique, on pourrait envisager son calcul sur un support analogique. Ainsi, la première implantation du formalisme de simulation modulaire CORDIS fut implantée sur carte électronique analogique.

³Par exemple, il n'est pas possible d'interfacer directement un moteur de simulation dont les entrées/sorties sont de type événementiel (date, emplacement et nature d'une collision) avec un TGR dont les entrées/sorties sont des flux de données numériques isochrones correspondant, pour la sortie, à l'état du système à un instant t , et pour l'entrée, à une consigne à appliquer au système.

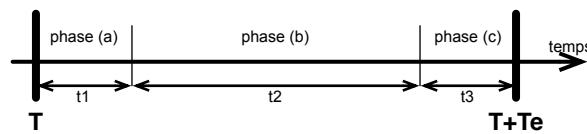


Figure 3.2. Découpage d'un cycle de simulation type

Nous pouvons constater qu'un système de simulation efficace visera à réduire le temps d'exécution des phases (a) et (c), afin de laisser le maximum de temps disponible à la phase (b), c'est-à-dire au calcul effectif du modèle simulé. Dans un simulateur idéal, t_2 tend vers T_e . Nous distinguerons donc trois types de processus qui s'exécuteront en parallèle sur simulateur-type :

- Le ou les *processus système* ; il s'agit des processus nécessaires au bon fonctionnement du système d'exploitation qui héberge l'application de simulation.
- Le *processus superviseur* : le processus père de tous les processus nécessaires à l'application de simulation. C'est ce processus qui contrôle l'application de simulation et en particulier la genèse du processus simulation et son arrêt, mais aussi l'interface graphique de contrôle de l'application, etc.
- Le ou les *processus simulation* : ces processus prennent en charge uniquement les procédures nécessaires à l'exécution de la boucle de simulation et leur synchronisation avec les périphériques de contrôle des transducteurs.

3 Approches asynchrones

3.1 Le compromis entre puissance de calcul et réactivité

Le développement d'une application de simulation haptique doit faire face à deux critères de performance contradictoires :

Réactivité Le processus de simulation doit communiquer en temps réel avec le TGR ; et ceci à des fréquences de simulation relativement importantes afin d'assurer un domaine de stabilité numérique suffisant. Le fait de travailler en boucle fermée impose des conditions de synchronisation relativement strictes, et demande que la fréquence des échanges de données soit suffisamment importante. Ainsi, il est généralement considéré que ce processus doit s'exécuter à une fréquence avoisinant 1 kHz [Salisbury *et al.*, 2004].

Puissance de calcul Les modèles simulés comportent de nombreux éléments et la tendance actuelle va vers la complexification des modèles simulés. Simuler rapidement un modèle complexe demande une puissance de calcul importante.

Afin de faire face à ces deux contraintes diamétralement opposées — d'une part pouvoir calculer *rapidement*, de l'autre pouvoir calculer une *grande quantité* d'éléments — la majorité des techniques pour la simulation haptique utilisent les techniques de calcul réparti. Le processus de simulation est séparé en plusieurs composants indépendants (simulation du modèle mécanique, synthèse des événements sonores, génération de la représentation graphique. . .) et chaque composant exécute sa boucle de calculs le plus souvent de manière asynchrone du reste des composants de l'architecture logicielle de simulation. Cette approche se base sur les caractéristiques de la

perception humaine et il est fait l'hypothèse que la fréquence de rafraîchissement de chaque composant doit être suffisamment rapide pour que le délai résultant ne soit pas perçu par l'utilisateur. Par exemple, soit la boucle de simulation du modèle mécanique est exécutée à 1 000 Hz et la boucle graphique à 25 Hz, alors le délai maximal entre deux mises à jour sera de 1 ms + 20 ms, soit 21 ms, et il est supposé que ce délai est inférieur au seuil de détection de la perception humaine pour la boucle geste-vision. Nous distinguons deux composants en particulier :

- Une boucle de commande prend en charge le minimum des calculs nécessaires à l'interaction avec le TGR, parfois aussi appelée dans la littérature *boucle de commande*, *boucle de contrôle* [Leuschke *et al.*, 2005] ou *boucle haptique* [MacLean *et al.*, 1999] ; dans la suite de ce mémoire, nous garderons l'appellation boucle de commande. Cette boucle contient a minima le processus de communication synchrone avec le TGR.
- Une boucle lente est chargée de la simulation du reste du modèle, et de la communication des données avec la boucle de commande. La fréquence d'exécution de cette boucle est le plus souvent calée sur les performances des processus graphiques [Ruspini *et al.*, 1997].

3.2 Haptic rendering

Une majeure partie de la communauté haptique, issue du champ de l'infographie, pratique une approche généralement dénommée haptic rendering [Salisbury *et al.*, 1995]. Cette approche considère la simulation haptique comme une représentation de solides (donc des formes géométriques), ou d'effets mécaniques le plus souvent associés aux textures des objets statiques représentés (granularité, friction, etc.).

Cette démarche relève du raisonnement suivant [Adachi *et al.*, 1995] :

- Le point manipulé, encore appelé avatar⁴ [Salisbury *et al.*, 2004], c'est-à-dire la métaphore du système gestuel dans le modèle, est représenté dans le modèle simulé par un élément ponctuel.
- On fait l'hypothèse que les objets simulés sont statiques : l'interaction avec le point manipulé est de type 'collision'.
- Les forces d'interaction entre le point manipulé et une surface peuvent être définies selon la normale à cette surface.
- Cette surface peut être approximée par un plan tangent à la surface au point de contact avec le point manipulé. Ceci permet de simplifier les calculs d'interaction entre le point manipulé et la surface de l'objet en interaction.

Cette méthode a le mérite de limiter la quantité de calculs nécessaires à la boucle de commande au strict minimum. Elle permet ainsi une meilleure répartition des ressources de calcul en permettant, par le choix d'une fréquence de simulation plus faible pour la simulation du modèle qui n'est pas en relation immédiate avec le TGR, d'augmenter la complexité du modèle et le nombre d'éléments calculables dans un pas de simulation. D'autre part, cette désynchronisation permet de maintenir une fréquence de simulation importante pour la boucle de commande, et donc de garantir une qualité de restitution de contacts relativement satisfaisante.

⁴« An avatar is the virtual representation of the haptic interface through which the user physically interacts with the virtual environment. » [Salisbury *et al.*, 2004]

En infographie, un avatar est un humanoïde animé par image de synthèse, et plus généralement, ce terme désigne l'apparence que prend un internaute dans un univers virtuel.

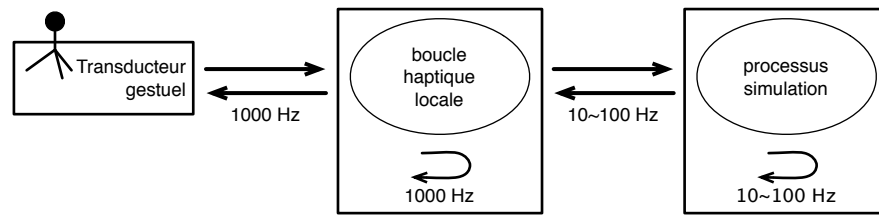


Figure 3.3. Approche haptic rendering : deux boucles asynchrones permettent une meilleure répartition des ressources de calcul du simulateur

Enfin, il est possible d’objecter que la fidélité de l’interaction avec le modèle peut se trouver appauvrie en fonction de la vitesse de calcul de la boucle lente. Cependant, les tenants de cette approche font le choix d’appauvrir la qualité de la simulation lorsque les ressources du système sont insuffisantes, plutôt que de risquer de créer des domaines d’instabilité :

By running the control loop at a high fixed clock rate, stability is easier to ensure and the fidelity of the haptic display degrades gracefully as the complexity of the environment is increased. If the proxy update procedure is unable to maintain the same rate as the controller, objects feel "sticky". While this effect may not be desirable, it is preferable to permitting unstable and dangerous behaviour of the haptic device.

D.C. Ruspini, K. Kolarov, O. Khatib “The Haptic Display of Complex Graphical Environments” – [Ruspini et al., 1997]

3.2.1 Boucle de commande

La littérature haptique a consacré la fréquence de simulation de cette boucle de calcul autour de 1 kHz [Zilles et Salisbury, 1995, Ruspini *et al.*, 1997, Salisbury *et al.*, 2004, Grange *et al.*, 2001a] : il est généralement considéré que cette fréquence de simulation constitue un bon compromis entre fidélité des sensations restituées à l’utilisateur, taille du domaine de stabilité, et coût en ressources de calcul⁵.

La figure 3.4 présente les différents composants mis en jeu dans la simulation haptique (et plus généralement multi-sensorielle) par les adeptes de l’approche dite *haptic rendering*. La boucle de commande comprend *a minima* le processus de contrôle du périphérique haptique (control algorithms – figure 3.4). Ce processus gère la communication des entrées / sorties du TGR et leur synchronisation. Suivant la morphologie du système haptique, il peut également être en charge

⁵Une fréquence de simulation de 1 kHz correspond à une période de une milliseconde. La résolution temporelle du système doit être au moins cent fois plus petite pour permettre un cadencement régulier par l’ordonnanceur du système. Ces ordres de grandeur correspondent aux résolutions temporelles des systèmes d’exploitation qui ne sont pas spécialisés dans la réalisation de tâches en temps-réel (Microsoft Windows, Linux non patché pour le temps réel, Apple OSX), ce qui peut expliquer pourquoi le kHz est encore aujourd’hui l’ordre de grandeur de référence pour la simulation haptique : si les architectures exigeantes pour la simulation haptique ne peuvent faire l’économie d’un système supportant le temps réel dur, les adaptations commerciales destinées à une utilisation plus large se doivent de proposer des solutions supportées par les systèmes d’exploitation les plus utilisés.

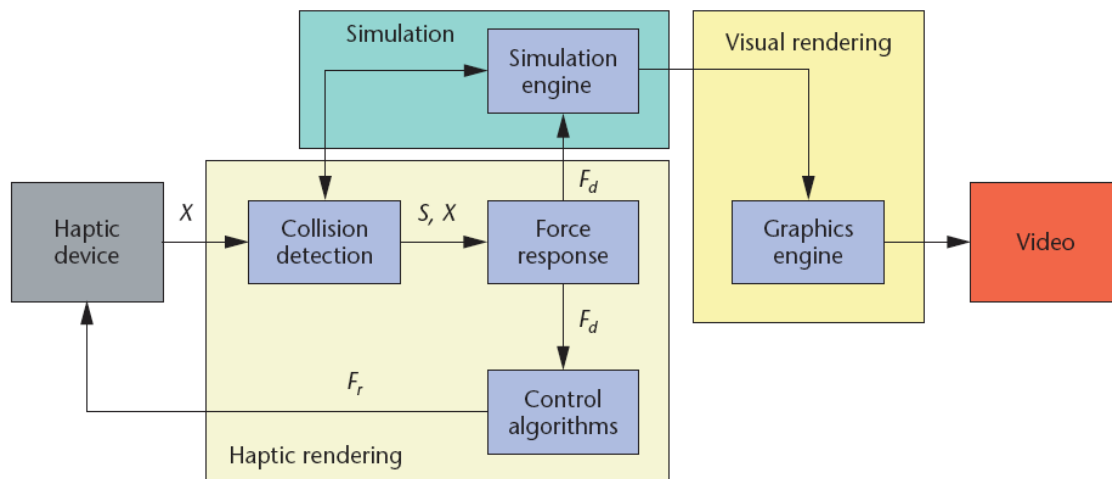


Figure 3.4. Organisation d'une architecture traditionnelle pour la simulation haptique – extrait de [Salisbury et al., 2004]

de la transformation géométrique des informations issues des capteurs de position du système gestuel afin de fournir la position de l'avatar en fonction du repère géométrique dans le système de modélisation.

L'algorithme de contrôle se trouve toujours calculé dans la boucle de commande, en synchronisme total avec les entrées / sorties du système gestuel.

L'algorithme de détection de collision (*collision detection* – figure 3.4) vérifie si l'avatar a pénétré dans un des objets du modèle simulé. Dans ce cas, il détermine quelle seront les surfaces de référence à partir desquelles seront calculées les forces d'interaction.

L'algorithme de réponse en force (*force response* – figure 3.4) calcule les forces d'interaction, c'est-à-dire la consigne en force qui sera retournée au système gestuel. Si l'algorithme de détection de collision peut être basé sur une modélisation géométrique, cet algorithme est nécessairement basé sur un principe de modélisation physique, mais qui doit rester simple pour des raisons d'efficacité computationnelle.

Suivant les pratiques, les algorithmes de détection de collision et de réponse en force seront exécutés dans la boucle de commande synchrone [Salisbury *et al.*, 2004], soit dans un processus rapide mais néanmoins plus lent que la boucle de commande [Salisbury *et al.*, 2004], soit dans la boucle lente [Ruspini *et al.*, 1997].

3.2.2 Boucle lente

La boucle lente est désynchronisée de la boucle de commande. Cette boucle prend en charge le calcul de simulation du reste du modèle (*simulation engine* – figure 3.4). Le moteur graphique est également exécuté dans cette boucle (*graphics engine* – figure 3.4).

La vitesse d'exécution de cette boucle dépend de la vitesse des processus graphiques. Elle fonctionne le plus souvent aux environs de quelques dizaines de Hz pour la simulation d'objets

rigides [Salisbury *et al.*, 2004], quelques centaines de Hz pour la simulation d'objets déformables [Cotin *et al.*, 1999, Sreng *et al.*, 2007].

3.2.3 Communication entre la boucle de commande et la boucle lente

Dans la plupart des implantations, la communication entre la boucle de commande et la boucle lente est assurée via une mémoire partagée [Ellis *et al.*, 1996, Burdea, 1996]. Les informations sont communiquées sur un bus VME ou PCI [Adachi *et al.*, 1995, Ellis *et al.*, 1996, Force Dimension, 2008]⁶, parfois par une communication asynchrone sur socket TCP/IP via Ethernet [Berkelman *et al.*, 1996, Mark *et al.*, 1996, der Linde *et al.*, 2002, fcs, 2008].

3.2.4 Approches multi-fréquences et stabilité

Les approches multi-fréquences qui implantent les composants logiciels de simulation dans différents processus (le plus souvent asynchrones) sont confrontées à un problème de stabilité supplémentaire dû à la mise en relation de deux boucles de calcul asynchrones dont les fréquences de rafraîchissement sont différentes. Plusieurs approches ont proposé des techniques permettant d'améliorer la stabilité générale du système de simulation haptique :

- En implantant des algorithmes d'interpolation du signal basse fréquence de la boucle de simulation (quelques dizaines de Hertz) à la fréquence de la boucle de commande haptique (exécutée le plus souvent à 1 kHz) [Cavusoglu et Tendick, 2000],
- Par des techniques plus complexes utilisant des représentations intermédiaires : par exemple, le *forcegrid* [Mazzella *et al.*, 2002] est une structure de données permettant l'interpolation ou l'extrapolation de signaux, dans le but de faire de l'adaptation en fréquence entre deux boucles de calcul asynchrones exécutées à des fréquences différentes. Bi et al. ont également proposé une méthode de couplage virtuel par logique floue [Bi *et al.*, 2005].
- Ou en séparant le modèle simulé en plusieurs boucles calculées à différentes fréquences de simulation [Astley et Hayward, 1998, Florens, 2003].

3.3 Réalité Virtuelle

Dans le domaine de la réalité virtuelle, les performances des calculateurs pour la simulation interactive sont évaluées dans les années 1990 sur la base des performances des architectures de rendu graphique : en matière d'architecture de calcul pour la réalité virtuelle, Burdea [Burdea, 1996] mentionne exclusivement les architectures pour le rendu graphique, pour la représentation de scènes visuelles, et pour leur connexion avec les dispositifs de visualisation. En effet, les premières applications de la réalité virtuelle étaient essentiellement focalisées sur une immersion *visuelle* de l'utilisateur, la stimulation tactile ou kinesthésique étant appliquée pour renforcer ce sentiment d'immersion. Les architectures de réalité virtuelles sont souvent complexes car elles doivent communiquer avec plusieurs périphérique (pour l'affichage, le son, etc.), et doivent supporter plusieurs boucles de calcul. Les principes de calcul réparti asynchrone que nous avons

⁶Ces types de bus sont propices à l'utilisation d'une mémoire partagée puisqu'ils permettent la projection d'une zone mémoire contenue dans l'espace d'adressage du périphérique dans la mémoire d'un processus de la machine hôte.

évoqués ci-dessus sont aussi appliqués en Réalité Virtuelle et étendus à l'utilisation de plusieurs unités de calcul connectées par réseau.

L'architecture de simulation utilisée pour les premiers prototypes du Rutgers Portable Master est représentative des systèmes construits pour la Réalité Virtuelle au début des années 1990 [Burdea et Speeter, 1989, Burdea, 1996]. Il s'agit d'une architecture de simulation distribuée client-serveur, où les calculs sont répartis sur quatre stations de travail connectées par liaison ethernet. En particulier, cette architecture comprend une machine Sun 4/380 spécialisée pour la connexion avec le système à retour d'effort : cette station permet la « remise à jour du niveau des forces de sortie, et la maintenance de l'information d'état de tous les objets du monde virtuel ». Dans cette architecture, deux boucles de calcul asynchrones sont exécutées pour la simulation :

- La boucle de contrôle, en relation avec le système à retour d'effort ; c'est aussi cette boucle qui calcule le comportement du modèle simulé.
- Une boucle spécifique pour l'affichage graphique de la scène calculée.

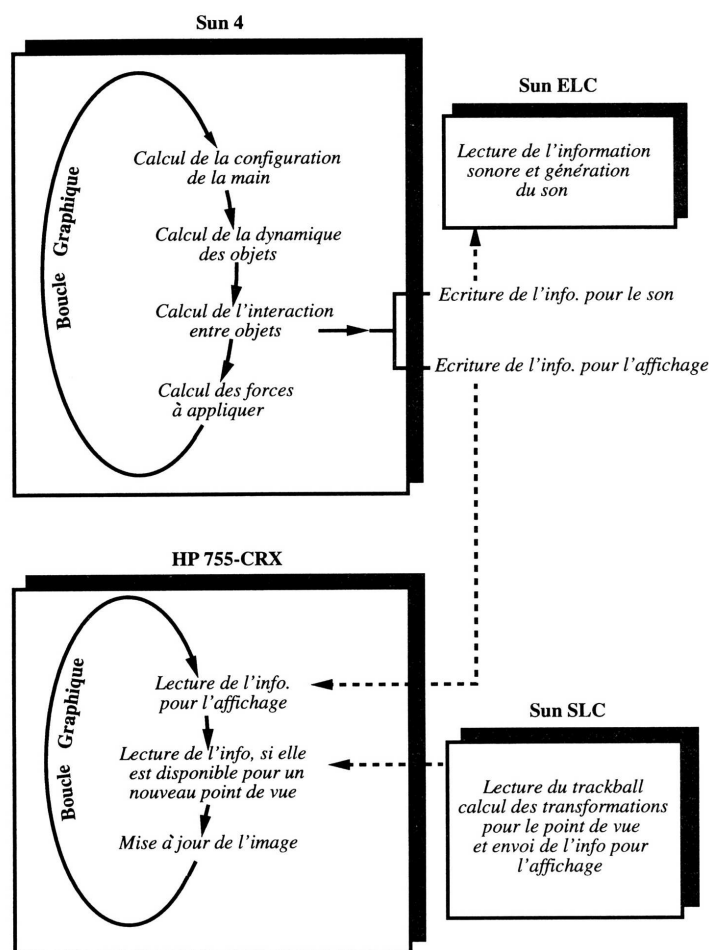


Figure 3.5. Architecture de simulation pour le Rutgers Portable Master – extrait de [Burdea, 1996]

La puissance des architectures de calcul actuelles a permis de porter les scènes de réalité virtuelle à un degré de complexité et de détail inespéré lors des balbutiements des Réalités Virtuelles. Malgré les améliorations constantes des performances de calcul des machines actuelles, les simulations de RV actuelles font toujours l'utilisation massive de plusieurs boucles de calcul asynchrones, parce

que la complexification des environnements simulés est allée de pair avec cette augmentation de la puissance des calculateurs.

Pour la simulation d'opérations chirurgicales par exemple, les modèles d'organes sont obtenus à partir de scanners médicaux, constituant une masse de données importante à traiter dans la simulation. Cotin et al. [Cotin *et al.*, 1999] ont implémenté une technique permettant la dissociation des calculs de déformations des tissus (boucle lente à 50 Hz), le calcul des interactions au point de manipulation (boucle plus rapide à 300 Hz) et la communication avec le TGR (500 Hz).

3.4 Performances des architectures de simulation actuelles

Il n'était pas imaginable, il y a dix ans, que la puissance des machines de calcul permette de simuler un modèle comportant plus de 10000 éléments avec une fréquence de rafraîchissement supérieure à 30 Hz. Selon la loi de Moore, la puissance des processeurs croît de façon exponentielle⁷, et croît d'un facteur 10 environ tous les 10 ans. Comment les gains en performances se sont-ils répercutés sur les performances des simulateurs actuels ?

À une époque où les boucles de simulation pour le geste fonctionnaient à des fréquences de quelques dizaines de Hertz, l'ACROE a, dès les premiers travaux de recherche, mis en avant la nécessité de l'utilisation d'une fréquence de simulation relativement importante. Les simulations furent dans un premier temps exécutées à 300 Hz sur une architecture AP120/PDP11 au début des années 1980, puis à 1 kHz avec une boucle de simulation totalement synchrone dès 1987, toujours sur le processeur vectoriel AP120 [Gibet, 1987].

Il semble que pour la majeure partie de la communauté haptique, les critères de performances en termes de fréquence de simulation n'ont pas changé : le fameux « 1 kHz » reste la référence. Ainsi, Duriez et al. écrivent en 2006 :

« Haptic rendering for rigid body simulation requires update rates ranging from 500 Hz to 1 kHz. »

C. Duriez et al. [Duriez et al., 2006]

Si le gain en puissance apporté par les nouvelles architectures de calcul n'a pas eu de répercussions sur la fréquence de simulation, par contre, il a eu pour effet d'augmenter la complexité des environnements simulés et des techniques de simulation et de modélisation. Duriez et al. présentent la simulation d'objets déformables en interaction complexe (i.e. en interaction par plusieurs points de contact potentiels) [Duriez *et al.*, 2006]. La boucle de commande au système haptique est exécutée à 250 Hz, et un pas de simulation, exécuté dans une seconde boucle asynchrone, occupe en moyenne entre 3000 et 4 000 μ s. Duriez et Andriot présentent une technique de simulation comportant plusieurs fils d'exécution parallèles, et où la boucle de simulation et la boucle de communication avec le TGR sont synchrones, et exécutées respectivement à 1000 et

⁷Gordon Moore a exprimé en 1965 une loi selon laquelle la densité des composants dans une puce microélectronique doublerait tous les 18 mois. Il a raffiné son jugement en 1975 en postulant que le nombre de transistors dans les circuits imprimés double tous les deux ans. Cette extrapolation empirique s'est révélée jusqu'à présent relativement exacte, la densité des transistors doublant chaque 1,96 année. Quoique ce ne soit pas tout à fait exact, par rapprochement entre le nombre de transistors contenus dans un processeur et sa puissance de calcul (en termes du nombre d'opérations par seconde), on simplifie souvent cette loi en statuant que la puissance des processeurs double tous les deux ans.

500 Hz [Duriez *et al.*, 2004]. Les techniques de simulation se sont améliorées et permettent la simulation d'objets complexes déformables, la simulation de phénomènes de contact, de friction et leur représentation visuelle de manière « réaliste » [Sreng *et al.*, 2007].

Les performances des ordinateurs standards actuels permettent de simuler une boucle de calcul relativement complexe avec une période de l'ordre de la milliseconde. La plupart des architectures de simulation haptique peuvent aujourd'hui faire tourner une boucle de simulation haptique à 1 kHz, qui pourra être synchrone avec les entrées/sorties du TGR [Nikolakis *et al.*, 2004]. Dans ce cas, la boucle graphique est toujours calculée à quelques dizaines de Hertz, puisque les dispositifs d'affichage ont une fréquence de rafraîchissement du même ordre de grandeur.

De Boeck et l'équipe de recherche du *Expertise center for Digital Media* (EDM) de l'Université de Hasselt ont réalisé un environnement pour l'interaction multimodale [de Boeck, 2006]. L'objectif de ce travail était de permettre dans un premier temps l'interaction à deux mains d'un utilisateur avec un environnement virtuel, ce qui suppose l'utilisation de deux TGR, puis d'étendre l'environnement de simulation à plusieurs postes de travail. Faisant l'observation qu'un simulateur ne peut supporter plus d'un seul TGR sans subir une perte de performance au détriment des processus de simulation, la communication avec chaque TGR est supportée par un simulateur différent. La communication entre les simulateurs est assurée par un réseau Ethernet sur protocole UDP. L'architecture logicielle de chaque simulateur est multithreadée, un thread principal supervisant l'exécution des autres threads (pour la simulation du modèle mécanique et la communication avec le TGR, pour la gestion de la partie graphique, pour l'émission sur le réseau Ethernet, pour la réception sur le réseau Ethernet). La communication entre les threads est asynchrone, basée sur un mécanisme de répartiteur de messages (*message dispatcher*). Enfin, chaque simulateur communique le nouvel état de son modèle courant à la fréquence de rafraîchissement vidéo, soit 25 Hz. Nous questionnons la pertinence de cette approche pour la réalisation de simulation multisensorielles (geste, son, vidéo) — l'exemple de test de ces travaux est un vibraphone virtuel —, où le calcul de simulation est réparti sur plusieurs simulateurs non synchronisés, où la simulation est elle-même répartie sur plusieurs unités de simulation spécialisées non synchrones. Typiquement sur cette architecture, la mise à jour du modèle mécanique calculé sur le simulateur A vers le simulateur B doit successivement parcourir les intermédiaires suivants : thread de simulation du modèle mécanique de A, message dispatcher de A, thread réseau pour l'émission de A, support Ethernet, thread réseau pour la réception de B, message dispatcher de B et enfin thread de simulation du modèle mécanique de B.

4 Le contrôleur haptique

4.1 Introduction

Nous avons vu précédemment que les approches de type *haptic rendering*, proches aussi dans une certaine mesure des approches de type *Réalité Virtuelle*, font l'utilisation de deux boucles de calcul asynchrones. Une boucle de commande est en relation directe avec les entrées / sorties du TGR, alors que la boucle lente se charge du calcul de simulation du modèle virtuel (hors partie haptique). Dans la littérature, le terme contrôleur haptique (*haptic controller*) désigne dans le domaine de la robotique le bras haptique complet, constitué de capteurs, d'actionneurs et d'une électronique de

contrôle, et dans le domaine de l'informatique le composant électronique chargé de la régulation des consignes en entrée du système⁸, chargé de l'exécution de boucles de contrôle. Le terme *haptic controller* tel que nous l'employons ici s'entend donc dans ce deuxième sens.

L'objet de cette partie est l'étude des différentes approches d'implantation de la boucle de contrôle par le contrôleur haptique. Nous allons aussi voir quel rôle et quelles opérations sont prises en charge par ce composant de l'architecture de simulation.

Le principe du contrôleur haptique est dans un premier temps de jouer le rôle d'interface entre les entrées / sorties du TGR et celles de l'algorithme de simulation, mais aussi d'exécuter des algorithmes relativement peu coûteux en calculs permettant à la boucle de contrôle de s'exécuter avec une fréquence de rafraîchissement relativement rapide (de l'ordre du kHz). Les opérations généralement exécutées sont :

- La régulation par algorithme numérique de la consigne provenant du processus de simulation. S'il est en effet relativement simple de concevoir une électronique analogique comprenant une boucle de régulation du signal de consigne, on trouve souvent des boucles de régulation implantées numériquement par exemple parce que les actionneurs prennent un signal d'entrée, ou parce qu'il est relativement facile aujourd'hui d'avoir à sa disposition des processeurs embarqués suffisamment puissants pour ce genre de tâche.
- Les transformations cinématiques, lorsque la cinématique du TGR est complexe.
- Le calcul des forces d'interaction mécaniques de la boucle de commande dans l'approche haptic rendering (au-dessus). Dans ce cas, l'algorithme de la boucle de commande met à jour la consigne de force à retourner au système gestuel, en fonction de la comparaison entre la position mesurée, et la position de référence fournie par l'application utilisateur (le programme de simulation de l'environnement virtuel – figure 3.6).

D'une manière générale, la boucle de calcul exécutée par le contrôleur haptique, chargée de la communication en temps réel avec le TGR, comprendra les composants suivants [Hollis et Salcudean, 1992] (*I/O device software* – figure 3.6) :

- Gestion des entrées/sorties en temps réel (*realtime I/O*) : ce composant est chargé de l'échange des données avec le TGR : lecture et écriture dans les mémoires tampon des convertisseurs, déclenchement des conversions, etc.
- Algorithmes de contrôle : il s'agit de la transformation cinématique des données de position obtenues du TGR dans le repère de modélisation. Les algorithmes de contrôle peuvent également assurer la réduction de l'espace de travail en contraignant algorithmiquement certains degrés de liberté du système gestuel.
- Bibliothèque d'objets de contrôle : une bibliothèque de transformateurs cinématiques contient les différents types de données et de codes. Ce composant peut être en relation avec l'application utilisateur.

⁸Dans le domaine informatique, un contrôleur est un dispositif électronique spécialisé dans la gestion d'un périphérique particulier :

- contrôleur graphique, chargé de la gestion de la communication des données à un écran ou une sortie vidéo,
- contrôleur réseau.

Par abus de langage, on parle souvent de carte Ethernet ou de carte graphique, pour signifier respectivement un contrôleur Ethernet ou un contrôleur graphique.

- Mesure de la position du système gestuel
 - Déclenchement des conversions A/N
 - Lecture des buffers des convertisseurs A/N
- Envoi de la position mesurée au programme utilisateur
- Comparaison de la position mesurée à la position de référence fournie par l'application utilisateur
- Mesure de l'erreur de position
- Calcul de la nouvelle consigne en force en fonction du vecteur de position
- Renvoi de la consigne en position
 - Écriture des buffers des convertisseurs N/A
 - Déclenchement des conversions N/A

Figure 3.6. Algorithme de contrôle de la boucle de commande (adapté de [Hollis et Salcudean, 1992])

La boucle lente contient les composants de l'application plus spécifiques à la simulation de l'environnement virtuel et à l'interaction avec l'utilisateur : édition de modèles, etc. (*Workstation software* – figure 3.6)) :

- Les *application bindings*, assurant la communication des données entre le composant de simulation au niveau utilisateur, et le bloc de contrôle du TGR dans la boucle de commande.
- Des modèles virtuels du TGR : ces objets sont en relation avec la bibliothèque d'objets de contrôle de la boucle de commande,
- Un éditeur de modèle,
- Le programme utilisateur de simulation et la gestion de son interface.

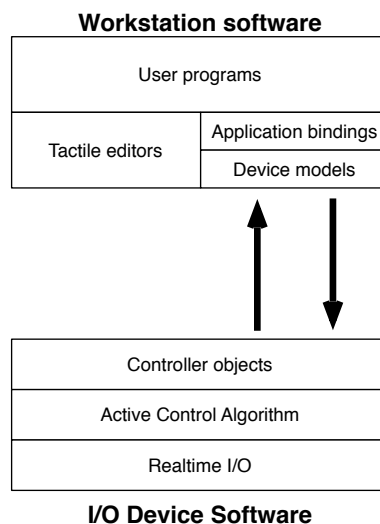


Figure 3.7. Composants logiciels de la boucle lente (workstation software) et de la boucle de commande (I/O device software) – extrait de [Hollis et Salcudean, 1992]

Comme nous allons le voir par la suite, la plupart des implantations du contrôleur haptique ne

mettent en œuvre que les composants (1) gestion des entrées/sorties en temps réel du TGR et (2) algorithmes de contrôle.

4.2 Exemples d'implantations du contrôleur haptique

Dans la plupart des architectures pour la simulation haptique, le contrôleur haptique est exécuté sur un processeur embarqué, à proximité de l'électronique de communication avec le TGR.

4.2.1 Rutgers Master II-ND – Rutgers University of New Jersey

Burdea et son équipe ont développé à l'université de Rutgers un gant à retour d'effort à cinq degrés de liberté [Burdea et Speeter, 1989, Burdea, 1996]. Des pistons pneumatiques sont utilisés comme actionneurs et sont contrôlés par un signal de commande obtenu par un contrôleur PWM. Le gant est équipé de capteurs de position LVDT pour les premières versions [Burdea et Speeter, 1989], puis de capteurs à effet Hall [Bouzit *et al.*, 2002b]. Aussi, des capteurs de pression sont utilisés afin de réguler la boucle de force. Le gant à retour d'effort développé est contrôlé par une interface électronique appelée « Haptic Control Interface », qui se compose des éléments suivants :

- Un ordinateur embarqué de type Pentium 233 MHz, comprenant des interfaces PC104, IDE, VGA et ethernet, prenant en charge le composant « contrôleur haptique » de l'architecture
- Des convertisseurs analogique/numérique (carte MPC550 de Micro/Sys), comprenant 16 canaux d'entrée et 8 canaux de sortie. Cette carte est montée sur le bus PC104 du Pentium embarqué
- Une électronique de filtrage, en amont des convertisseurs A/N (filtrage, amplification et multiplexage des signaux analogiques)
- Une électronique de puissance, pour l'amplification des signaux en sortie des convertisseurs N/A
- Le ordinateur Pentium embarqué communique de manière asynchrone avec la machine hôte par une liaison série RS232 (vitesse de transmission entre 38 400 et 115 200 bauds)

Le contrôleur haptique traite les données de position en provenance des convertisseurs A/N, échantillonnées à 1 000 Hz. Il effectue les tâches suivantes :

- Filtre et calibre les signaux de position en entrée
- Applique une transformation cinématique
- Sous-échantillonne les signaux en fonction de la fréquence de calcul de la boucle de contrôle, et transmet les données de position soit (1) à la boucle de commande sur le Pentium embarqué, soit (2) à une boucle de contrôle exécutée sur la machine hôte.

Cette architecture permet deux modes de simulation. Le premier permet de synchroniser le processus de simulation avec les signaux de position et de force ; le second désynchronise le processus de simulation des entrées/sorties [Popescu *et al.*, 1999] :

1. Le modèle mécanique simulé est pris en charge localement par le contrôleur haptique, mais peut être paramétré depuis la machine hôte. La fréquence de simulation est de 500 Hz, mais la complexité du modèle simulable est limitée par la faible puissance du ordinateur embarqué.
2. Le processus de simulation est exécuté sur la machine hôte. Dans ce cas, le signal de force en provenance de la machine hôte est transmis via le port série de manière asynchrone. Aussi, la

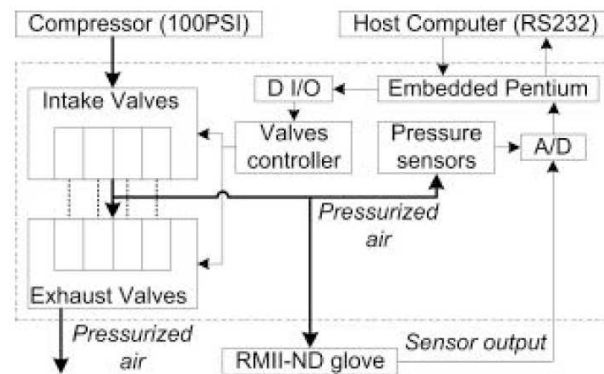


Figure 3.8. Diagramme fonctionnel du bloc « contrôleur haptique » [Bouzit et al., 2002b]

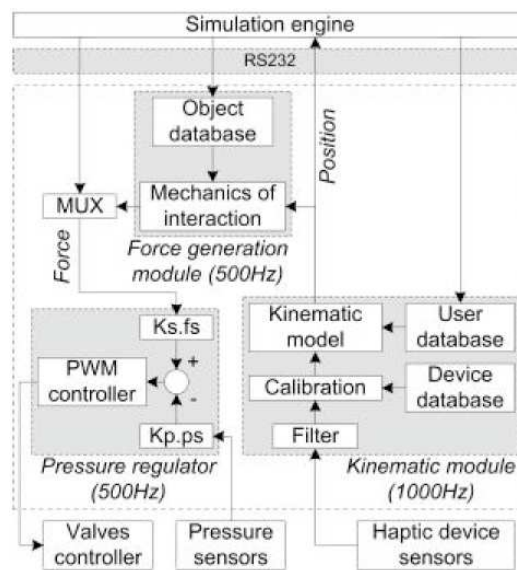


Figure 3.9. Détail - Diagramme de la boucle de servo-contrôle [Bouzit et al., 2002b]

fréquence de calcul du modèle simulé tombe à 100 Hz en raison du mode de communication par liaison série.

4.2.2 MagLev – Robotic Institute – Carnegie Mellon University

Ralph Hollis, Septimiu Saldudean et al. ont développé un système haptique utilisant les forces de Lorentz [Hollis et Saldudean, 1993]. Ce système fut ensuite amélioré par Peter Berkelmann sous la direction de Ralph Hollis [Berkelman et Hollis, 1997]. Il s'agit d'un bol équipé de bobines maintenu en lévitation par les forces électro-magnétiques, dont les 6 degrés de liberté en mouvement sont contrôlés par les courants parcourant les bobines électromagnétiques entourant le bol.

Dans ce système, l'algorithme de contrôle est exécuté par un processeur spécifique embarqué (DSP). Le DSP contrôle aussi les convertisseurs nécessaires à la communication avec les entrées / sorties du système haptique (capteurs LVDT et bobines électromagnétiques).

L'application de simulation est exécutée sur une station de travail indépendante plus puissante. Les commandes de l'application de simulation, les paramètres du modèle sont mis à jour de manière asynchrone et avec une fréquence de rafraîchissement beaucoup plus faible que la fréquence de simulation (de l'ordre du Hz, jusqu'à 10 Hz).

Le contrôleur haptique communique avec l'application graphique utilisateur via une connexion Ethernet sur sockets TCP/IP.

Les dernières améliorations sur ce système [Unger *et al.*, 2003] ont porté la fréquence de fonctionnement du contrôleur haptique jusqu'à 4 kHz, et la fréquence de la mise à jour des données communiquées entre la boucle de commande et l'application utilisateur jusqu'à 1 000 Hz.

4.2.3 Delta Haptic Device – ForceDimension

Le système haptique « Delta » est développé à l'EPFL [Grange *et al.*, 2001a, Grange et Conti, 2001, Grange *et al.*, 2001b]. Dès 2001, la société Force Dimension fut créée afin de commercialiser ce système haptique.

Le bras haptique est relié au PC hôte par l'intermédiaire d'une connexion USB 2.0. Le contrôleur haptique est intégré dans l'électronique du système haptique ; l'application de contrôle est exécutée sur un DSP qui assure les fonctionnalités suivantes de communication avec les capteurs et actionneurs, et de sécurité :

- Coupe les actionneurs en cas de vitesse de mouvement trop rapide de la nacelle (une vitesse trop élevée étant considéré comme un signe d'instabilité)
- Coupe les actionneurs en cas de sortie de l'espace de travail ; la position de la nacelle ne doit pas atteindre les limites mécaniques de l'espace de travail, sous peine de mettre en danger la structure du système
- Vérification du fonctionnement du PC : si la boucle du PC hôte stoppe, le contrôleur arrête immédiatement le fonctionnement de la simulation : le taux de fonctionnement de la boucle maintenue par le PC hôte est au minimum 100 Hz.

L'application de simulation est exécutée en intégralité sur le PC hôte. Le contrôleur haptique remplit donc seulement les fonctionnalités de contrôle et de sécurité, et d'interface pour la communication des données entre le PC hôte et le système haptique lui-même. L'application de simulation est séparée en deux boucles, à la manière de [Ruspini *et al.*, 1997] :

- Une boucle de commande (env. 1 kHz), synchronisée sur les entrées/sorties du contrôleur haptique
- Une boucle lente (appelée global force interaction low-frequency loop [Grange *et al.*, 2001a]), traitant la simulation de l'ensemble du modèle ainsi que l'application graphique.

4.2.4 High Performance Haptic Interface – Queen's University, Kingston

Ellis et al. décrivent la conception et l'évaluation d'un système à retour d'effort à 3 degrés de liberté, permettant des mouvements dans un plan horizontal de 15×15 cm [Ellis *et al.*, 1996] (le troisième degré de liberté actif étant l'axe de rotation du manipulandum). Le contrôle du système et l'application de simulation sont dissociés en deux composants distincts :

- Le logiciel de contrôle est embarqué dans deux DSP (TMS320C30 de Texas Instruments, 34 Mflops), faisant l'interface avec les convertisseurs D/A et A/D, et permettant de relier le système à un second système haptique maître ou esclave, pour des applications de téléopération ; la boucle de contrôle est exécutée à 1 200 Hz, mais la bande passante des actionneurs n'est que de 85 Hz
- La simulation de l'environnement virtuel est assurée par la machine hôte (SUN SPARC-2)
- La communication entre la machine hôte et les deux DSP de contrôle s'effectue via une mémoire partagée.

4.2.5 L-Exos – PERCRO

L-Exos est un exosquelette haptique adapté aux mouvements de bras, développé par Antonio Frisoli et al. au laboratoire PERCRO [Frisoli *et al.*, 2005]. Ce système comprend 5 degrés de liberté, dont 4 actifs. Outre les problèmes technologiques liés à la conception mécanique d'un exosquelette (par exemple : nombreux degrés de liberté en série rendant la cinématique complexe), la boucle de contrôle haptique est d'autant plus coûteuse en calculs que la cinématique du bras haptique est complexe.

Une des particularités de structures cinématiques telles que les exosquelettes est que le contrôleur haptique doit effectuer des calculs complexes dans la boucle de contrôle (calculs matriciels, en particulier le Jacobien), et ceci à une fréquence de rafraîchissement que l'on souhaite aussi rapide que possible.

La boucle de contrôle du L-Exos est exécutée à 1 kHz sur une machine relativement puissante (Pentium IV 2,4 GHz – système temps réel) et est cadencée de manière précise grâce à l'utilisation d'un système d'exploitation temps réel. Les entrées/sorties vers les actionneurs et les capteurs du bras sont eux-aussi cadencés à 1 kHz en temps réel.

La connexion avec le modèle est assurée par la méthode god-object de Zilles et Salisbury [Zilles et Salisbury, 1995], et le signal d'erreur de position en entrée de la boucle de contrôle est rafraîchi à la fréquence de calcul du modèle virtuel : 20 Hz (celui-ci étant cadencé sur la boucle graphique).

4.2.6 Haptic Master – Moog FCS

Le Haptic Master est commercialisé par la société Moog FCS. Ce système fonctionne en mode admittance, à l'encontre de la majorité des autres systèmes haptiques prototypes ou commerciaux, et jouit d'une certaine réputation dans la communauté haptique de par sa capacité à rendre des contacts rigides ou des inerties importantes. De fait, il peut retourner des forces jusqu'à 250 N, ce qui est encore rare compte tenu des dimensions de l'espace de travail qu'il permet : 0,36 m × 0,40 m × 1.0 rad.

Le contrôle du système haptique est assuré par une boucle numérique, exécuté en même temps que la boucle de rendu haptique sur une machine dédiée à 2 500 Hz, sur le système temps réel VxWorks.

La boucle de simulation de l'environnement virtuel est exécutée indépendamment sur un autre ordinateur (fréquence de simulation inconnue) sur OS temps réel. Elle communique avec le « serveur haptique » par le biais d'une connexion Ethernet.

4.2.7 SPIDAR-G – Tokyo Institute of Technology

Le SPIDAR-G est un système haptique basé sur la transmission du mouvement par câble. Dans la première version, présentée en 2002 puis 2004 [Murayama *et al.*, 2004], le système d'actuation est basé sur des moteurs DC, et les informations de position sont obtenues par la mesure de la longueur de chaque câble, par l'intermédiaire d'encodeurs incrémentaux. Des convertisseurs A/N et N/A sont reliés au PC hôte par l'intermédiaire d'un bus VME.

La nouvelle version du SPIDAR-G présentée en 2006 permet de travailler avec des signaux d'entrée/sortie échantillonnés à 5 kHz [Akahane *et al.*, 2005], puis 10 kHz [Akahane *et al.*, 2006].

Dans ce système, une attention particulière a été portée au contrôleur afin de pouvoir interfacer le bras haptique, dont les entrées/sorties sont échantillonnées à 10 kHz avec un système de modélisation travaillant à 1 kHz [Akahane *et al.*, 2006].

Le contrôleur haptique comprend :

- une unité de calcul (processeur SuperH4, Renesas Technology⁹)
- un port USB2.0 pour la communication avec le PC hôte équipé d'un contrôleur permettant des transferts à faible latence
- un FPGA chargé du contrôle de l'encodeur (signaux de sortie du système haptique – le SPIDAR-G)
- des convertisseurs N/A, faisant l'interface entre le processeur SuperH4 embarqué dans le contrôleur haptique, et les actionneurs du SPIDAR-G (moteurs DC).

La relation entre l'interface avec le système haptique, dont les entrées/sorties sont cadencées à 10 kHz, et le PC hôte, dont les entrées/sorties sont cadencées à 1 kHz, est assurée par un algorithme d'interpolation asynchrone implémentant une interaction de type viscoélasticité (méthode du couplage virtuel – Colgate).

4.3 LHIFAM – CEIT

Le système LHIFAM fut développé au laboratoire CEIT par Joan Savall et al. Il s'agit d'un système conçu pour fournir à l'utilisateur 6 degrés de liberté dans un grand espace de travail (un cylindre de 0,75 m de rayon sur une hauteur de 1,5 m). Ce système fut initialement conçu pour la simulation de maintenance dans l'aéronautique, les pièces manipulées pouvant avoir des dimensions de plusieurs mètres. Pour de telles applications, les interactions de l'utilisateur avec son environnement de travail sont essentiellement de type contact [Savall *et al.*, 2002]. Le réalisme de la simulation sera donc essentiellement jugé sur la qualité des mouvements possibles pour l'utilisateur (déterminés par les dimensions de l'espace de travail et la cinématique du bras haptique) et la fidélité de la représentation graphique de l'environnement simulé. En conséquence, l'effort de conception fut essentiellement porté sur la mécanique du système et sur le moteur de rendu graphique.

⁹Les processeurs SuperH sont fabriqués par la compagnie japonaise Hitachi (Renesas technology). Il s'agit de processeurs 32 bits basés sur une architecture RISC. Le processeur SuperH4 est un processeur super-scalaire (2-way) équipé d'une unité arithmétique au format flottant, d'une mémoire cache et d'une MMU (Memory Management Unit).

L'environnement de simulation est fortement dérivé des applications de l'infographie : deux ordinateurs sont utilisés, connectés par liaison Ethernet sur protocole UDP. Le premier ordinateur exécute la boucle de contrôle du bras haptique à 1 000 Hz, tandis que le deuxième ordinateur calcule le rendu graphique de la simulation et l'algorithme de détection de collisions à une fréquence plus faible.

67

Nom du système, et référence biblio.	Type de processeur (contrôle haptique)	Fréquence de la boucle de commande (kHz)	Architecture de calcul pour l'application de simulation	Fréquence de la boucle de simulation	Liaison avec les E/S du TGR	Liaison avec le processus de simulation	Techno. des actionneurs	Techno. des capteurs
MagLev, [Hollis et Salcudean, 1993]	Carte DSP Mercury 3200 (bus VME)	1 kHz	IBM RS/6000 workstation	NC	Carte fille avec CNA/CNA sur le bus de la carte DSP	Mémoire partagée	Bobines électromagnétiques	Photodiodes
MagLev, [Berkelman et Hollis, 1997]	Carte DSP Mercury 3200 (bus VME)	1 kHz	Sun3	50 Hz	Carte fille avec CNA/CNA sur le bus de la carte DSP	Mémoire partagée (bus VME), Ethernet via sockets TCP/IP	Bobines électromagnétiques + ampli PWM	Photodiodes
High performance Haptic Interface [Ellis <i>et al.</i> , 1996]	Deux cartes avec DSP TMS320C30 (TI) [34 Mflop] sur bus VME ^(α)	1,2 kHz	SUN Sparc-2	NC	Carte VME CNA/CAN [12 bits]	Mémoire partagée (bus VME)	Moteurs à balais + linear current amplifiers	LVDT

Suite page suivante . . .

Suite de la page précédente								
Nom du système, et référence biblio.	Type de processeur (contrôle haptique)	Fréquence de la boucle de commande (kHz)	Architecture de calcul pour l'application de simulation	Fréquence de la boucle de simulation	Liaison avec les E/S du TGR	Liaison avec le processus de simulation	Techno. des actionneurs	Techno. des capteurs
Portable Rutgers Master [Bouzit <i>et al.</i> , 2002b, Bouzit <i>et al.</i> , 2002a]	Pentium 233 MHz	1 kHz	Dual-proc. Pentium III 933 MHz	500 Hz sur processeur embarqué, 100 Hz sur la machine hôte	Carte MPC550 Micro/Sys ; CNA/CAN sur bus PC104	Liaison série RS232 asynchrone [38 400 à 115 200 bauds]	Pistons pneumatiques + contrôleur PWM	LVDT puis capteurs à effet Hall
Delta Haptic Device [Grange <i>et al.</i> , 2001b]	Carte mère embarquée ; proc. Inconnu	1 à 4 kHz ; security watchdog ^(β)	Suivant application	Suivant application	NC	Liaison USB 2.0	Amplificateurs PWM	Encodeurs optiques
Phantom, Force Dimension	NC	1 kHz	Suivant application	Suivant application	NC	Parallèle, USB2.0 ou FireWire		
Pantograph, McGill [Ramstein et Hayward, 1994]		10 kHz	Ordinateur grand public (1994) : IBM 486 DX2/66 MHz	10 kHz	Deux cartes CAN et CNA [12 bits]	Pas de liaison	Ampli. de courant linéaires	
L-Exos [Frisoli <i>et al.</i> , 2005]	Pentium IV 2,4 GHz (RTOS)	1 kHz	NC	20 Hz	Deux cartes I/O 8 canaux	Liaison Ethernet via protocole UDP/IP	DC Permanent Torque Motor	Encodeur optique
HapticMaster, MoogFCS [fcs, 2008]	Carte mère embarquée (VxWorks)	2,5 kHz	Suivant application	Suivant application	NC	Liaison Ethernet	NC	NC
Suite page suivante ...								

<i>Suite de la page précédente</i>								
Nom du système, et référence biblio.	Type de processeur (contrôle haptique)	Fréquence de la boucle de commande (kHz)	Architecture de calcul pour l'application de simulation	Fréquence de la boucle de simulation	Liaison avec les E/S du TGR	Liaison avec le processus de simulation	Techno. des actionneurs	Techno. des capteurs
Spidar-G	SuperH4 ; [7Mflops] + FPGA ⁽⁷⁾	10 kHz	PC grand public	1 kHz	CNA pour consigne moteur, interface numérique pour encodeur	USB 2.0 (contrôleur EZ-USB FX2)	Moteur DC	Encodeur à incrémentation (numérique)
Lhifam, CEIT [Savall <i>et al.</i> , 2002, Borro <i>et al.</i> , 2004]	Pentium II 233 MHz	1 kHz	Pentium III Xeon + Intense 3D Wildcat 4210	10-100 Hz (calcul collisions) ; 20-50 Hz (visualisation)		Ethernet : contrôle de la simulation (TCP) ; informations de collision (UDP)	Moteur DC Maxon RE40 148877	Encodeurs optiques
High Fidelity USB Force Feedback Joystick [Jöhl <i>et al.</i> , 2005]	Freescall DSP	1 kHz	NC	NC		USB 2.0 (contrôleur USB spécialisé pour latence faible : temps de cycle 1 ms)	Brushless DC Motors + PWM	Encodeurs optiques (moteurs) ; potentiomètres (articulations)
<i>Suite page suivante ...</i>								

<i>Suite de la page précédente</i>								
Nom du système, et référence biblio.	Type de processeur (contrôle haptique)	Fréquence de la boucle de commande (kHz)	Architecture de calcul pour l'application de simulation	Fréquence de la boucle de simulation	Liaison avec les E/S du TGR	Liaison avec le processus de simulation	Techno. des actionneurs	Techno. des capteurs

- ^(α) Un DSP est utilisé pour l'algorithme du contrôleur haptique : le second est utilisé pour l'asservissement éventuel du bras haptique sur un système maître.
- ^(β) Le security watchdog a pour rôle de couper la simulation si l'intensité des forces ou la vitesse de la nacelle dépasse un seuil déterminé par les conditions de sécurité.
- ^(γ) Les processeurs SuperH sont fabriqués par la compagnie japonaise Hitachi (Renesas technology). Il s'agit de processeurs 32 bits basés sur une architecture RISC. Le processeur SuperH4 est un processeur super-scalaire (2-way) équipé d'une unité arithmétique au format flottant, d'une mémoire cache et d'une MMU (Memory Management Unit).

5 Systèmes commerciaux

Nous présentons maintenant les deux sociétés *Sensable* et *Force Dimension* et les systèmes qu'elles commercialisent. Nous avons choisi de présenter en particulier les produits de ces deux sociétés parce que nous les estimons représentatives des TGR qui sont commercialisés à l'heure actuelle.

5.1 Sensable et les bras haptiques *Phantom*

Le Phantom fut conçu au MIT au début des années 1990 par Thomas Massie et Kenneth Salisbury [Massie et Salisbury, 1994]. Il fut rapidement commercialisé par la société Sensable, créée en 1993.

Le Phantom a rapidement gagné en popularité grâce à une conception mécanique simple et robuste, et est devenu le système haptique le plus utilisé dans le monde. Aujourd'hui la gamme Sensable se compose de 3 types de systèmes :

- Le Phantom Omni est un système économique à 3 degrés de liberté.
- Le Phantom Desktop est un modèle intermédiaire pour une utilisation de table, de conception plus robuste que le Phantom Omni et présentant de meilleures capacités en termes d'espace de travail et de puissance des actionneurs.
- La série Phantom Premium comprend plusieurs systèmes de 3 à 6 degrés de liberté permettant un espace de travail allant des mouvements de main avec le poignet, jusqu'à un espace de travail permettant les mouvements du bras.

Les systèmes Phantom sont vendus avec le kit de développement OpenHaptics [SensableTechnologies, 2005]. Ce kit de développement est construit sur le modèle de la bibliothèque OpenGL¹⁰, permettant une prise en main rapide par les développeurs déjà familiers avec la programmation d'applications graphiques¹¹. Il se décompose en deux composants majeurs :

- HLAPI (Haptic Library API) est une bibliothèque de haut niveau qui permet de mettre en place des applications de simulation haptique rapidement, en mettant à la disposition du programmeur une bibliothèque « d'effets » et de formes géométriques. L'utilisation de cette bibliothèque dispense le programmeur de se préoccuper de la synchronisation et de la planification des différentes boucles de simulation : HLAPI assure que la boucle servo sera cadencée à 1 000 Hz. Cette boucle est exécutée dans un processus indépendant qui se voit attribuer une priorité importante pour lui garantir une exécution temps réel.
- HDAPI (Haptic Device API) est une bibliothèque de plus bas niveau sur laquelle repose HLAPI. Cette librairie permet de programmer une application communiquant directement avec le système haptique par l'intermédiaire de fonctions de bas niveau telles que `getPosition()` ou

¹⁰OpenGL est une spécification standard pour l'implantation de bibliothèques pour le développement d'applications graphiques 2D ou 3D. On trouve aujourd'hui des bibliothèques de développement OpenGL sur la majorité des systèmes d'exploitation.

¹¹La proximité de cette bibliothèque pour la simulation haptique avec OpenGL, si elle présente des avantages pratiques certains, est aussi révélatrice d'une manière de concevoir la simulation haptique fondée sur une approche géométrique de la modélisation et sur la représentation des formes statiques. Cette approche est issue de la communauté du *haptic rendering*.

`setForce()`. Cette bibliothèque donne également accès à des primitives de synchronisation et de gestion d'applications multi-processus.

Il est à noter que, si cette bibliothèque ne permet pas l'exécution du processus de simulation selon des contraintes temps réel (qui sont dépendantes pour la plus grande part du système d'exploitation), elle comprend tout de même des mécanismes permettant d'assurer que la charge globale du système permette une exécution satisfaisante du processus haptique (en termes de temps réservé à l'exécution d'une passe dans la boucle haptique). Ainsi, un mécanisme de chien de garde est implémenté, permettant d'interrompre la simulation lorsque le temps processeur alloué au calcul de la boucle de commande descend sous un certain seuil, signe que la charge du système est trop importante pour permettre l'exécution de l'application de simulation dans des conditions satisfaisantes.

5.2 Force Dimension – systèmes *Delta* et *Omega*

ForceDimension est une société suisse, créée en 2001 à la suite des travaux de développement du système haptique Delta, basé sur une mécanique en tripode [Grange *et al.*, 2001a, Force Dimension, 2008]. La société commercialise aujourd'hui deux gammes de systèmes haptiques :

- Les systèmes *Omega* sont des systèmes économiques procurant de 3 à 7 degrés de liberté.
- Les systèmes *Delta* sont des systèmes de 3 à 7 degrés de liberté de meilleures performances que les systèmes Omega (espace de travail plus grand, capacité en retour de force plus importantes).

Ces systèmes sont aujourd'hui livrés avec la bibliothèque de développement « DHD » [ForceDimension, 2007]. Il s'agit d'une bibliothèque multi-plateforme facilitant le développement d'applications pour la simulation haptique (gestion multi-processus) et la communication bas-niveau avec le TGR (accès direct aux valeurs des signaux de communication, tels que les registres des encodeurs ou les registres qui commandent les courants moteur).

Les systèmes de ForceDimension sont interfacés avec l'ordinateur exécutant l'application de simulation par le biais d'une électronique de contrôle. Cette électronique consiste en une carte mère qui s'interface à l'ordinateur via une connexion USB 2.0. Les amplificateurs utilisés sont de type PWM et les capteurs de positions sont des encodeurs optiques. Les deux librairies de développement de Sensable et de ForceDimension ont en commun les points suivants :

- La séparation entre le processus dit *haptique*, exécuté à 1 000 Hz environ, et le processus de simulation, parfois confondu avec le processus réservé à la gestion de l'affichage, exécuté à quelques dizaines de Hertz.
- Le processus haptique agit en maître sur l'électronique de contrôle du TGR. Les mises à jour de la consigne en force sur les actionneurs sont subordonnées à l'envoi d'une nouvelle consigne de force par le processus haptique, et de même, une nouvelle lecture des capteurs de position est déclenchée par une demande de lecture des positions par le processus haptique.

6 Approches synchrones

6.1 Introduction

La réactivité des systèmes est un des enjeux majeurs rendant la conception d'architectures de calcul synchrones difficile, parce que l'évolution actuelle des architectures de calcul privilégie les améliorations de performance en termes de puissance pure au détriment de la réactivité des systèmes [Patterson, 2004]. De plus, les systèmes d'exploitation actuels pour le grand public (Windows, OS X, Linux) ne sont pas conçus pour supporter les contraintes du temps réel : typiquement, la résolution temporelle de l'ordonnanceur de leur noyau est relativement importante (de l'ordre de la milliseconde).

L'approche asynchrone tire d'ailleurs parti de ce fait : ne pas synchroniser les différents processus intervenant dans l'application de simulation, c'est contourner les faibles performances des systèmes actuels en termes de réactivité et économiser les temps d'attente nécessaires à la synchronisation. La libération de cette contrainte permet d'utiliser les systèmes d'exploitation courants que nous venons de citer plutôt que des systèmes spécialisés.

Dans l'approche synchrone, tous les calculs, traités de manière indépendante, doivent être exécutés au cours d'une même période de temps. Il n'existe alors qu'une seule source horloge, sur laquelle doivent se synchroniser tous les processus impliqués dans la simulation. Deux grandes catégories de solutions existent.

La première consiste en l'utilisation d'un système d'exploitation spécialisé, dit *temps-réel*, permettant de garantir un certain déterminisme temporel quant à l'exécution des différents processus impliqués dans la simulation. Sur une architecture de calcul mono-processeur, le processus simulation, le processus superviseur et le processus système doivent se partager les ressources d'un seul processeur. Ceci est une contrainte forte puisque, si seul le processus simulation doit être exécuté en temps réel, le système d'exploitation doit tout de même veiller à allouer des ressources pour les processus superviseur et système. Dans le cas d'une architecture multi-processeurs, il devient possible d'allouer les ressources d'un ou plusieurs processeurs à l'usage exclusif du processus simulation, qui peut exploiter pleinement la puissance de calcul des processeurs qui lui sont alloués. Dans cette hypothèse, un processeur supplémentaire reste nécessaire pour le processus superviseur et le processus système, mais celui-ci n'est pas soumis à des contraintes d'exécution temps-réel.

La seconde solution consiste à programmer l'application de simulation synchrone à bas niveau, en utilisant directement les ressources matérielles de l'architecture et sans l'aide d'un système d'exploitation. Il s'agit d'une manière courante de procéder dans le domaine de l'embarqué, où les architectures sont relativement simples et/ou le système d'exploitation et ses mécanismes d'abstraction rudimentaires. Le développement de programmes est alors beaucoup plus proche du matériel que sur un système d'exploitation conventionnel, et permet un meilleur contrôle de l'utilisation matérielle de l'architecture de calcul, en particulier afin de garantir l'exécution temporelle de l'application (et donc de garantir des caractéristiques de synchronisme).

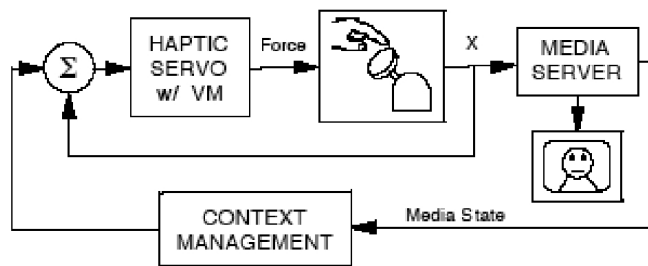


Figure 3.10. Architecture pour le contrôle haptique de média (MacLean et al.). L'architecture de simulation haptique est synchrone, mais une boucle lente (context management) peut modifier la boucle haptique 1 kHz – extrait de [MacLean et al., 1999]

6.2 Réalisations de simulateurs synchrones

Peu de travaux dans le domaine de l'haptique utilisent l'approche synchrone. Dans les travaux de MacLean, Snibbe et Shaw [MacLean *et al.*, 1999], la modalité haptique est envisagée comme un outil de contrôle ou de navigation sur des informations digitales de type images ou sons. La boucle haptique intègre le calcul du modèle mécanique simulé à la boucle de calcul synchronisée sur les entrées/sorties du TGR, calculée à 1 kHz. La boucle haptique est en relation, sur le modèle client/serveur, avec un serveur « media » par le biais d'une connexion série ou Ethernet. Le client média permet la navigation intelligente dans les données multimédia. Il s'agit d'un processus fonctionnant à 60 Hz, qui peut être synchronisé ou non (suivant les applications et les implantations) sur la boucle haptique. De plus, la boucle du serveur média peut modifier l'état du modèle mécanique simulé, afin de retourner à l'utilisateur des informations gestuelles. Ainsi la boucle haptique n'est pas seulement dépendante de l'exécution de la boucle de commande à 1 kHz, mais aussi des signaux de rétroaction de la boucle du serveur média. Comme dans le cas de l'approche asynchrone, ceci peut constituer une atteinte grave à la stabilité de la boucle haptique et à l'intégrité de la simulation mécanique en cours.

Hayward et al. ont développé le Pantograph à l'université de McGill, Montréal. Il s'agit d'un système haptique à deux degrés de liberté, procurant un espace de travail de 10×10 cm environ, conçu comme un outil de substitution à la souris d'ordinateur [Ramstein et Hayward, 1994]. Le simulateur accède aux capteurs et aux actionneurs du Pantograph par l'intermédiaire de cartes de conversion et de génération numérique-analogique, et la boucle de calcul est exécutée sur un ordinateur grand public équipé d'un système d'exploitation modifié pour le temps réel (Linux RT). La complexité des objets simulés étant relativement faible (par exemple, l'interaction avec les fenêtres et les boutons d'une interface graphique), la fréquence de simulation est de 10 kHz et permet d'exploiter les propriétés dynamiques du système.

Les premiers travaux sur la perception des propriétés rhéologiques de base ont été menés par Hong Tan à partir d'un système haptique à un seul degré de liberté conçu spécialement pour cet usage. Le « hand function tester » [Pang *et al.*, 1991] est un système équipé de deux plaques en aluminium destinées à être pincées par le sujet entre le pouce et l'index, actionnées par un moteur DC. La force exercée par le sujet est mesurée à l'aide d'une jauge de contrainte, la position de la partie mobile par un FLDT, et la vitesse par un capteur à induction. La boucle de contrôle est totalement

synchrone, exécutée par un DSP TMS320C20, et des convertisseurs A/N et N/A à 16 bits ferment la boucle de contrôle sur les signaux d'entrée du moteur DC et les signaux de sortie des capteurs. Dans cet exemple, l'utilisation d'une seule boucle synchrone avec les entrées/sorties des capteurs et des actionneurs est rendue possible par la petite taille des algorithmes de simulation (simulation d'une inertie simple, d'une visco-élasticité, etc.), et rendue nécessaire afin d'obtenir une mesure précise sur l'expérience.

6.3 Les simulateurs de l'ACROE

Les travaux de recherche de l'ACROE en matière de simulation multisensorielle en temps réel se situent dans le cadre de l'approche synchrone. Dans ce cadre, tous les processus de simulation sont synchronisés sur une référence temporelle unique, de même que la communication de l'état d'un modèle à destination des transducteurs (TGR, sortie sonore, etc.).

Les premières architectures de calcul utilisées à l'ACROE étaient basées sur un développement proche de la machine, permettant à la fois un contrôle rigoureux de l'exécution temporelle de la simulation et une utilisation efficace des ressources grâce à une optimisation poussée [Florens, 2008]. En 1978, un premier simulateur, basé sur un processeur LSI11 sur PDP11, puis en 1981, un processeur spécialisé CTR permettait la simulation synchrone à 25,6 kHz (entrées gestuelles non-synchrones). À partir de 1983, le processeur vectoriel AP120 fut utilisé dans plusieurs configurations : d'abord avec une machine PDP11 permettant le traitement du geste à 300 Hz, puis en utilisant une configuration à base de processeurs 68000 sur bus VME à partir de 1985, permettant une synchronisation complète de la simulation avec les entrées/sorties gestuelles jusqu'à 1 kHz.

À partir de 1993, les premières études ont été réalisées pour moderniser l'architecture de calcul. Le choix fut fait d'utiliser une architecture Power Challenge quadri-processeurs (flottants 64 bits cadencés à 75 MHz) de Silicon Graphics [Uhl, 1996]. Cette architecture, baptisée TELLURIS, était équipée d'un système d'exploitation complet de type Unix (IRIX) et adapté au temps réel. La problématique de conception des applications de simulation dut alors être envisagée sous un nouvel angle, puisque cette architecture matérielle n'était accessible que par la programmation à haut niveau (langage C) au travers de la couche d'abstraction du système d'exploitation IRIX (mécanismes d'indirection et de virtualisation). Sur cette architecture, le noyau de simulation implanté permettait une plus grande généricité ainsi que la répartition des processus de simulation sur les différents processeurs de l'architecture [Giraud, 1999]. Ce noyau fut ensuite porté sur une nouvelle architecture similaire mais plus puissante, dotée de six processeurs R8000, dont quatre cadencés à 90 MHz et deux à 75 MHz [Mendès, 2002]. Le noyau de simulation fut modifié pour supporter le calcul réparti sur plusieurs processeurs dédiés, ainsi qu'une version prototype d'architecture client/serveur. TELLURIS fut intensivement utilisé pour tous les travaux de recherche temps-réel jusqu'au début de l'année 2007. Cette plateforme de simulation était avant tout orientée vers le développement pour la recherche, et — hormis quelques rares déplacements de la plateforme pour des démonstrations — restait dans les murs du laboratoire en raison des difficultés de transport liées à la complexité de son architecture.

Son utilisation est progressivement remplacée par les résultats des réalisations de cette thèse puis les travaux menés dans la thèse de J. Castet sous la direction de J.L. Florens (en cours).

En matière de simulation synchrone, l'ACROE a d'abord connu une première phase basée sur un développement proche de la machine permettant une utilisation efficace des ressources de calcul mais parfois complexe et coûteuse. Plusieurs noyaux de simulations génériques et modulaires ont été développés sur ces architectures soit pour le temps différé, soit avec des performances pour le temps réel relativement faibles. La plupart des implantations étaient réalisées de manière ad hoc. Dans une deuxième phase, le simulateur TELLURIS, basé sur un système d'exploitation de type Unix, a permis la réalisation d'un véritable noyau de simulation CORDIS générique, modulaire et taillé pour le temps réel, permettant la simulation de modèles en deux fréquences : 3 kHz pour la partie geste, et 30 kHz pour la partie sonore. Cependant, la couche d'abstraction apportée par la programmation à haut niveau et par le système d'exploitation, apportait une perte d'efficacité en même temps qu'elle apportait un gain en modularité de conception [Uhl, 1996].

La réalisation du simulateur ERGON_X, où la mise en oeuvre du temps réel synchrone est supportée par une architecture de type DSP, nous place dans une position intermédiaire entre les deux situations que nous venons d'évoquer. Si les architectures DSP sont aujourd'hui de plus en plus complexes, elles restent néanmoins accessibles à un bas niveau et permettent une utilisation efficace des ressources de calcul, pour peu que l'on s'en donne les moyens. Les outils de développement modernes que nous avons à notre disposition permettent cependant une prise en main et un développement à haut niveau dans un premier temps (langage C/C++), qui, si elles ont un coût en termes d'efficacité, nous ont permis d'obtenir des résultats et de valider l'architecture rapidement. L'ouverture de cette architecture et la possibilité d'un développement à bas niveau permettront toujours dans un deuxième temps d'améliorer l'efficacité de notre simulateur selon les besoins.

6.4 TELLURIS

Nous présentons maintenant l'architecture du simulateur TELLURIS. Au début des travaux de cette thèse, TELLURIS était le référentiel du groupe de recherche ACROE & ICA en matière de simulation interactive multisensorielle. La réalisation de ERGON_X en a repris les grands principes architecturaux.

6.4.1 Architecture de la plateforme

Le cœur de l'architecture TELLURIS est le calculateur Power Challenge de Silicon Graphics. Cette machine comporte 6 processeurs R8000, dont quatre cadencés à 90 MHz, et deux cadencés à 75 MHz. Tous les processeurs travaillent au format flottant 64 bits, et comportent chacun 4 unités arithmétiques, ce qui leur confère respectivement une puissance de 360 et de 300 Mflops par processeur.

La machine M6 est un calculateur temps réel supplémentaire. Elle est équipée de deux processeurs de la famille des Motorola 68000, et sa fonction dans l'architecture de simulation est d'assurer la communication des données entre le processus simulation sur le calculateur Power Challenge, et le TGR ERGOS. Un dispositif de mémoire miroir connecte les deux bus VME de la machine Power Challenge, et de la machine M6, permettant ainsi l'échange des données entre le processus simulation et le TGR. La machine M6 contrôle les convertisseurs A/N et N/A reliés au TGR.

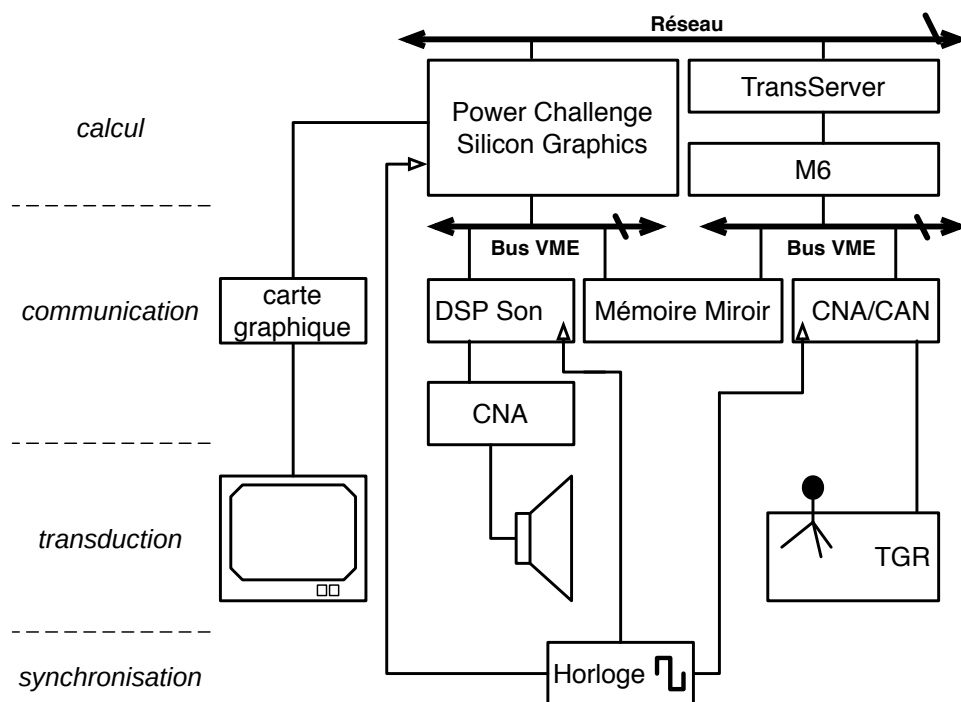


Figure 3.11. Architecture de la plateforme de simulation temps-réel TELLURIS

Le contrôle des données son en provenance du simulateur est assuré par une carte DSP dédiée, équipée d'un processeur de la famille des C40 (Texas Instruments). Le DSP achemine les données du simulateur vers les convertisseurs spécifiques aux sorties son. La carte DSP a pour autre fonction de réaliser l'interpolation entre le signal de données émis par le simulateur, de fréquence variable (typiquement 30 kHz) et le signal à destination des convertisseurs audio, au format AES/EBU (fréquence d'échantillonnage 48 kHz).

Enfin, la gestion des processus de visualisation s'effectue sur la machine Power Challenge qui dispose de sa propre carte graphique. Le processus de visualisation est exécuté sur les ressources du processeur système.

La synchronisation des processus de visualisation sur les processus de simulation est encore à ce jour un problème épineux pour les architectures de simulation en temps réel. En effet, les processus de visualisation sont le plus souvent des processus lourds, dont il est difficile de garantir l'exécution en temps réel dur. De plus, pour ces processus, l'exécution synchrone avec une référence temporelle externe (typiquement donnée par le processus de simulation) est délicate parce que la gestion de l'affichage est masquée par plusieurs couches logicielles (pilote de la carte graphique, processus exécuté sur la carte graphique, etc.) et matérielles (carte graphique, connectique vers l'écran, etc.).

« Nous estimons de toute façon que quelle que soit la complexité choisie pour la représentation graphique des simulations en temps réel, nous aurons des problèmes d'instabilités et de temps de calculs importants en regards des périodes des cycles de simulation.

Ainsi [...], nous devons prévoir un processus de production des images pour la sortie visuelle qui n'impose pas son propre rythme (instable) à la simulation synchronisée. »

Olivier Giraud, 1999 [Giraud, 1999]

De fait, le processus de visualisation implanté dans TELLURIS entretient une relation asynchrone avec le processus de simulation, utilisant le principe de moindre latence : le processus de visualisation utilise les données de simulation immédiatement disponibles au moment où celui-ci se trouve prêt à lancer un nouvel affichage. Ainsi, si nous ne pouvons assurer la synchronisation des données affichées avec les données de simulation, nous pouvons tout de même garantir que les données de visualisation seront affichées avec un délai inférieur à la période d'exécution du processus de visualisation (sachant que celui-ci travaille à une fréquence qui peut varier dans une certaine mesure).

Une horloge globale permet la synchronisation de tous les processus. Elle est fournie par le module son. Le signal d'horloge original fonctionne à 48 kHz, et un diviseur permet d'obtenir la fréquence geste (3 kHz).

L'horloge de synchronisation est distribuée sur la carte de conversion du module geste, où elle déclenche les conversions en entrée et en sortie.

6.4.2 Le compromis entre généralité et efficacité

Le moteur de simulation TELLURIS existe en deux versions :

- Une version générique, permettant la conception de modèle directement à partir d'une interface texte.
- Une version *ad hoc*, pour laquelle la conception d'un modèle physique demande de reprogrammer le moteur de simulation en fonction de la topologie du modèle et des types de modules à simuler. Dans ce cas, il existe en fait autant de versions du moteur de simulation que de modèles *ad hoc*.

Dans le cadre de la version générique de TELLURIS, la conception des modèles de simulation s'effectue par l'intermédiaire de fichiers texte (format .ct). Le concepteur du modèle peut directement concevoir son modèle en déclarant toutes les instances de modules CORDIS qui seront utilisées, et en déclarant les valeurs de chacun de leurs paramètres. Ces fichiers de paramètres constituant les modèles sont transmis au moteur de simulation de TELLURIS qui instancie leurs composants en allouant l'espace mémoire dédié à chaque module. La boucle de simulation du moteur TELLURIS est adaptée au nombre et au type de modules instanciés dans le fichier .ct. Il suffit d'être familier avec le formalisme de modélisation CORDIS pour pouvoir concevoir des modèles physiques pour le moteur générique de TELLURIS.

La conception de modèles *ad hoc* consiste à modifier directement la structure du moteur de simulation en fonction de la topologie du modèle et des types de modules à simuler. Plus

précisément, la conception d'un modèle *ad hoc* nécessite de reprogrammer les composants suivants du programme TELLURIS à partir d'un squelette du programme de simulation :

- La boucle de simulation : sélection des algorithmes à simuler parmi une bibliothèque de modules déjà disponibles, voire programmation de nouveaux modules CORDIS.
- Les allocations nécessaires pour les variables de simulation (allocation de l'espace mémoire pour chaque module, etc.).
- Adaptation du squelette de simulation en fonction des besoins (par exemple : utilisation de simulation multi-fréquence).
- Adaptation du processus d'affichage graphique.
- Gestion des entrées-sorties (transducteur gestuel et son).

Nous le voyons, l'implémentation d'un modèle *ad hoc* présente des inconvénients certains par rapport à l'utilisation du moteur générique de TELLURIS :

- Le concepteur des modèles doit posséder une bonne connaissance de la programmation, mais aussi de l'architecture matérielle de TELLURIS.
- Le défaut d'une interface de modélisation adaptée rend la conception du modèle lui-même plus longue (choix de la topologie CORDIS adéquate, ajustement des jeux de paramètres) ; la modification du programme de simulation est source d'erreurs parfois difficiles à détecter.
- De tels modèles sont pas génériques par définition ; ainsi les modifications ou ajustements de tels modèles *a posteriori* s'avèrent plus difficiles.

Néanmoins, si la conception de modèles *ad hoc* est toujours pratiquée alors que nous avons à notre disposition un moteur de simulation générique, c'est qu'elle comporte quelques avantages :

- La rapidité d'exécution du code, puisque optimisé pour un modèle précis.
- La possibilité d'instancier de nouveaux modules CORDIS spécifiques, n'existant pas dans la bibliothèque du moteur générique.
- La possibilité d'accéder à toutes les fonctions et toutes les mémoires tampon du processus temps-réel, en particulier la possibilité de traiter les entrées-sorties de manière non conventionnelle.
- L'ajout de fonctionnalités pour un usage particulier (par exemple, l'enregistrement de traces de gestes).

7 Conclusion

Nous avons présenté dans ce chapitre les différentes approches utilisées dans le domaine de la recherche haptique pour la connexion entre le TGR et les entrées/sorties du processus de simulation, ainsi que les relations entre les composants logiciels de l'application de simulation. Il ne s'agit pas d'une présentation exhaustive des différents systèmes, mais plutôt d'une présentation des principales approches en prenant pour exemples des systèmes caractéristiques des travaux menés actuellement.

Nous pouvons y distinguer deux grandes écoles. La première est issue du champ de la robotique. Elle considère les traitements locaux aux entrées/sorties du TGR, et les critères de fonctionnement de la boucle de commande. Elle vise une connaissance approfondie des problèmes à la frontière entre les mondes analogique et numérique et à leurs effets sur les propriétés physico-mécaniques du TGR. La seconde école est issue du champ de l'infographie. Elle s'intéresse aux techniques

numériques de simulation en temps réel d'environnements ou d'objets physiques, et cherche avant tout à accroître la complexité des objets simulés et à rendre les environnements simulés plus crédibles. La relation avec le TGR, sa causalité et ses effets ne sont que peu questionnés dans cette approche : le TGR est considéré comme une boîte noire dont on suppose le fonctionnement (parfaitement) maîtrisé. Le domaine d'étude de cette école commence là où le domaine d'étude du champ de la robotique s'arrête. La frontière entre ces deux écoles se situe au niveau des ports de communication des processus de traitements locaux aux entrées/sorties du TGR.

Dans ces deux approches, on retrouve deux principes fortement ancrés. Le premier est que les traitements locaux au TGR (conditionnement des signaux, pré-filtrage...) sont isolés du processus de simulation. Ainsi, on retrouve systématiquement une boucle de traitement locale aux entrées/sorties du TGR (la boucle de commande), dont le but est de mettre en forme les signaux à destination du processus de simulation. Ces traitements sont également le lieu de boucles de régulation des signaux de consigne en provenance du processus de simulation. Ces traitements étant indépendants du processus de simulation, ils sont le plus souvent exportés sur une architecture de calcul dédiée qui peut être intégrée à l'électronique de conditionnement ou de puissance du TGR. Ils sont exécutés sur un système temps-réel afin de garantir leur synchronisation avec les signaux d'entrée/sortie du transducteur gestuel. Toutes ces fonctionnalités sont regroupées dans le composant que nous avons appelé le *contrôleur haptique*.

Le second principe est que dans la plus grande partie des implantations présentées dans la littérature, la boucle de commande implantée dans le contrôleur haptique et la boucle de simulation sont asynchrones. Cette approche relève d'un pragmatisme simple : synchroniser deux processus de calculs est coûteux en temps. Dans le cas où la boucle du contrôleur haptique et la boucle de simulation utilisent les ressources d'un même processeur¹², l'approche asynchrone repose sur l'hypothèse que l'ordonnanceur du système sait répartir efficacement les ressources de calcul (ce qui est heureusement le plus souvent le cas), puisqu'aucun temps d'attente n'est à réserver dans le processus de simulation. L'approche asynchrone se montre pertinente dans le cas de la simulation d'objets rigides statiques ou peu mobiles. Dans ce cas, la surface de l'objet mobile qui se trouve palpé par l'utilisateur par le biais du TGR est approximée à une surface statique, dont les coordonnées spatiales sont mises à jour par le processus de simulation. Observons enfin que la synchronisation des différentes composantes logicielles utilisées dans une architecture de simulation pour l'haptique, ou son absence, est une caractéristique extrêmement importante d'un simulateur en temps réel. Cette caractéristique est pourtant passée sous silence le plus souvent. La culture de la simulation en temps réel multisensorielle questionne peu les questions de synchronisation au sein d'une architecture de simulation. Ceci constitue néanmoins, à notre avis, une lacune grave de la culture scientifique dans le champ de l'haptique aujourd'hui. Ce point sera discuté à nouveau dans le chapitre 6, dans le cadre de l'utilisation de la technologie du retour d'effort pour l'expérimentation psychophysique.

Alors qu'il n'est généralement fait que de peu de cas des techniques de synchronisation dans une architecture pour la simulation multisensorielle, la littérature haptique abonde d'informations relatives d'une part à la conception mécanique des systèmes, l'approche théorique du contrôle (domaine hérité de la robotique), d'autre part aux techniques de simulation d'environnements réel complexes, de calculs d'interaction entre objets rigides, de calculs de déformations d'objets

¹²Ce qui est souvent le cas lorsqu'une application haptique est exécutée sur un système non spécialisé.

mous. Mais la jonction entre l'approche issue de l'infographie et l'approche issue de la robotique n'est pas encore faite. La littérature relevant de la robotique ne fait référence que d'une manière allusive à l'implémentation de ces algorithmes dans une machine de calcul et leur synchronisation aux autres composants de la simulation. La littérature relevant des réalités virtuelles ou de l'infographie considère le plus souvent le système haptique comme une boîte noire (contrôleur haptique y compris) dont on suppose qu'elle répondra de manière idéale (causalité et linéarité) à une commande donnée.

Cependant, dans la plupart des architectures que nous avons présentées dans ce chapitre, la problématique du couplage est n'est jamais explicitement posée : d'une part l'approche héritée de la robotique s'attache aux conditions de stabilité de la boucle locale au TGR, d'autre part l'approche héritée de l'infographie s'attache à la technique de simulation d'une scène virtuelle mais en traitant le TGR comme une boîte noire. Seule l'approche synchrone, qui, nous l'avons vu, est peu représentée dans la littérature, s'intéresse à la question du couplage entre le TGR et le processus de simulation, puisqu'elle ne fait pas l'hypothèse que le fonctionnement du processus de simulation d'un côté et du TGR de l'autre peuvent être traités indépendamment. Au contraire, dans les architectures synchrones, la question du couplage entre la main de l'homme et l'objet simulé est centrale puisqu'elle prend le parti de traiter le processus de simulation de l'objet virtuel et la communication avec le TGR au sein d'une même boucle. C'est le cadre conceptuel qui est choisi pour les travaux de simulation en temps réel effectués à l'ACROE : l'existence d'un couplage structurel fort entre tous les éléments de modélisation constitutifs d'un même objet simulé est reconnue.

Ce choix nécessite :

1. de disposer d'un formalisme de modélisation qui intègre la multisensorialité ; dans le cadre de cette thèse, nous utilisons le formalisme CORDIS-ANIMA (cf. annexe D).
2. de pouvoir garantir l'exécution en temps réel d'un modèle autonome basé sur ce formalisme ; l'annexe D détaille comment ce point est garanti par le formalisme CORDIS-ANIMA
3. d'assurer que les communications entre les transducteurs de la plate-forme de simulation et le processus de simulation présenteront des caractéristiques temporelles adéquates ; en d'autres termes, que le temps de communication entre l'intérieur et l'extérieur du simulateur ne mettront pas en péril le respect des contraintes d'exécution en temps réel et de synchronie de l'architecture de simulation.

Le HPU, que nous présentons dans le chapitre suivant, est en partie né de la nécessité présentée dans le dernier point ci-dessus : disposer d'un composant matériel à même de garantir les conditions d'exécution synchrone de notre processus de simulation, et ce jusqu'à l'interface avec les transducteurs de la plate-forme de simulation (des afficheurs, des sorties sonores, des transducteurs gestuels rétroactifs). Nous présentons dans le chapitre suivant les solutions envisageables pour sa mise en oeuvre.

Chapitre 4

Le HPU — *Haptic Processor Unit*

Sommaire du Chapitre

1	Introduction	83
2	Composants fonctionnels	84
3	Définition du HPU	87
3.1	Le choix du terme HPU	87
3.2	Rôle du HPU	88
3.3	Place du processus simulation	89
3.3.1	Configuration 1 : le HPU héberge le processus simulation	89
3.3.2	Configuration 2 : le HPU n'héberge pas le processus simulation	89
3.3.3	Configuration mixte : combinaison des configurations 1 et 2	90
3.4	Interface « geste » du HPU	90
3.4.1	Latence	91
3.4.2	Synchronisation sur une horloge	91
3.4.3	Nombre de canaux de communication	91
3.4.4	Traitements pré- et post-simulation	92
3.4.5	Interface « geste » analogique	93
3.4.6	Interface « geste » numérique	95
3.5	Interface « simulateur » du HPU	97
3.5.1	Le temps de transfert n'est pas inversement égal au débit maximal d'un bus numérique	97
3.5.2	Interface « simulateur » connectée au processus superviseur	97
3.5.3	Bus de données pour l'interface simulateur	98
3.6	Le processeur du HPU	100
4	Conclusion	101

1 Introduction

Le chapitre 3 présentait les grands principes des architectures de simulation pour la réalité virtuelle. Il insistait sur la rupture technologique entre, d'une part, les approches héritées de la robotique qui étudient la boucle de simulation locale à l'interface de communication avec le TGR,

et d'autre part, les approches héritées de l'infographie qui s'intéressent aux architectures pour l'exécution du processus de simulation. Nous y avons montré qu'il n'existe actuellement que très peu d'architectures qui traitent la question de la relation entre l'exécution du processus de simulation et la communication avec le TGR. La mise en évidence de ce manque n'a d'ailleurs pas été une question triviale. Plus exactement, nous avons montré que seules les approches que nous avons qualifiées de synchrones s'intéressent par principe à cette question. Ces approches sont minoritaires dans les champs disciplinaires concernés par nos objectifs : robotique, réalités virtuelles, synthèse de sons et d'images.

Nous développons dans ce chapitre le concept de HPU, acronyme de *Haptic Processor Unit*, destiné au minimum à garantir le synchronisme entre les entrées / sorties du TGR et le processus de simulation. De ce fait, il s'agit d'un composant qui, à l'image du GPU pour la connexion à une sortie graphique, ou la carte son pour la connexion à une sortie audio, se doit de jouer le rôle d'interface entre le TGR et le reste de l'architecture de simulation.

Ce composant aura pour fonction première un rôle de communication entre le TGR et le reste de l'architecture de simulation, mais pourra aussi endosser un rôle de simulation, comme cela s'est vu dans l'évolution des fonctions supportées par les cartes GPU ou les cartes son.

Nous avons vu dans le chapitre précédent que, dans les architectures actuelles de réalités virtuelles, les fonctions endossées par le HPU sont présentes, mais mal identifiées, et par conséquent dispersées plus ou moins explicitement dans tout le système. Nous avons expliqué cela par le fait que, face à la diversité des architectures de simulation et des technologies de transduction, la conception d'un tel composant qui puisse répondre de manière générique et intégrée à ces deux fonctions soit délicate.

C'est la tâche à laquelle nous nous sommes attelés, par la conception et la réalisation d'un composant unique, le HPU, de manière à garantir de manière explicite, bien localisée et bien maîtrisée, les processus de synchronisation et de multisensorialité. Ainsi, ce sera uniquement au siège de ce composant que jouera la mise en relation entre le TGR et l'algorithme de simulation de l'objet virtuel. Le HPU sera alors de fait le garant de *l'instrumentalité* de la plate-forme de simulation.

Ce chapitre développe ce concept de HPU, ses fonctionnalités et les solutions technologiques susceptibles de répondre à ces objectifs.

2 Composants fonctionnels

Considérons d'un point de vue général les composants d'une plateforme pour la simulation multisensorielle (figure 4.1). Le simulateur se compose des unités de calcul en charge du traitement de la simulation, de périphériques (les transducteurs) ainsi que des composants matériels permettant la connexion aux périphériques. Les transducteurs transforment (traduisent) les signaux en provenance du simulateur en phénomènes sensibles : par exemple, un haut-parleur traduit un signal en vibration mécanique, un écran traduit un signal en énergie lumineuse. Entre le simulateur et les transducteurs se trouvent un ou plusieurs composants matériels dont le rôle est d'assurer la communication des informations et leur éventuelle traduction dans un format

de communication adapté. En particulier, chaque transducteur (haut-parleur, écran...) est adapté à une modalité perceptive humaine particulière, mais nécessite l'utilisation de périphériques de connexion adaptés. Ainsi, écrans et haut-parleurs d'un ordinateur sont connectés au processeur par le biais de cartes spécialisées : la carte graphique pour l'écran ou les sorties vidéo, la carte son pour les sorties audio.

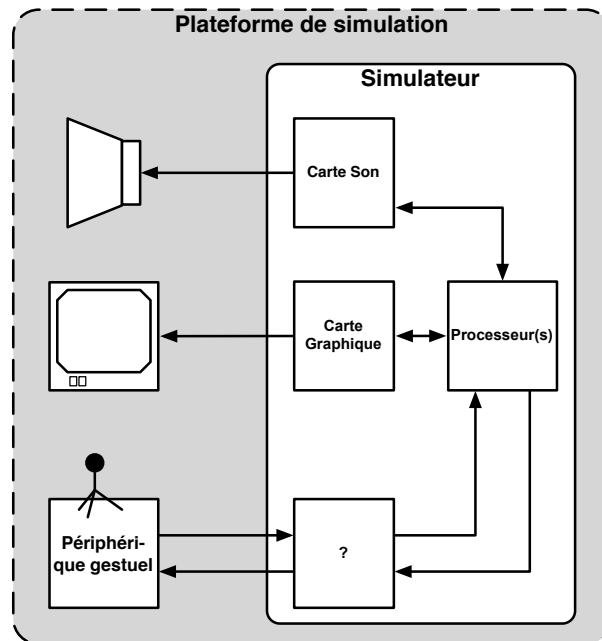


Figure 4.1. Schéma général d'une plateforme de simulation multisensorielle : chaque transducteur est connecté au processeur principal par le biais de périphériques spécialisés.

Dans le domaine du son et de l'image, il existe depuis longtemps des standards de communication à la fois pour le format des données échangées¹, mais aussi pour l'interface matérielle elle-même². Dans le domaine des interfaces gestuelles, plusieurs propositions de formats de codage du signal gestuel ont récemment vu le jour [Luciani *et al.*, 2006a, Luciani *et al.*, 2006b, Jensenius *et al.*, 2007], mais à l'heure actuelle aucune de ces propositions n'a encore été communément adoptée, parce que chaque usage donne lieu à des implantations matérielles et logicielles spécifiques. Les seules interfaces gestuelles dont les formats de données et les périphériques matériels corres-

¹Certains formats de fichiers propriétaires ou libres tels que *AVI*, *WAV*, *MOV*, *MP3*, *OGG*, etc. sont actuellement pratiqués comme des standards, et sont reconnus par la plus grande majorité des logiciels pour le multimédia ; on peut noter également l'existence du *Moving Pictures Expert Group*, comité de travail de l'*ISO* et de la *CEI*, chargé de travailler sur les normes pour le traitement et le codage des signaux audio et vidéo. Ce comité a, entre autres, défini les formats *MPEG-2* ou *MPEG-4*.

²Le port AGP (Accelerated Graphics Port) était il y a encore peu de temps le port de communication standard pour les cartes graphiques. Les cartes son ont été longtemps supportées sur port ISA. Le port PCI, successeur du port ISA, tend à devenir la norme, à la fois pour les cartes graphiques et pour les cartes son. AGP, ISA et PCI définissent des standards pour les ports internes aux ordinateurs, permettant de relier la carte mère à différents périphériques. Les ports externes des ordinateurs sont également standardisés, et les ports les plus couramment utilisés aujourd'hui sont USB et IEEE 1394, encore appelé FireWire. Tous ces ports de communications correspondent à des standards précisément définis, décrivant entre autres, le protocole de communication en termes de traitement du signal, c'est-à-dire la manière d'encoder les informations, mais aussi les caractéristiques physiques des ports de communication telles que les dimensions du support mécanique, les niveaux de tension sur chacune des broches, etc.

pondent à des standards sont le clavier informatique, la souris ou les périphériques pour les jeux vidéo comme les joysticks.

De la même manière, dans le domaine des architectures pour l'haptique, de nombreuses architectures de simulation ont été développées, mais il n'existe pas à l'heure actuelle de standards clairement définis. En effet, seuls quelques principes émergent des pratiques, tel l'utilisation d'un contrôleur, pratique issue du champ de la robotique (chapitre 3). À notre connaissance, le HPU est la première proposition de composant destiné à une architecture de simulation multisensorielle spécifiquement pour la simulation haptique.

La figure 4.2 présente les différents composants que nous trouvons dans un simulateur, à partir d'un point de vue fonctionnel sur l'organisation de la chaîne de simulation pour le geste. Sans présager encore des fonctions supportées par le HPU, nous voyons apparaître plusieurs fonctions :

- La fonction de *transduction*, assurée par l'écran, le haut-parleur ou le TGR, qui permet de transformer une information, codée dans un signal, sous la forme d'un phénomène physique sensible par un humain (un flux lumineux, une onde acoustique).
- La fonction de *communication* entre les unités de transduction et les unités de calcul. Il ne s'agit pas seulement de la transmission d'informations d'un point à un autre de l'architecture de simulation, mais aussi des traitements qui peuvent être rendus nécessaire pour une communication effective. Il peut s'agir par exemple de la mise en correspondance entre (1) du format de données supporté par le processeur de la machine hôte et (2) le format des signaux d'entrée et/ou de sortie du transducteur.
- La fonction de *traitement de données (processing)*. Dans le cas du geste, cette fonction comprend entre autres la simulation d'un modèle physique. Dans les architectures informatiques modernes, la communication des données du processeur central aux périphériques est accompagnée d'un certain nombre de traitements, assurés par un composant spécialisé hébergé dans la carte graphique ou la carte son. Ces composants embarquent beaucoup de fonctionnalités plus avancées que la simple traduction d'un format de signal à un autre : les cartes graphiques permettent le calcul d'objets 3D à partir de primitives simples envoyées par le processeur central, les cartes graphiques possèdent des fonction de filtrage, de spatialisation, de synthèse, etc. Ces calculs sont embarqués dans un composant spécifique non seulement parce qu'ils permettent une réduction de la charge des unités de calcul, mais aussi parce qu'ils demandent le plus souvent une architecture de calcul *spécifique* pour être effectués efficacement. Notons aussi que, dans les cartes graphiques ou dans les cartes son, ces fonctionnalités sont maintenant standardisées : par exemple, toutes les cartes graphiques permettent l'utilisation de OpenGL³ ou DirectX⁴.

³OpenGL (Open Graphics Library) est une spécification standard qui définit une API pour l'écriture d'applications graphiques 2D ou 3D. Les fabricants de matériel graphique fournissent des bibliothèques de programmation suivant les spécifications OpenGL pour la plupart des systèmes actuels.

⁴Microsoft DirectX est un jeu d'APIs pour la programmation d'applications liées au multimedia (graphisme, son, etc.).

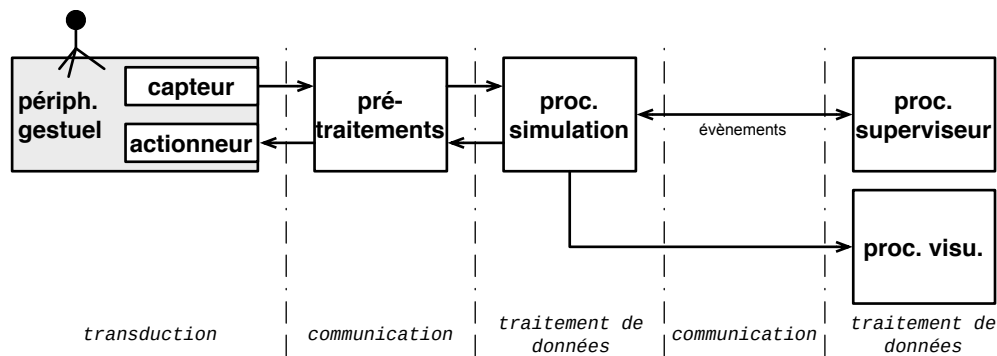


Figure 4.2. Composants fonctionnels de la boucle de simulation haptique

3 Définition du HPU

3.1 Le choix du terme HPU

Dans les architectures informatiques actuelles, la connexion de l'unité de calcul centrale (qui peut être composée de un ou plusieurs processeurs) aux périphériques est prise en charge par des composants électroniques spécialisés : une *carte son* permet la connexion avec les entrées et les sorties sonores, une *carte graphique* permet la connexion avec les dispositifs d'affichage et les entrées vidéo éventuelles. L'utilisation de composants intermédiaires entre les processeurs centraux et les périphériques permet de soulager les processeurs centraux des calculs spécifiques aux traitements des signaux de sortie, et permet d'augmenter la puissance de calcul disponible pour ces traitements par l'utilisation d'architectures de calculs spécialisées. Citons par exemple les cartes graphiques, qui sont maintenant toutes équipées d'une unité de calcul spécialisée dans le traitement des signaux à envoyer aux périphériques d'affichage, le *GPU* (*Graphics Processing Unit*) : un afficheur étant traditionnellement constitué d'une matrice de points lumineux, le traitement des signaux d'affichage consiste à effectuer du calcul sur des matrices de grandes dimensions. Les GPU sont spécialisés dans le calcul matriciel, et leur architecture est optimisée pour effectuer efficacement les traitements sur des données vidéo. Le processeur central communique alors avec la carte graphique par des instructions de haut niveau du type « *afficher une sphère* », « *afficher un cube* », que le processeur graphique sait traduire en signaux utilisables par l'afficheur.

Si les premières cartes graphiques étaient relativement rudimentaires et n'offraient guère plus de fonctionnalités que la connexion entre le processeur central et le dispositif d'affichage, les cartes graphiques actuelles embarquent un GPU dont les fonctionnalités permettent le traitement de scènes graphiques 3D complexes, mais aussi le traitement photo ou vidéo en temps réel⁵. Dans les cartes graphiques actuelles, les fonctionnalités de calcul (*processing*) prennent le pas sur les fonctionnalités de connectivité, en partie parce que ces dernières font l'objet depuis longtemps de standards sur lesquels se calque toute l'industrie graphique. Ainsi, le choix d'une carte graphique se fait aujourd'hui essentiellement en fonction de son GPU, et par synecdoque, on parle aujourd'hui souvent de GPU pour désigner en fait le composant « carte graphique » dans son intégralité.

⁵http://www.nvidia.com/content/nsist/module/what_gpu.asp?partner=

De la même manière, plutôt que de parler de « carte haptique » [Couroussé, 2007a], nous utiliserons le terme *HPU* (*Haptic Processor Unit*) pour désigner le composant se trouvant entre le TGR et le reste de l'architecture de simulation, c'est-à-dire l'*unité* spécialisée dans le *traitement haptique* au sens large.

3.2 Rôle du HPU

Par analogie avec le GPU, nous pouvons alors définir le HPU comme le composant dont le rôle premier est de connecter une architecture informatique — notre simulateur — à un Transducteur Gestuel Rétroactif dont l'unique fonction est, comme son nom l'indique, la transduction. La première fonction du HPU est donc une fonction de *communication* entre le TGR et le simulateur (figure 4.2). Le HPU doit donc posséder deux interfaces que nous décrirons plus en détails dans la suite de ce chapitre : une *interface* « *geste* » pour la connexion du HPU au TGR, et une *interface* « *simulateur* » pour la connexion du HPU au simulateur et à son ou ses processeurs principaux (figure 4.3).

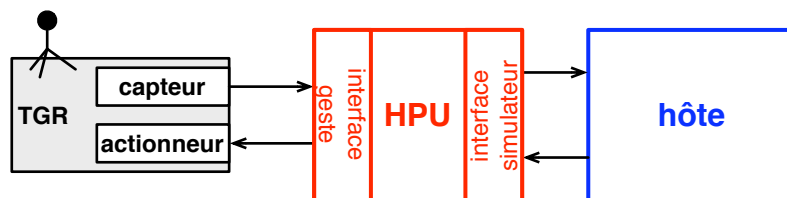


Figure 4.3. Les interfaces « *geste* » et « *simulateur* » du HPU

Définissons maintenant plus précisément les fonctions supportées par le HPU, ainsi que ses frontières. En définissant le HPU comme le composant qui permet de connecter le simulateur au TGR, l'interface « *geste* » du HPU est nécessairement en relation avec le TGR et notre définition de l'interface « *geste* » devra prendre en compte les particularités du fonctionnement des TGR (section 3.4 de ce chapitre). La définition de l'interface « *simulateur* » du HPU est délicate parce qu'elle dépend de la position de la frontière entre le HPU et le reste de l'architecture du simulateur. La seule certitude que nous pouvons avoir est que le processus superviseur, chargé du contrôle et la gestion de l'application de simulation, ne sera pas exécuté sur le HPU. Nous pouvons en effet supposer que le processus superviseur supporte entre autres une interface grâce à laquelle un utilisateur peut contrôler l'exécution du programme de simulation par l'intermédiaire du clavier, d'une souris et d'un affichage graphique adapté. La gestion d'une telle interface relève du domaine des HCI et fait repose sur l'utilisation d'un ordinateur standard essentiellement. L'interface « *simulateur* » du HPU se trouve donc nécessairement avant le processus superviseur, mais a contrario, il est possible d'exécuter le processus simulation sur le HPU ou sur le processeur du simulateur. Nous allons détailler ces différentes options dans la section suivante.

Le HPU n'est pas un simulateur autonome. Il s'agit seulement d'un composant que nous pouvons intégrer à une architecture pour la simulation multisensorielle et dédié à la connexion de ce simulateur à un TGR. Dans la suite de ce chapitre, nous désignerons l'architecture de simulation par le terme *simulateur hôte*, en référence au fait que le HPU est un composant *hébergé* sur une autre architecture informatique (figure 4.3).

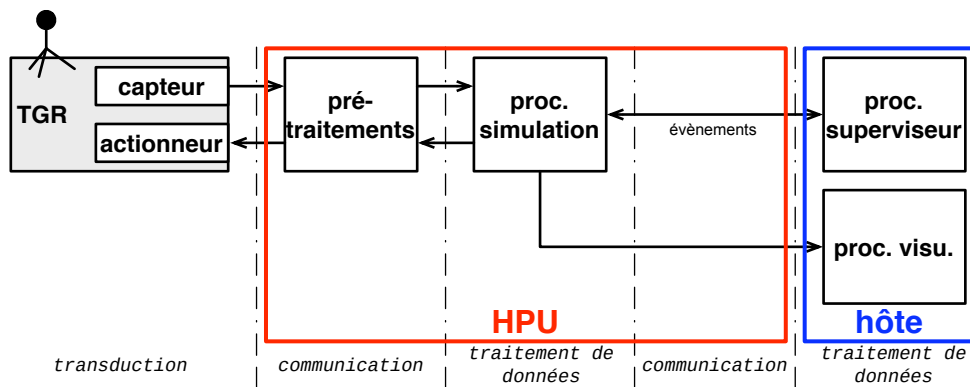


Figure 4.4. Configuration 1 : le HPU héberge le processus simulation

3.3 Place du processus simulation

Nous devons alors considérer deux configurations élémentaires : soit le processus simulation est intégré à le HPU, soit le processus simulation est externe au HPU.

3.3.1 Configuration 1 : le HPU héberge le processus simulation

Dans la première configuration, l'objet simulé reste confiné au HPU, et dans ce cas le HPU doit supporter intégralement le processus simulation, qui calcule l'objet physique simulé (figure 4.4). Le processus simulation doit pouvoir communiquer avec le processus superviseur par l'interface simulateur du HPU. Les signaux qui transitent sur l'interface simulateur sont alors de nature événementielle [Couroussé, 2002] : le processus superviseur communique avec le processus simulation par le biais de commandes — par exemple, démarrer ou arrêter la simulation —, ou de flux de données asynchrones dont le débit n'est pas critique — par exemple, charger un nouveau modèle, envoyer un nouveau jeu de paramètres algorithmiques. Dans cette configuration, l'interface simulateur du HPU doit pouvoir supporter seulement un mode de communication asynchrone à faible bande passante, et le temps de latence de communications n'est pas critique pour la réalisation de la simulation.

3.3.2 Configuration 2 : le HPU n'héberge pas le processus simulation

Dans cette configuration, le calcul de l'objet ou de l'environnement simulé s'effectue en dehors du HPU (figure 4.5). Cela revient à dire que le HPU joue le rôle d'un canal de communication entre le processus simulation et les entrées/sorties du TGR. Dans ce cas, l'interface simulateur du HPU est soumise aux mêmes contraintes que l'interface geste : le temps de transfert des données doit s'effectuer avec un délai minimal, et le déclenchement des envois de données doit pouvoir être asservi sur une horloge externe. Les temps de transferts bidirectionnels des données entre l'interface geste et l'interface simulateur du HPU deviennent aussi une caractéristique critique pour la réalisation d'une simulation dans les conditions adéquates.

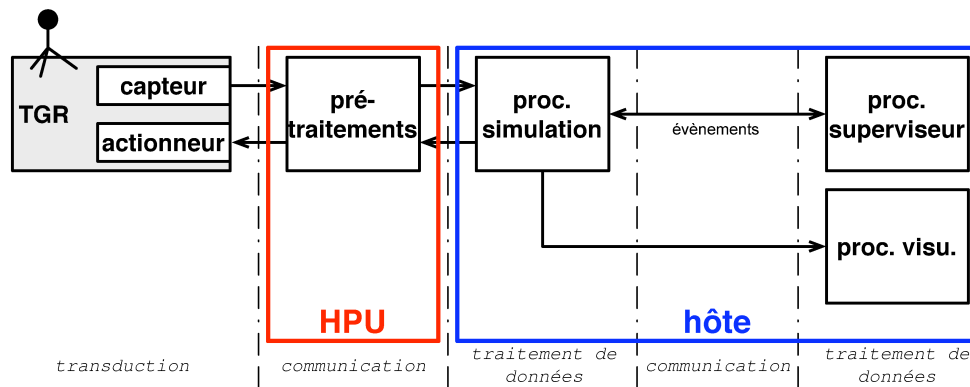


Figure 4.5. Configuration 2 : le HPU n'héberge pas le processus simulation

3.3.3 Configuration mixte : combinaison des configurations 1 et 2

Nous pouvons imaginer une configuration composée à partir des deux configurations élémentaires précédentes : la simulation de l'objet ou de l'environnement haptique est partagée sur plusieurs processus ou sur plusieurs unités de calcul (figure 4.6). Dans le cas des approches asynchrones dites de *haptic rendering*, une boucle de simulation rapide (la boucle de commande), proche des entrées sorties du TGR, calcule le modèle mécanique de l'environnement proche du point manipulé, et une boucle de simulation plus lente, hébergée par une autre unité de calcul, prend en charge la simulation du reste du modèle. Dans ce cas, l'interface simulateur du HPU doit présenter les mêmes caractéristiques que dans la configuration 2 où le processus simulation est hébergé sur le simulateur hôte.

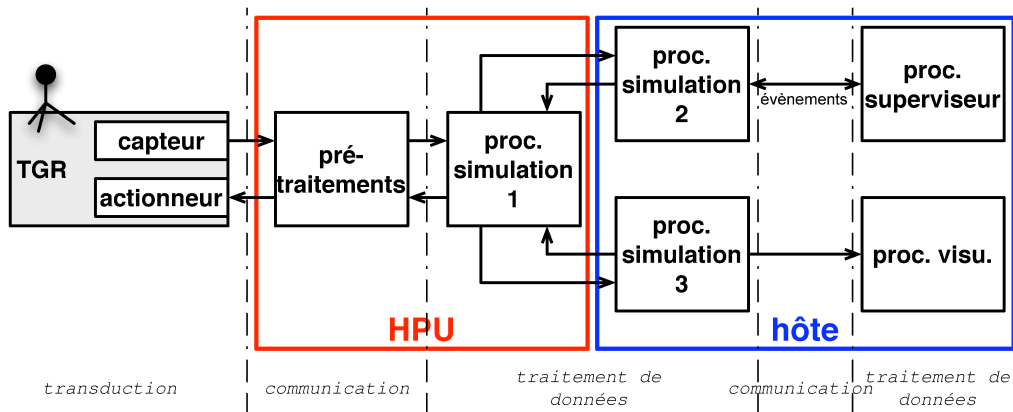


Figure 4.6. Le processus simulation est partagé sur le HPU et sur la machine hôte

3.4 Interface « geste » du HPU

Dans les configurations que nous avons évoquées ci-dessus, c'est l'interface geste du HPU qui permet la connexion du TGR au reste de l'architecture de simulation. Sans présager de la place du processus simulation (qu'il soit hébergé par le HPU ou non), l'interface geste constitue, dans toutes les configurations présentées ci-dessus, le canal de communication entre le processus simulation et

le TGR, et ses caractéristiques sont invariables. L'interface geste doit présenter des caractéristiques suffisantes pour la réalisation d'une boucle de simulation synchrone : temps de transfert des signaux minimaux, synchronisation sur une horloge extérieure.

3.4.1 Latence

L'interface geste jouant avant tout le rôle d'un canal de communication entre le processus simulation et le TGR, la latence avec laquelle les transferts de données peuvent s'effectuer est un point critique de l'architecture de simulation, et le critère principal de performance de l'interface geste du HPU. Ce critère est d'autant plus important si nous souhaitons réaliser des simulations à haute fréquence.

Dans une étude de 2005 portant sur l'étude des systèmes d'exploitation et des architectures informatiques pour la réalisation d'un simulateur multisensoriel haute performance, Florens, Castagné et al. [Luciani *et al.*, 2005a, Castagné *et al.*, 2005] préconisent que le temps nécessaire au transfert des données ne dépasse pas 5 μ s, afin de laisser 95 % de la période d'un pas de simulation à 10 000 Hz pour le calcul de la simulation proprement dit.

Selon le format des signaux échangés sur l'interface geste et les technologies employées, ce critère peut encore rester une valeur encore difficile à atteindre.

3.4.2 Synchronisation sur une horloge

La synchronisation entre le processus simulation, la communication avec le TGR et les éventuels traitements pré- ou post-simulation ne peut se faire qu'à l'aide d'un signal de synchronisation commun à toutes ces unités de traitement. Concrètement, cela signifie que l'interface geste doit pouvoir synchroniser les transferts de données sur un signal d'horloge externe.

3.4.3 Nombre de canaux de communication

Dans les cartes graphiques et les cartes son, le nombre de sorties correspond à des standards bien définis. Une carte graphique dispose d'un canal de sortie pour la sortie vidéo principale, plus éventuellement un canal supplémentaire de sortie vidéo. Les cartes son modernes se distinguent en deux catégories : (1) les cartes son grand public, intégrant deux entrées (ligne et micro) et de nombreuses sorties adaptées aux formats multimédia actuels (audiophonie 5.1, 7.1, etc.) ; (2) les cartes son professionnelles, intégrant de 2 à 16 lignes d'entrée et de sortie.

Dans le domaine haptique, nous sommes confrontés à des systèmes gestuels présentant une large versatilité de morphologie, ce qui signifie une grande variabilité dans le nombre de canaux de communication. Celui-ci sera en grande partie dépendant du nombre de degrés de liberté offerts par l'espace de travail d'un système haptique. Par rapport à ce critère, on distingue les grandes familles de systèmes suivantes :

- Systèmes 3D, de la famille du Phantom Desktop
- Systèmes 6D, correspondant à l'ajout des 3 rotations sur la manipulation d'un point 3D (Lhifam de CEIT, Virtuouse du CEA. . .)

- Systèmes redondants (le système VISHARD de TUM possède 10 degrés de liberté); ces systèmes sont toujours basés sur la métaphore du point (i.e. manipulation à 6 degrés de liberté)
- Exosquelettes, basés sur la morphologie humaine, dont le nombre de degrés de liberté est variable : le bras L-Exos de PERCRO à 5 degrés de liberté conçu pour les mouvements du bras ; le système CyberForce de Immersion permet d'intégrer tous les mouvements de la main dans une application haptique.

Cette grande diversité s'explique par la versatilité du geste [Luciani *et al.*, 2006a], qui couvre la manipulation d'un point dans un espace bi- ou tridimensionnel, jusqu'au jeu du piano⁶ (cf. chapitre 1). De fait, il est difficile de standardiser le nombre de canaux de communication nécessaires au HPU, puisqu'il dépend de l'application ou de la situation instrumentale visée et du transducteur gestuel utilisé (ces deux facteurs étant liés).

Nous pouvons tout de même observer que le minimum fonctionnel pour un système haptique actuel semble être 6 canaux de communication bidirectionnels indépendants. Le nombre de canaux de communications nécessaires est par nature supérieur à celui des périphériques vidéo ou audio.

3.4.4 Traitements pré- et post-simulation

En tant que canal de communication entre le TGR et le processus simulation, c'est l'interface geste qui doit prendre en charge les traitements préliminaires au calcul d'un pas de simulation, ainsi que les traitements de conclusion d'un pas de simulation. La couche logicielle de l'interface geste peut inclure un pilote de périphérique adapté au port de communication utilisé, et d'une manière générale il s'agit de traiter les opérations suivantes :

- De transmettre au simulateur des informations représentant l'état du transducteur gestuel,
- Vice versa, de transmettre au TGR des informations traitées par le simulateur,
- D'assurer la mise en correspondance des deux formats de signaux utilisés (1) en entrée et en sortie du simulateur et (2) en entrée et en sortie du transducteur gestuel. Cette « mise en correspondance » est nécessaire en particulier lorsque l'espace des signaux de sortie des capteurs du TGR n'est pas de même dimensionnalité que l'espace des variables d'entrée du processus simulation, et vice versa, lorsque l'espace des variables de sortie du processus simulation est différent de l'espace des signaux d'entrée des actionneurs du TGR. Si nous prenons l'exemple de la technologie ERGOS, les signaux obtenus en sortie des capteurs de position correspondent aux mouvements verticaux de chacune des tranches indépendantes. Si nous utilisons un stick 3D afin de manipuler des objets dans un espace tri-dimensionnel, une conversion cinématique est nécessaire afin de convertir le mouvement des trois tranches sur lequel est monté le stick 3D au mouvement tri-dimensionnel de l'extrémité du manche du stick, que nous utiliserons dans l'espace de modélisation. Une fois un pas de simulation calculé, la conversion inverse pour les forces est aussi nécessaire, permettant de convertir la commande en force représentée dans un espace 3D en une commande en force à appliquer à chacune des trois tranches ERGOS indépendamment.
- Permettre certains traitements sur les signaux d'entrée et de sortie du TGR : par exemple l'estimation du signal de vitesse, en l'absence de signal de vitesse fourni par le TGR.

⁶Le clavier de piano comporte 88 touches.

3.4.5 Interface « geste » analogique

Introduction Les ports de communication du TGR sont des points sensibles d'une plateforme multisensorielle de simulation. En fonction des solutions technologiques choisies pour les actionneurs et les capteurs du TGR, il est possible d'éviter l'étage analogique. En effet, supporter un étage analogique nécessite de concevoir une électronique de conditionnement des signaux en sortie des capteurs, et une électronique de puissance pour le contrôle des actionneurs à partir du signal de commande obtenu du processus simulation. La conception de cet étage analogique est relativement lourde et coûteuse. Depuis de nombreuses années, une partie des recherches en haptique (issues du domaine de la robotique et de l'automatique) travaille intensivement sur l'implantation de boucles de contrôle numériques permettant de remplacer l'étage analogique de puissance.

Cependant, la suppression de l'étage analogique entraîne des choix technologiques parfois incompatibles avec les performances nécessaires pour une interaction gestuelle de grande qualité. Dans le rapport d'activité sur la technologie haptique du réseau d'excellence Enactive, nous avons montré que pour des applications haptiques de haute qualité, il n'était pas encore possible actuellement de faire l'impasse sur un étage analogique ([Couroussé et Florens, 2005] et [Florens et Couroussé, 2007]).

Technologies pour la conversion analogique/numérique Nous n'allons pas ici présenter extensivement les technologies pour la conversion analogique/numérique ou numérique/analogique. Un aperçu des technologies de conversion pertinentes pour la simulation haptique est présenté dans [Florens et Couroussé, 2005] et [ENACTIVE, 2007a], et on pourra consulter [Prévot, 2004b, Prévot, 2004a, Prévot, 2004c] pour une présentation plus complète de l'état de l'art.

Sans plus entrer dans les détails, nous pouvons tout de même observer que les techniques de conversion se mesurent à l'aide du critère facteur de mérite, qui est le produit de 2 à la puissance du nombre de bits de conversion N_{eff} , multiplié par la fréquence d'échantillonnage $f_{\text{éch}}$ et divisé par la puissance consommée P (équation 4.1).

$$\text{Facteur de mérite} = 2^{N_{\text{eff}}} \times f_{\text{éch}} / P_{\text{consommée}} \quad (4.1)$$

Actuellement, la valeur du facteur de mérite est relativement constante quelque soit la technologie utilisée, mais certaines technologies permettent une meilleure précision de conversion (nombre de bits de conversion efficaces élevés) au détriment du temps de conversion, alors que d'autres technologies fournissent des temps de conversion extrêmement courts au détriment de la précision. Remarquons également que pour une même fréquence d'échantillonnage, les performances des convertisseurs numérique-analogique sont meilleures de 2 bits efficaces environ par rapport aux convertisseurs analogique-numérique.

On distingue plusieurs grandes familles de convertisseurs :

- Les convertisseurs à haute précision (24 bits ou plus) avec un temps de conversion relativement long, utilisés par exemple dans les systèmes de pesée.
- Les convertisseurs pour les applications audio haute-fidélité, avec une précision d'une vingtaine de bits pour une fréquence d'échantillonnage à 44,1 kHz.

- Les convertisseurs pour les applications vidéo ou des télécommunications, avec une précision de 8 à 10 bits pour une fréquence d'échantillonnage de plusieurs MHz.
- Dans la plage de précision intéressante pour les utilisations en haptique, c'est-à-dire autour de 16 bits, les fréquences d'échantillonnage sont de l'ordre de 100 kHz (jusqu'au MHz pour une précision de 12 à 14 bits), c'est-à-dire un temps de conversion entre 1 et 10 μ s.

Précision La résolution fournie par les capteurs de mouvement en haptique est souvent supérieure à celle des capteurs de robotique : alors qu'une précision de 10 à 12 bits est largement suffisante en robotique, une précision supérieure est nécessaire dans le domaine de l'haptique pour tenir compte des seuils de perception humains et de la dynamique des phénomènes représentés. En effet, là où une application robotique peut s'accommoder de petites vibrations dans l'asservissement des organes mobiles, les mêmes vibrations modifieraient complètement le confort de manipulation, voire l'objet virtuel inféré par l'utilisateur dans une utilisation haptique.

Dès 1937, David Katz observe que la sensibilité tactile permet de détecter des différences d'épaisseurs de moins de 0,01 mm entre deux feuilles de papier [Katz, 1989], p147. Tan & al. [Tan *et al.*, 2006] ont montré les seuils de discrimination pour la perception de textures de 50 μ m d'amplitude étaient de 3,90 μ m en situation réelle (au toucher) et de 5,05 μ m en situation virtuelle (interaction outillée).

Dans le cas de la technologie ERGOS, la plage de mouvements est variable : de 20 mm à 150 mm par axe selon l'habillage mécanique utilisé (cf. annexe A). Pour obtenir une résolution inférieure à 5 μ m quel que soit l'habillage mécanique utilisé, c'est-à-dire pour une plage de mouvement utile de 150 mm, il est alors nécessaire d'échantillonner le signal de position sur 15 bits au minimum. En effet, si nous notons n le nombre de bits sur lesquels le signal de position est discrétisé, d la plage de déplacement totale et p la précision, alors $n = 14,87$ avec $d = 150 \times 10^{-3}$ m et $p = 5 \times 10^{-6}$ m (équation 4.2).

$$2^n = \frac{d}{p} \quad (4.2)$$

Une précision élevée pour les signaux de position est aussi nécessaire parce que le mouvement humain demande une *échelle dynamique* importante. Dans le cas de la percussion, par exemple, les mouvements d'entraînement du percuteur peuvent être des mouvements de grande amplitude (mouvement du poignet ou de l'avant-bras), alors que les phénomènes se manifestant au niveau de la structure vibrante percutée sont de l'ordre de grandeur du micromètre. Ainsi, une simulation haptique exemplaire pour cette situation instrumentale devrait rendre possible, en effleurant la surface de l'objet percuté, de percevoir gestuellement la vibration du corps sonore. Le frottement d'archet est un autre exemple où l'instrumentiste exécute des mouvements de grande amplitude — plusieurs dizaines de centimètres — pour exciter la corde, mais où les phénomènes vibratoires qui ont lieu entre la corde et l'archet sont de l'ordre de grandeur inférieur au millimètre. Les signaux gestuels doivent représenter la superposition de ces deux mouvements, ce qui demande une grande dynamique, donc une grande précision dans le format utilisé pour les données.

Latence Nous avons vu ci-dessus que l'utilisation d'une précision de conversion de 16 bits est souhaitable, mais que les temps de latence de communication des entrées/sorties du processus

simulation ne doit pas dépasser $5\ \mu\text{s}$ idéalement. Ce temps de latence, dans le cadre de l'interface geste analogique du HPU, comprend le temps de conversion et le temps de transfert des données numériques des buffers des convertisseurs au processus simulation. Compte tenu de l'état actuel des technologies pour la conversion, une telle performance reste encore dans le haut-de-gamme des produits sur le marché.

Échantillonnage simultané Il est courant d'utiliser des techniques de multiplexage afin de réduire le nombre de convertisseurs utilisés sans réduire le nombre de canaux A/N et N/A disponibles. Dans le cas d'une utilisation pour l'haptique, l'utilisation du multiplexage n'est pas envisageable car elle conduit inévitablement à augmenter le temps de conversion total des entrées/sorties.

3.4.6 Interface « geste » numérique

Introduction La réalisation d'une interface geste possédant un port de communication au format numérique ne pose plus de problèmes particuliers aujourd'hui. Faire fonctionner un port de communication numérique à plus de quelques centaines de MHz ne pose plus de problèmes aujourd'hui, et le temps de montée des signaux sur les ports numérique ne dépasse pas quelques nano-secondes. Ces caractéristiques permettent donc d'établir des protocoles de communication synchrones avec des délais de transmission faibles. La difficulté de l'utilisation du format numérique sur l'interface geste reside plutôt à l'heure actuelle dans le choix limité de technologies, autant pour l'actuation que pour la capture de mouvement.

Capteurs numériques Les capteurs numériques sont courants dans les applications haptiques, et la technologie la plus largement répandue est la mesure optique. Les capteurs optiques sont alors le plus souvent directement montés sur l'arbre du moteur. On en trouve principalement deux types de capteurs optiques : (1) basés sur le comptage (capteurs incrémentaux), ou bien (2) basés sur la lecture d'une information absolue (capteurs absolus). On trouvera une présentation plus complète des solutions en capture numérique dans [Jean-Loup Florens, 2004], [Florens et Couroussé, 2005] ou [ENACTIVE, 2007a].

Capteurs de position incrémentaux Ces types de capteurs sont largement répandus sur le marché, mais la précision standard ne dépasse guère 10 ou 12 bits. Par interpolation de quadrature par exemple, il est maintenant possible d'améliorer quelque peu la précision. En outre, l'utilisation de capteurs incrémentaux demande une électronique numérique travaillant à une grande fréquence d'horloge afin de compter si la bande passante du phénomène mesuré est importante.

Capteurs de position absolus Les capteurs absolus sont eux aussi répandus aujourd'hui, et ont l'avantage de fournir un signal qui peut-être traité à plus basse fréquence, parce que donnant immédiatement la position mesurée dans un référentiel absolu. En revanche, la résolution maximale possible ne va guère au-delà de 13 bits.

Latence due au format de sortie des encodeurs Le délai temporel après lequel la donnée est obtenue à la sortie d'un encodeur numérique est directement dépendant de sa fréquence d'horloge. À l'inverse, la mesure est obtenue avec un délai négligeable à la sortie d'un encodeur analogique, mais la conversion des données analogiques vers le format numérique introduira un délai similaire à celui obtenu dans la solution utilisant l'encodeur numérique. De ce point de vue, les deux solutions numérique et analogique sont équivalentes.

Commande numérique des actionneurs : la PWM L'utilisation de la modulation en largeur d'impulsion (PWM) ou de techniques dérivées a parfois été présentée comme une alternative à l'utilisation de signaux analogiques pour les applications haptiques. L'intérêt de l'amplification PWM est dû à son excellent rendement énergétique, mais aussi au fait que certains moteurs DC brushless sont dotés d'une entrée acceptant les signaux PWM. Par contre, la modulation PWM est aussi connue pour générer du bruit, qui est perçu comme un effet rugueux lorsqu'il est utilisé dans les interfaces haptiques. L'équipe de recherche de la Rutgers University ont utilisé la modulation PWM pour le gant à retour d'effort Rutgers Master II-ND [Bouzit *et al.*, 2002a]. Pour ce système, les actionneurs étaient des cylindres pneumatiques, dont la commande de position était indiquée par un signal PWM à quelques centaines de Hz (300 Hz dans [Bouzit *et al.*, 2002b], 500 Hz dans [Bouzit *et al.*, 2002a]). Le temps de réponse relativement important des électrovannes (environ 2 ms) permet en effet l'utilisation d'une faible fréquence pour la PWM, et permet de filter correctement les transitoires PWM à haute fréquence.

Au Queen's University de Kingston, Canada, Randy Ellis, et son équipe ont conçu une interface haptique utilisant des moteurs brushless comme actionneurs [Ellis *et al.*, 1996]. Le choix fut fait d'utiliser des amplificateurs linéaires (analogiques) plutôt que la PWM, et ce malgré une plus forte dissipation calorimétrique et un encombrement plus important, parce que les amplificateurs PWM faisaient entrer en résonance des modes audibles et perceptibles par le toucher, même avec une fréquence de commutation de 20 kHz.

Çavuşoğlu, Feygin et Tendick publient en 2002 une analyse critique des propriétés mécaniques et électriques du PHANTOM [Çavuşoğlu *et al.*, 2002]. Une des limitations les plus importantes de ce système présentées dans cet article est l'utilisation d'actionneurs dont la commande est modulée par PWM ; les auteurs présentent la modulation PWM comme inadéquate pour les applications haptique à haute performance. En effet, la commutation de la PWM génère des composantes à haute fréquence sur le signal de commande des actionneurs, ce qui peut entraîner des instabilités par effet de résonance, ou générer un bruit perceptible pour l'utilisateur, alors qu'une amplification linéaire ne présenterait pas ces effets. Les auteurs montrent que, en remplaçant l'amplification PWM par des amplificateurs linéaires de courant, le bruit tactile a disparu et les performances dynamiques du Phantom sont améliorées (« *Implementing the high-performance, linear current amplifiers eliminated the effects of motor cogging and allowed us to increase system bandwidth without exciting the resonances of the system.* »).

3.5 Interface « simulateur » du HPU

3.5.1 Le temps de transfert n'est pas inversement égal au débit maximal d'un bus numérique

On compare souvent, à tort, la relation inverse entre fréquence d'échantillonnage et période d'échantillonnage à la relation entre la latence et le temps de transfert sur un bus numérique. Or, dans le domaine numérique, le temps de transfert n'est pas inversement égal au débit maximal. En effet, la plupart des protocoles de communication actuels utilisent des modes de communication par paquets : il est ainsi possible de transférer un volume important de données dans une fenêtre temporelle réduite, mais entre deux trains successifs de données, un délai important peut s'écouler permettant à l'émetteur et au récepteur de paramétrer la prochaine communication. Ou si nous utilisons une analogie ferroviaire, ce n'est pas parce qu'un train de marchandises peut transporter 24 wagons de Lyon à Brest en 24 heures, que un seul wagon de marchandises mettrait seulement une heure pour effectuer le même trajet !

Si nous insistons sur ce point, c'est parce que tous les standards pour les ports de communication actuels conviennent parfaitement pour la simulation haptique si nous nous référons au débit maximal supporté. En effet, le processus simulation traite à chaque pas de calcul n variables d'entrées, où n correspond généralement au nombre de degrés de liberté du TGR utilisé, soit généralement 3 ou 6, et jamais plus de 16. Le nombre de variables de sorties produites par le processus simulation à destination du TGR est souvent égal au nombre de variables d'entrées. Enfin ces variables sont le plus souvent codées sur 32 bits, voire 6 bits pour les architectures de calcul les plus récentes. Dans le cas le plus défavorable, le canal de communication entre le TGR et le processus simulation doit supporter deux flux de données présentant un débit de $16 \text{ canaux} \times 64 \text{ bits}$, soit 128 octets par pas de simulation, c'est-à-dire 125 ko/s pour une fréquence de simulation de 1 kHz, et 12,5 ko/s pour une fréquence de simulation de 10 kHz. Ces chiffres sont donc largement en deçà des débits théoriques maximaux pour les bus de communication courants : 132 Mo/s pour le bus PCI 32 bits à 33 MHz, respectivement 12 Mo/s et 480 Mo/s pour les bus USB1.1 et USB2.0, et entre 100 et 3 200 Mo/s pour le bus IEEE 1394 suivant les normes.

3.5.2 Interface « simulateur » connectée au processus superviseur

Lorsque le processus simulation est hébergé sur le HPU, l'interface simulateur du HPU communique avec le processus superviseur. Interface « simulateur » connectée au processus simulation.

Lorsque le processus simulation n'est pas hébergé sur le HPU, mais exécuté par une autre unité de calcul de l'architecture de simulation, l'interface simulateur devient partie intégrante de la boucle de simulation. Les temps de transfert des données au travers de l'interface simulateur viennent s'ajouter aux temps de transfert des données dans l'interface geste du HPU. Les contraintes que nous avons présentées précédemment à l'interface geste s'appliquent aussi dans ce cas à l'interface simulateur. En particulier, les temps de transfert doivent être les plus courts possibles.

3.5.3 Bus de données pour l'interface simulateur

Le choix du bus de données pour l'interface simulateur n'est critique que dans le cas où le processus simulation n'est pas hébergé par le HPU. Dans un tel cas, nous avons vu que ce sont essentiellement les temps de transferts de données qui sont critiques, alors que les débits nécessaires dans notre cas seront largement inférieurs à 1 Mo/s, c'est-à-dire parfaitement accessibles au regard des performances des performances des bus de données actuels.

Ethernet Ethernet est parfois utilisé comme support de communication au sein même d'une architecture pour la simulation multi-sensorielle en temps réel. Il présente l'avantage de permettre une grande flexibilité dans la conception d'architectures modulaires et réparties. La plupart des architectures de calcul, même embarquées, peuvent aujourd'hui se connecter aisément sur un réseau Ethernet par le biais d'extensions matérielles adaptées.

Dans le cadre de la simulation haptique, Traylor et al. ont présenté une étude de l'utilisation du protocole UDP pour la simulation haptique [Traylor *et al.*, 2005]. Dans cette étude, le système haptique était contrôlé par une architecture embarquée (un microcontrôleur 8 bits relié à une carte Ethernet adaptée) en réseau local avec le simulateur (Pentium 2,8 GHz et une carte Ethernet Gigabit). La fréquence maximale atteignable était inférieure à 2 kHz pour une boucle de simulation simple (le simulateur se met en attente des données du contrôleur haptique, calcule un pas de simulation, puis envoie les nouvelles consignes au contrôleur haptique), et de l'ordre de 3,8 kHz si les boucles étaient entrelacées. Le temps de transmission était de 61 μ s, et les temps de traitement des données pour l'émission et la réception uniquement étaient de 230 μ s et de 155 μ s respectivement pour le contrôleur et pour le simulateur.

Au regard de ce simple exemple, il apparaît que l'utilisation d'Ethernet comme support de communication au sein d'une architecture pour la simulation haptique n'est pas possible. Premièrement, les temps de transfert et de traitement en émission comme en réception sont trop importants. Sans tenir compte du temps de calcul de la simulation, la fréquence maximale d'une boucle de type lecture des positions, calcul du modèle, envoi des forces est déjà inférieure à 2 000 Hz dans l'étude de Traylor et al.

De plus, à notre connaissance, il est impossible sur un réseau Ethernet de garantir les temps de communication. Dans le protocole TCP, l'émetteur reçoit une confirmation du récepteur pour chaque paquet de données transmis, mais l'envoi d'un paquet de confirmation est coûteux et en cas de perte de donnée, le paquet perdu doit être transmis à nouveau avant de pouvoir continuer la transmission ; cette dernière contrainte est rédhibitoire pour le temps réel. Le protocole UDP est un protocole simplifié ce qui le rend adapté aux applications en temps réel relâché où la rapidité de la transmission est un facteur important (par exemple : voix sur IP). Le protocole UDP ne prévoit pas de protocole d'acquittement et il n'y a aucun moyen pour le récepteur de savoir si des paquets ont été perdus. Le temps de transmission n'est pas non plus garanti.

Le bus PCI Le bus PCI (Peripheral Component Interconnect) est un bus interne, dont la première spécification fut publiée en 1992. Le bus PCI est un bus synchrone, prévu pour travailler avec des systèmes 32 ou 64 bits, dans un fonctionnement maître-esclave sous le contrôle d'un arbitre intégré

au bus. La fréquence d'horloge du bus est de 33 MHz, mais est portée à 66 MHz ou plus dans les nouvelles spécifications.

Un transfert PCI est constitué de plusieurs phases : prise de possession du bus par le maître et initialisation d'une transaction, phase d'adressage et phase de transfert des données. Le temps total d'un transfert PCI est variable. Il dépend à la fois des cartes électroniques ou des processeurs impliqués dans le transfert, et des temps de réponse au moment de la transaction. La norme PCI prévoit un protocole de base avec des échanges de longueur variables ; il est en effet permis d'insérer des états d'attente de longueur variable, mais les délais sont limités pour une cible à seize période d'horloge pour la première réponse, puis huit pour les suivantes, et pour le maître à huit périodes. Si aucune attente n'a lieu, un cycle de lecture occupe huit période d'horloge et un cycle d'écriture occupe aussi huit périodes d'horloges, soit un temps de cycle de 0,24 ns pour une fréquence d'horloge de 33 MHz.

Au regard des latences sur les temps de transferts comme sur les débits théoriques maximum, le bus PCI est le candidat idéal pour supporter l'interface simulateur du HPU.

Le bus USB La première spécification du bus USB 1.0 fut présentée en 1996. Le bus USB est un bus externe, à la différence des bus ISA, VME ou PCI. Deux caractéristiques importantes du bus USB et qui en ont fait sa popularité sont la possibilité d'alimenter un périphérique directement par le câble de liaison USB, et la possibilité de connexion ou de déconnexion à chaud. Les débit maximaux théoriques sont respectivement de 12 Mbits/s et de 480 Mbits/s pour les normes USB 1.1 et USB 2.0.

Il s'agit d'un bus maître-esclave, et c'est le maître qui gère le réseau de bus, notamment lors de la connexion ou de la déconnexion d'esclaves. Le signal d'horloge est transmis avec les données, codé en NRZI : un paquet SYNC permet aux récepteurs de synchroniser leurs horloges avant chaque paquet de données. Quatre types de services sont définis :

- service de transfert isochrone : paquets de taille réduite, transferts cycliques avec délai court
- service de transfert d'interruption : messages courts, transferts périodiques (éventuellement multi-cycliques)
- service de transfert bulk : sans garantie de délai, mais avec acquittement, permettant le transfert de grandes quantités de données
- service de transfert contrôle : non-périodique avec acquittement, tolérant des latences importantes, plus destiné à la configuration du réseau USB

Chaque transfert a lieu entre deux évènements marquant un cycle de bus. Le cycle a une durée fixe de 1 ms pour la version USB 1.x, et de 125 μ s pour la version USB 2.0. Tous les types de services peuvent avoir lieu dans un même cycle.

Si nous considérons l'utilisation du bus USB comme candidat potentiel pour l'implantation de l'interface système du HPU, et en supposant que des transferts bidirectionnels puissent avoir lieu dans un même cycle, la période la plus courte de rafraîchissement est alors de 125 μ s. Dans le cas de la communication avec un processus simulation synchrone au travers de l'interface système, la fréquence de simulation sera limitée à 8 kHz.

Le bus FireWire (norme IEEE 1394) A l’instar du bus USB, le bus FireWire est un bus externe. L’appellation FireWire fut créée par Apple, son concepteur, à la fin des années 1980. Le nom IEEE 1394 correspond à la norme sous laquelle est défini le bus FireWire.

Le bus FireWire est un bus de communication sériel numérique prévu pour le haut débit. Son débit maximal théorique va actuellement jusqu’à 800 Mbits/s, mais les spécifications prévoient un débit maximal théorique jusqu’à 3 200 Mbits/s.

Dans certains domaines d’applications, le bus FireWire est en concurrence directe avec le bus USB, puisque ces deux bus présentent des caractéristiques relativement proches : connexion ou déconnexion à chaud, alimentation via le câble de données. Cependant, à la différence du bus USB, le bus FireWire permet le multi-arbitrage, et garantit la bande passante allouée à un périphérique, ce qui en fait un bus couramment utilisé dans les applications vidéo, qui nécessitent en débit important et constant.

Les transferts FireWire sont répartis dans le temps par cycles de 125 μ s. Durant un cycle, il est possible de procéder à plusieurs échanges pour des services différents. Quatre différents types de services sont possibles : un service synchrone, et trois services asynchrones (lecture, écriture et verrou (*lock*)).

Avec un temps de cycle similaire au bus USB 2.0 de 125 μ s, le bus FireWire permet l’implantation de boucles de simulation synchrones jusqu’à 8 kHz seulement.

3.6 Le processeur du HPU

Nous avons vu que le HPU doit nécessairement être équipé d’une unité de calcul pour effectuer les opérations de traitements nécessaires à la communication avec le TGR (section 3.4.4 de ce chapitre, page 92). Aussi, cette unité de calcul doit permettre au HPU de supporter éventuellement l’exécution d’un processus simulation. Le HPU est donc un composant du simulateur qui peut être soumis à une lourde charge de calculs, alors que d’autre part le HPU est confronté aux contraintes similaires à celles des systèmes embarqués, essentiellement en termes de limitation de taille, ce qui réduit dramatiquement le choix des micro-processeurs utilisables pour le HPU.

Il n’existe pas à l’heure actuelle de consensus sur les techniques de simulation pour la simulation physique et il n’en existera vraisemblablement jamais, mais quelques processeurs dédiés à la simulation physique ont été développés : on les appelle PPU, pour *Physics Processing Unit*. Actuellement, un seul PPU existe sur le marché : il s’agit du processeur PhysX, originellement développé par la société Ageia (ensuite rachetée par Nvidia). Certaines cartes graphiques de Nvidia sont maintenant équipées du processeur PhysX, et une petite centaine de jeux vidéo peuvent maintenant exploiter ce processeur⁷.

Nous envisageons essentiellement le HPU pour sa fonction de communication entre le TGR et le simulateur. Il s’agit néanmoins d’une fonction de communication intelligente puisqu’elle peut inclure la simulation physique. Cependant, l’utilisation d’une architecture de calcul adaptée à la simulation physique peut permettre au simulateur de faire l’économie de la charge du calcul de la simulation physique en laissant le HPU prendre totalement en charge cette fonction. Le HPU endosserait alors la fonction supplémentaire de *co-processeur*. Depuis quelques années, on voit

⁷Source : http://www.nvidia.com/object/nvidia_physx.html

apparaître dans le domaine de l'infographie de nouvelles technologies permettant de développer les fonctionnalités de co-processing des cartes graphiques. Les fabricants de cartes graphiques viennent ainsi concurrencer les CPU sur leur terrain en permettant l'utilisation des GPU pour des tâches qui ne sont pas nécessairement en relation avec le calcul graphique. À titre d'exemple, citons la technologie logicielle CUDA (*Compute Unified Device Architecture*)⁸, développée par nVidia, qui permet d'utiliser un GPU pour des tâches de calculs généralement destinées à un CPU.

4 Conclusion

La technologie pour le retour d'effort se trouve actuellement dans une phase d'ébullition, où une multitude de nouvelles orientations scientifiques et technologiques apparaissent, mais qui ne laisse pas encore présager de la future organisation de ce champ de recherche. Face à la versatilité du phénomène gestuel, à la versatilité des applications du geste utilisé dans l'ordinateur et à l'émergence rapide de nouvelles technologies, il est difficile de mettre sur pied les futurs standards technologiques.

Faisant le constat qu'un composant matériel est néanmoins nécessaire entre le TGR et le reste de l'architecture de simulation, de même que les composants « carte son » et « carte graphique » sont maintenant un standard informatique, nous avons présenté dans ce chapitre le concept de HPU. Il s'agit du composant matériel qui permet d'interfacer un TGR avec une architecture de calcul. Le HPU se compose donc de deux interfaces : une interface « geste » pour la connexion avec le TGR, et une interface « simulateur » pour la connexion avec le simulateur hôte. L'interface geste doit nécessairement pouvoir supporter la synchronisation avec les entrées/sorties du TGR et supporter l'échange de signaux bidirectionnels avec une faible latence de transfert. Idéalement, l'interface geste doit posséder à la fois des entrées/sorties numériques et analogiques par le biais de convertisseurs analogiques-numériques pour pouvoir être interfacée avec des TGR présentant des entrées/sorties analogiques ou numériques.

Le HPU doit posséder une unité de calcul dédiée à l'exécution des traitements nécessaires à la communication avec le TGR et à l'exécution éventuelle d'un processus simulation. Lorsque tout ou partie du processus simulation est exécuté sur le simulateur hôte, l'interface simulateur du HPU doit posséder les mêmes caractéristiques de réactivité que l'interface geste afin de permettre l'exécution du processus simulation en respectant les contraintes de réactivité et de synchronisme. En définitive, parmi les types de bus de données les plus répandus aujourd'hui, seul le bus PCI (et ses successeurs : PCI-X, PCI Express, etc.) présente des caractéristiques adéquates.

Observons enfin que *qui peut le plus, peut le moins*, l'interface simulateur du HPU, telle que nous l'avons définie, supporte l'implantation des architectures de simulation suivant l'approche asynchrone (cf. chapitre 3).

Dans le chapitre suivant, nous présentons la réalisation de l'architecture de simulation baptisée ERGON_X. Cette architecture est construite à partir de la concrétisation du concept de HPU dans un composant matériel. Dans cette architecture, le processus de simulation est exécuté sur le HPU. La machine hôte, qui héberge le HPU, est alors libérée des contraintes d'exécution temps réel

⁸http://www.nvidia.com/object/cuda_home.html

nécessaires pour le processus de simulation, et nous se satisfait d'une machine disposant d'une faible puissance de calcul.

Ce composant nous a permis d'implanter une simulation multisensorielle synchrone tout en garantissant les conditions d'exécution du processus de simulation en temps réel dur et synchrone avec les communications geste et son.

Face au nombre croissant de nouveaux systèmes à retour d'effort développés chaque année, et plus généralement devant le développement et la diversification des applications en mécatronique, il est à supposer que les années à venir verront apparaître la commercialisation de HPU et peut-être des standards d'architectures et/ou de connectivité. La carte Q8/HIL de Quanser est un exemple du futur composant HPU : Quanser est une société spécialisée dans la mécatronique qui développe en particulier depuis quelques années une activité de développement et de commercialisation de nouvelles technologies pour l'haptique. Quanser commercialise la carte « Q8 Hardware in the Loop (H.I.L.) », destinée à être intégrée dans des applications de mécatronique et en particulier des simulateurs haptiques. Cette carte présente toutes les caractéristiques du HPU : elle présente une interface geste équipée d'un port analogique et d'un port numérique, et comprend :

- 8 canaux de conversion AN avec une précision de 14 bits (avec extension à 16 bits) multiplexés (temps de conversion $2,4 \mu s$ par canal)
- 8 canaux de conversion NA avec une précision de 12 bits
- 8 canaux d'entrée pour capteurs de rotation (24 bits)
- un port numérique sur 32 bits
- 2 sorties de contrôle PWM
- 2 horloges
- Synchronisation par horloge interne ou externe
- Développement d'applications sous Simulink/Matlab ou WinCon ; WinCon est une application développée par Quanser pour le développement d'applications temps réel sous Windows XP.

Chapitre 5

ERGON_X : un Simulateur Multisensoriel Compact et Transportable

Sommaire du Chapitre

1	Introduction	103
2	Architecture matérielle du simulateur	105
2.1	La carte TORO de Innovative Integration	105
2.2	Organisation fonctionnelle	106
2.3	Composition d'un simulateur-type ERGON_X	108
3	Architecture logicielle	109
3.1	Pilote de périphérique — la bibliothèque <code>libtd</code>	109
3.2	Développement d'une API — <code>libErgonSim</code>	110
3.3	Exécution d'un pas de simulation et chaîne de transmission des signaux	111
3.4	Communication hôte-DSP	113
3.5	Synchronisation du calcul de la simulation	114
3.6	Le calcul d'un pas de simulation	115
3.7	Automate d'états du simulateur	115
4	Conclusion	116

1 Introduction

Au cours du chapitre 1, nous avons présenté les propriétés du geste instrumental, et nous avons avancé l'hypothèse qu'il était possible de restituer les propriétés de cette catégorie particulière de gestes dans l'interaction de l'homme avec un objet virtuel. Les chapitres 2 et 3 ont mis en avant la nécessité, pour implanter ce geste instrumental, de disposer d'un simulateur qui possède des propriétés particulières, à savoir la réalisation d'une boucle de simulation synchrone avec les entrées/sorties des transducteurs. Cette boucle de simulation synchrone nous permet d'aborder la question de la restitution des corrélations authentiques entre les signaux sensoriels produits sous

l'angle plus écologique de la multisensorialité intégrée, plutôt que sous l'angle de la reconstruction cognitive de ces corrélations par la multimodalité conventionnelle, c'est à dire la superposition des modalités. En d'autres termes, le degré "d'instrumentalité" que nous pourrions atteindre avec ce simulateur dépendra fortement du degré d'intégration multisensorielle. Le chapitre 4 présente alors le HPU comme le maillon essentiel de cette chaîne de simulation en garantissant l'exécution du processus de simulation de manière synchrone avec les entrées et les sorties des transducteurs.

De plus, pour obtenir une simulation multisensorielle qui possède les qualités instrumentales requises par nos exigences, cette boucle de simulation doit présenter une excellente réactivité. En d'autres termes, le degré "d'instrumentalité" que nous pourrions atteindre avec ce simulateur dépendra fortement de la fréquence de simulation maximale atteignable.

Ce chapitre présente les principes et les techniques du simulateur, nommé ERGON_X, que nous avons réalisé pour répondre à ces attentes : réactivité, compacité et portabilité.

Ainsi, tous nos choix conceptuels et technologiques ont été orientés de manière à obtenir d'abord et avant tout une fréquence de simulation la plus élevée possible, plutôt que de donner la priorité à la capacité de calcul. En d'autres termes, pour la raison fondamentale de disposer d'un véritable outil d'évaluation des propriétés de la situation instrumentale ergotique, encore inexistant, et contrairement aux développements usuels des machines informatiques, nous avons fait le choix de privilégier la réactivité interne et en entrées / sorties plutôt que la puissance de calcul.

Le chapitre 6, qui présente les simulations instrumentales obtenues grâce à ce simulateur, montre que le choix d'une fréquence de simulation très élevée et d'un synchronisme poussé nous a permis d'obtenir des situations particulièrement relevantes de l'enaction, et qui présentent des caractéristiques essentielles pour le jeu instrumental.

Dans le cas de l'architecture TELLURIS, que nous avons présentée dans le chapitre 3 (page 76), le système de conversion analogique-numérique représentait un des verrous les plus importants à lever pour pouvoir améliorer les performances de simulation. En effet, le temps de conversion des signaux et de leur transmission entre le TGR et le processus simulation occupait une part non négligeable de l'ensemble de la période de simulation. Les convertisseurs utilisés présentaient une latence de conversion trop importante — $9\text{ }\mu\text{s}$ par canal pour la conversion A/D, la conversion de chaque canal se faisant un à un ; $10\text{ }\mu\text{s}$ pour l'ensemble des canaux D/A —, ainsi qu'une résolution de 12 bits qui demandait également à être améliorée. En raison des temps de conversion importants, il n'était pas possible de porter la fréquence de simulation à plus de 3 kHz. Le simulateur ERGON_X résout cette limitation de TELLURIS : les convertisseurs utilisés présentent une latence faible — inférieure à $5\text{ }\mu\text{s}$ — pour une précision de 16 bits.

Pour pallier à ce verrou important du simulateur TELLURIS, le laboratoire a acheté à la fin de l'année 2003 une nouvelle carte d'acquisition et de génération de signaux performante : la carte TORO de Innovative Integration. Cette carte était dans un premier temps destinée à augmenter la réactivité du simulateur en l'équipant de canaux de conversion A/N et N/A beaucoup plus rapides. Rapidement, les développements techniques autour de la carte TORO nous ont incité à concevoir une nouvelle utilisation de cette carte comme un composant autonome d'une plate-forme de simulation multisensorielle, et à affiner le concept de HPU tel que nous l'avons présenté dans le chapitre précédent. La carte TORO, utilisée en tant que HPU, fut intégrée dans le développement d'un nouveau simulateur autonome et transportable qui fait l'objet d'une partie de cette thèse :

ERGON_X, mais aussi dans le développement de la nouvelle version du simulateur TELLURIS (thèse de J. Castet, en cours).

La réalisation du simulateur ERGON_X se situe dans la continuité des travaux de recherche et développement pour la simulation multisensorielle du groupe de recherche ACROE & ICA, en reprenant les principes fondamentaux de toutes les plate-formes pour la simulation multisensorielle réalisées par le laboratoire, et en particulier de TELLURIS : la simulation est synchronisée avec les entrées/sorties du TGR et basée sur le formalisme CORDIS-ANIMA.

Dans le cadre de ERGON_X, nous avons fait le choix de tirer pleinement parti de la puissance du HPU, en exécutant le processus simulation sur le DSP de la carte TORO. L'intérêt est de libérer le processeur de la machine hôte des contraintes du temps réel, le processus de simulation étant exécuté sur le processeur du HPU. Le DSP présent sur la carte TORO est une architecture de calcul peu puissante, puisque avec seulement 900 Mflops en puissance de crête, elle présente des performances de calcul qui se trouvent en deçà de celles des microprocesseurs les plus courants qui équipent les ordinateurs pour le grand public. Mais nous avons tiré avantage de la proximité du DSP avec les convertisseurs A/N et N/A pour augmenter considérablement la réactivité globale de la boucle de simulation, et réaliser ainsi des simulations synchrones incluant le retour d'effort jusqu'à 44,1 kHz. Une telle fréquence représente encore aujourd'hui une limite difficilement atteignable avec des contraintes de temps réel dur, et en particulier dans le cadre de la simulation haptique.

2 Architecture matérielle du simulateur

2.1 La carte TORO de Innovative Integration

Une première étude fut effectuée en 2002–2003, portant sur l'ensemble de la chaîne de communication de données entre le calculateur et l'interface geste du simulateur TELLURIS. Ce travail fut mené par Y. Chara sous la tutelle de J.-L. Florens. Une première carte de conversion équipée de convertisseurs 16 bits $\Delta\Sigma$ fut testée, mais abandonnée parce que cette technologie de conversion présentait des temps de conversion beaucoup trop grands (temps de conversion total de 700 μ s pour 16 canaux).

C'est finalement la carte TORO de Innovative Integration qui fut choisie (figure 5.1) à l'issue de cette étude. Il s'agissait de la carte la plus performante que nous pouvions trouver sur le marché à la fin de l'année 2003. La carte TORO est une carte d'acquisition et de génération haut de gamme, développée pour supporter deux modes de fonctionnement : (1) acquisition en boucle ouverte, permettant le post-traitement des données en temps différé sur le DSP puis mémorisation sur la carte ou envoi au host par paquets via le port HPI ; (2) une utilisation de type « contrôle » pour la robotique (boucle fermée), pour l'utilisation de boucles d'acquisition, traitement puis génération de signaux en boucle fermée. À cette fin, la carte possède un mode de fonctionnement spécifique appelé « servo » permettant l'implantation de boucles de contrôle à faible latence et l'utilisation des convertisseurs avec une réactivité maximale. C'est ce dernier mode de fonctionnement qui nous intéresse dans le cadre de son implantation dans l'architecture du simulateur ERGON_X.

La carte TORO possède les caractéristiques suivantes [Tor04] :

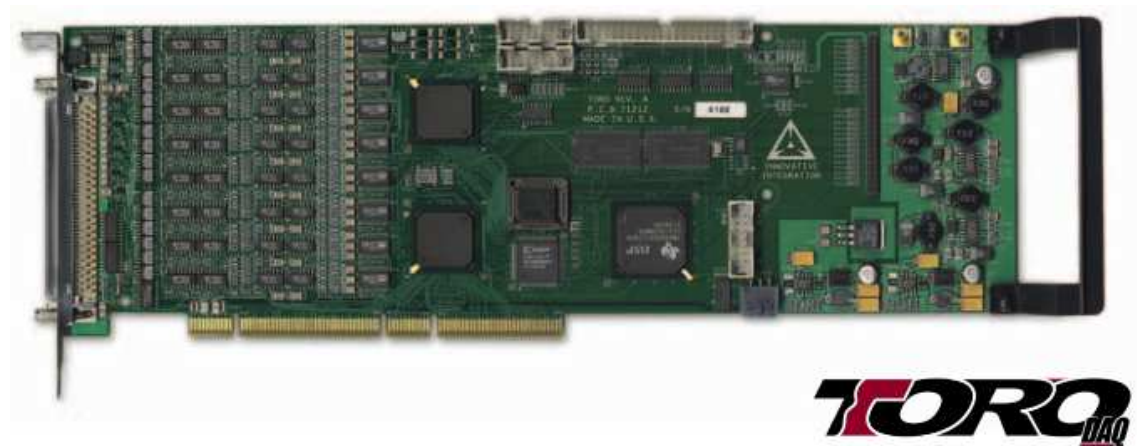


Figure 5.1. La carte TORO (Innovative Integration)

- Une interface PCI 32 ou 64 bits ; le port PCI est le port de communication le plus répandu actuellement, avec le format USB, et permet d'utiliser la carte sur la majorité des architectures de calcul existant actuellement sur le marché.
- 16 convertisseurs A/N [Technology, 2000b] à 16 bits de précision et pouvant travailler jusqu'à 250 kHz ; il s'agit de convertisseurs à approximations successives offrant une latence de conversion de $4\mu s$ (temps total maximum nécessaire pour la conversion — $2,5\mu s$ — puis pour l'acquisition — $1,5\mu s$ —). Les conversions des 16 canaux sont déclenchées simultanément (i.e. pas de multiplexage).
- 16 convertisseurs N/A [Technology, 2000a] à 16 bits de précision. Il s'agit de convertisseurs parallèles, et le temps d'établissement du courant de sortie est de $1\mu s$ seulement, mais les canaux de conversion N/A de la carte TORO sont conçus pour ne travailler seulement que jusqu'à 250 kHz. La conversion des 16 canaux est simultanée.
- Un DSP pouvant travailler au format flottant 32 ou 64 bits (TMS320C6711 de Texas Instruments, 150 MHz / 900 Mflops).
- Mémoire interne de 32 Mo (SDRAM), accessible par la machine hôte depuis le port PCI.
- Entrée horloge pour synchronisation externe.
- Deux horloges/compteurs internes sur 32 bits.
- Un port I/O numérique sur 64 bits.
- La carte TORO se décline en deux versions : une version à 8 canaux de conversion A/N et N/A (TORO-8), et une version à 16 canaux de conversion A/N et N/A (TORO-16).

2.2 Organisation fonctionnelle

Le simulateur ERGON_X est une architecture de simulation complètement synchrone et se situe de ce point de vue dans la continuité des précédentes architectures de simulation en temps réel réalisées par le groupe de recherche ACROE & ICA. La conception de l'architecture ERGON_X se devait d'apporter une amélioration significative du critère de réactivité par rapport aux simulateurs précédents, et le verrou technologique principal se trouve dans les latences dues aux temps de transferts des données entre le TGR et le processus simulation. Nous avons vu dans le chapitre 3 (page 76) qu'il s'agissait d'une des principales limites de l'architecture TELLURIS.

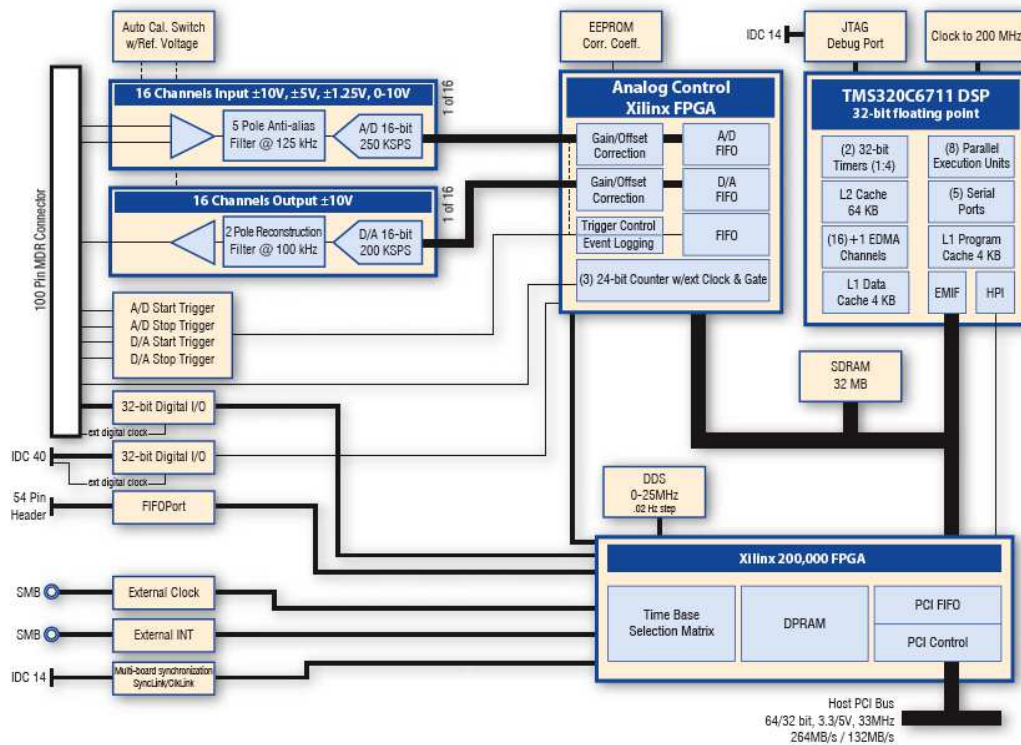


Figure 5.2. Schéma synoptique de la carte TORO

Un des apports du simulateur ERGON_X au regard des réalisations précédentes du laboratoire se trouve donc dans l'augmentation importante de la réactivité de la simulation. Pour cela, nous avons minimisé les temps de transferts de données entre le processus simulation et les entrées/sorties, en particulier du TGR. Le DSP C6711 de la carte TORO se trouve à proximité immédiate des FIFO des convertisseurs A/N et N/A. En faisant exécuter le processus simulation sur le DSP, nous sommes en mesure de bénéficier de temps de transfert extrêmement réduits entre le processus simulation et les convertisseurs.

L'exécution du processus simulation est cadencée soit par une des deux horloges internes à la carte TORO, soit par une source externe ; il est possible d'utiliser le même signal d'horloge pour communiquer le signal de synchronisation à d'autres éléments du simulateur.

Le processus superviseur n'est pas soumis à des contraintes d'exécution particulière, et est complètement désynchronisé du processus simulation. De plus, à la différence du processus simulation, le processus superviseur n'est pas non plus soumis à des contraintes de performance de calcul. Ce processus peut donc être exécuté sur un système d'exploitation conventionnel, non modifié pour le temps réel.

De la même manière que sur l'architecture TELLURIS, nous choisissons de désynchroniser le processus de visualisation du processus simulation (cf. la présentation de TELLURIS dans le chapitre 3, page 76). La seule contrainte reposant sur le processus de visualisation est alors de disposer de suffisamment de ressources de calcul pour pouvoir représenter les environnements

simulés. Cette contrainte repose presque exclusivement sur les performances de la carte graphique, et a contrario, peu sur les performances de la machine hôte.

Les trois composants d'une architecture de simulation : processus simulation, processus superviseur, et processus système sont alors répartis de la manière suivante (figure 5.3) :

- Processus superviseur : sur le processeur du PC hébergeant la carte TORO (PC hôte),
- Processus simulation : exécuté sur le DSP de la carte TORO,
- Seul le PC hôte nécessite un processus système, le DSP ne possédant pas de système d'exploitation.

L'utilisation du formalisme CORDIS a de particulier qu'il permet la synthèse de modèles multisensoriels à partir d'un seul et unique calcul de simulation ; nous avons en effet dit lors de la présentation de CORDIS-ANIMA (annexe D) qu'il était possible de faire de la synthèse de son en exportant le signal obtenu par les positions successives de un ou plusieurs modules simulés vers des haut-parleurs. Le processus de simulation étant exécuté par le DSP de la carte TORO, le fait de traiter les sorties sonores à partir de la machine hôte imposerait de pouvoir assurer un flux avec un débit de données relativement important entre le DSP et la machine hôte. En outre, si le traitement du signal sonore peut s'effectuer en boucle ouverte par rapport à la boucle haptique, nous sommes soumis à des contraintes importantes sur les délais temporels : la littérature montre que les délais de perception pour la boucle geste-son sont de l'ordre de 10 ms, voire de la milliseconde dans le cas de gestes experts. D'autre part, les convertisseurs N/A de la carte TORO ne comportent pas une précision conforme aux standards actuels pour l'audio de haute qualité (24 bits), mais permettent néanmoins de restituer un signal audio avec une qualité satisfaisante. En conclusion, nous choisissons de réserver deux canaux des convertisseurs N/A pour les sorties audio. En ce qui concerne la latence de la boucle geste-son, l'intérêt de cette solution est aussi de pouvoir travailler avec la même résolution temporelle que la fréquence de simulation, c'est-à-dire moins que 1 ms (la simulation de phénomènes audio demandant une fréquence de simulation supérieure à 10 kHz).

2.3 Composition d'un simulateur-type ERGON_X

En ce qui concerne le choix de la machine hôte, le seul critère important est la capacité de pouvoir héberger la carte TORO, qui est une carte PCI dite « full-size ». La liste des composants de notre simulateur est alors la suivante :

- PC hôte conventionnel (système d'exploitation Linux) et carte graphique
- Carte TORO, hébergée sur le port PCI de la machine hôte
- Transducteur gestuel ERGOS
- Haut-parleurs
- Écran

Durant la première phase de développement, nous avons donc utilisé un PC standard de faible puissance :

Processeur	Pentium III 1 GHz
Mémoire RAM	256 Mo
Carte graphique	nVidia NV20 Gforce3 (64 Mo RAM)

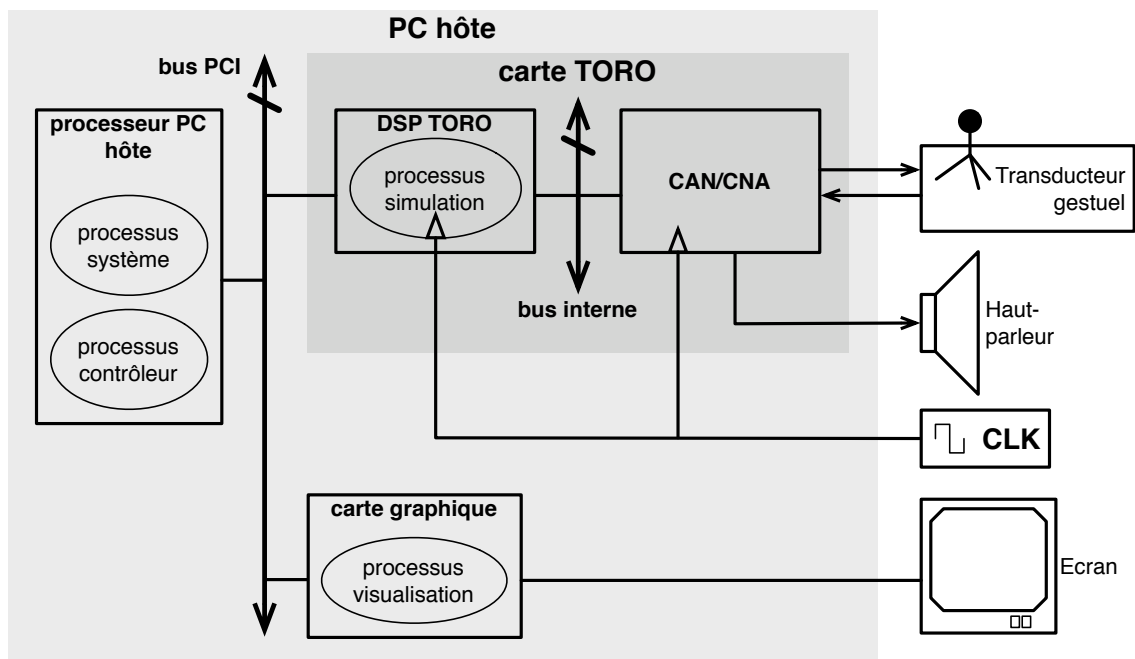


Figure 5.3. Architecture de la plateforme de simulation ERGON_X et répartition des différents processus sur les unités de calcul

Puis en 2007, pour la préparation des EES et leur présentation à la conférence Enactive/07, le laboratoire a acquis trois nouvelles machines afin de disposer de plusieurs plateformes de simulation. Les caractéristiques des nouvelles machines sont les suivantes :

Processeur	2,8 GHz (Intel Pentium e2140 Dual Core)
Mémoire RAM	1 Go
Carte graphique	ASUS 7200GS (carte bas de gamme)
Coût	moins de 400 € (fin 2007)

La carte TORO est aujourd'hui commercialisée aux environs de 6 000 €. Nous le voyons, c'est donc la carte TORO qui constitue le principal élément de puissance de cette configuration, puisqu'elle représente plus de 90 % du coût total de cette configuration.

3 Architecture logicielle

3.1 Pilote de périphérique — la bibliothèque `libtd`

La première partie du travail technique a consisté à prendre en main cette nouvelle carte d'acquisition. L'objectif était de développer un environnement de simulation sous Linux. La carte TORO n'étant pas supportée sous Linux, il nous a fallu dans un premier temps développer un pilote de périphérique minimal pour cette carte supportant les fonctionnalités suivantes :

- Reset : initialisation du DSP et des autres composants de la carte (mémoire et périphériques de communication)
- Configuration de la carte et initialisation

- Gestion complète du port HPI : accès en lecture et en écriture au bus d'adresses interne à la carte depuis la machine hôte
- Chargement des binaires pour le DSP
- Primitives de communication entre le DSP et le host via une mémoire partagée (SDRAM de la carte TORO)

Sous Linux, le port PCI de chaque périphérique est accessible depuis le mode utilisateur par mapping mémoire : au chargement de la machine hôte, une séquence de parcours des périphériques présents sur le port PCI alloue un espace mémoire réservé à chaque périphérique dans le système. Cet espace mémoire est vu comme un espace mémoire classique pour les programmes de l'espace utilisateur. Dans la première phase de développement, il nous fut ainsi possible de prendre en main la carte et de développer les fonctionnalités de base du pilote de périphérique sans être confrontés au problème supplémentaire du développement d'un module de noyau spécifique.

Le pilote de périphérique fut donc développé en mode utilisateur, et constitue simplement une bibliothèque logicielle, dénommée `libtd`, à intégrer au développement de l'application finale. En parallèle au développement de `libtd`, la structure du programme DSP fut progressivement affinée pour fournir un squelette de simulation performant.

La première phase de développement a permis d'atteindre un stade suffisamment avancé pour obtenir une station de simulation autonome, et les premiers modèles multisensoriels simulés sur la carte TORO furent développés directement sur la base de la bibliothèque `libtd`. Cependant, le développement de nouveaux modèles sur cette base logicielle demandait une connaissance poussée dans la structure du code, à la fois dans le code de l'application de contrôle sur la machine hôte, et dans le code du programme exécuté sur le DSP. La séparation entre le squelette de l'application de simulation et les portions de code utiles au calcul du modèle simulé n'était pas clairement définie.

3.2 Développement d'une API — `libErgonSim`

Il devenait nécessaire de développer une nouvelle architecture logicielle présentant une architecture plus modulaire, et permettant une distinction plus claire entre le code propre à la conception d'un modèle de simulation et le squelette de simulation.

Les objectifs étaient :

- Un meilleur cadre pour le développement du modèle utilisateur : développement plus facile et plus sécurisé,
- Une approche plus modulaire des différents composants du programme de simulation,
- Une interface plus souple pour des programmes de plus haut niveau (permettant par exemple l'implémentation des EES).

Durant l'année 2007, il était aussi prévu de développer plusieurs scénarios utilisateurs autonomes pour le réseau Enactive (les *Enactive Emblematic Scenarii*). Dans ce cadre, il était nécessaire de pouvoir contrôler la simulation par une application de plus haut niveau en charge de la présentation d'un parcours didactique pour l'utilisateur/visiteur. La bibliothèque développée devait permettre l'intégration d'une application de simulation dans un tel scénario utilisateur. Face à ces objectifs, il devenait intéressant de migrer le code existant du C vers le C++.

La bibliothèque `libErgonSim` offre un mécanisme de classes *abstraites*. Ces classes, fournies par la bibliothèque, constituent le squelette du programme de simulation, et offrent un cadre de développement pour l'application visée. Lors du développement d'un programme de simulation donné, le concepteur de la simulation construit alors son programme en créant des classes dérivées issues de ces classes abstraites. Seules les procédures (méthodes) propres au fonctionnement du modèle nécessitent alors d'être définies dans le code utilisateur.

3.3 Exécution d'un pas de simulation et chaîne de transmission des signaux

Décrivons la chaîne de signal mise en œuvre par le processus simulation, depuis les entrées/sorties du TGR, jusqu'au processus simulation (figure 5.4) :

- Pour chaque degré de liberté utilisé sur le moteur ERGOS, les signaux de position fournis par l'électronique ERGOS sont calibrés sur une échelle analogique de ± 10 V.
- Au coup d'horloge n , une conversion A/N est déclenchée, produisant des données de position numériques au format entier signé sur 16 bits,
- Les données de position sont converties au format flottant 32 bits : l'échelle des valeurs de positions est alors $[-1 ; 1 - \frac{1}{2^{15}}]$,
- Un gain de position global (α) est appliqué. Il permet la mise à l'échelle entre les grandeurs du monde physique réel, déterminant l'amplitude des signaux de position en entrée des convertisseurs A/N, et les grandeurs du monde simulé. De même, les gains de force δ/γ permettent d'adapter les grandeurs des signaux de force du monde simulé aux grandeurs des signaux de force à appliquer au système ERGOS.
- Puis la transformation cinématique des positions des tranches : l'échelle de position est alors $[-\beta ; +\beta]$, et les données sont de dimension n , suivant le transformateur cinématique utilisé.
- Nous avons vu dans le chapitre présentant le formalisme CORDIS, que, le TGR fonctionnant en mode impédance peut-être assimilé, depuis le modèle, à un module de type <MAT>. Les données de position n -dimensionnelles obtenues après l'étape précédente sont donc appliquées à des modules de type <LIA>, qui retournent des signaux de force n -dimensionnels.
- La chaîne de traitement inverse à celle parcourue jusqu'à maintenant est alors appliquée : transformation cinématique inverse pour les signaux de force, application d'un gain global, puis conversion au format entier signé sur 16 bits.
- Au coup d'horloge $n + 1$, les signaux analogiques de position sont convertis en signaux analogiques.

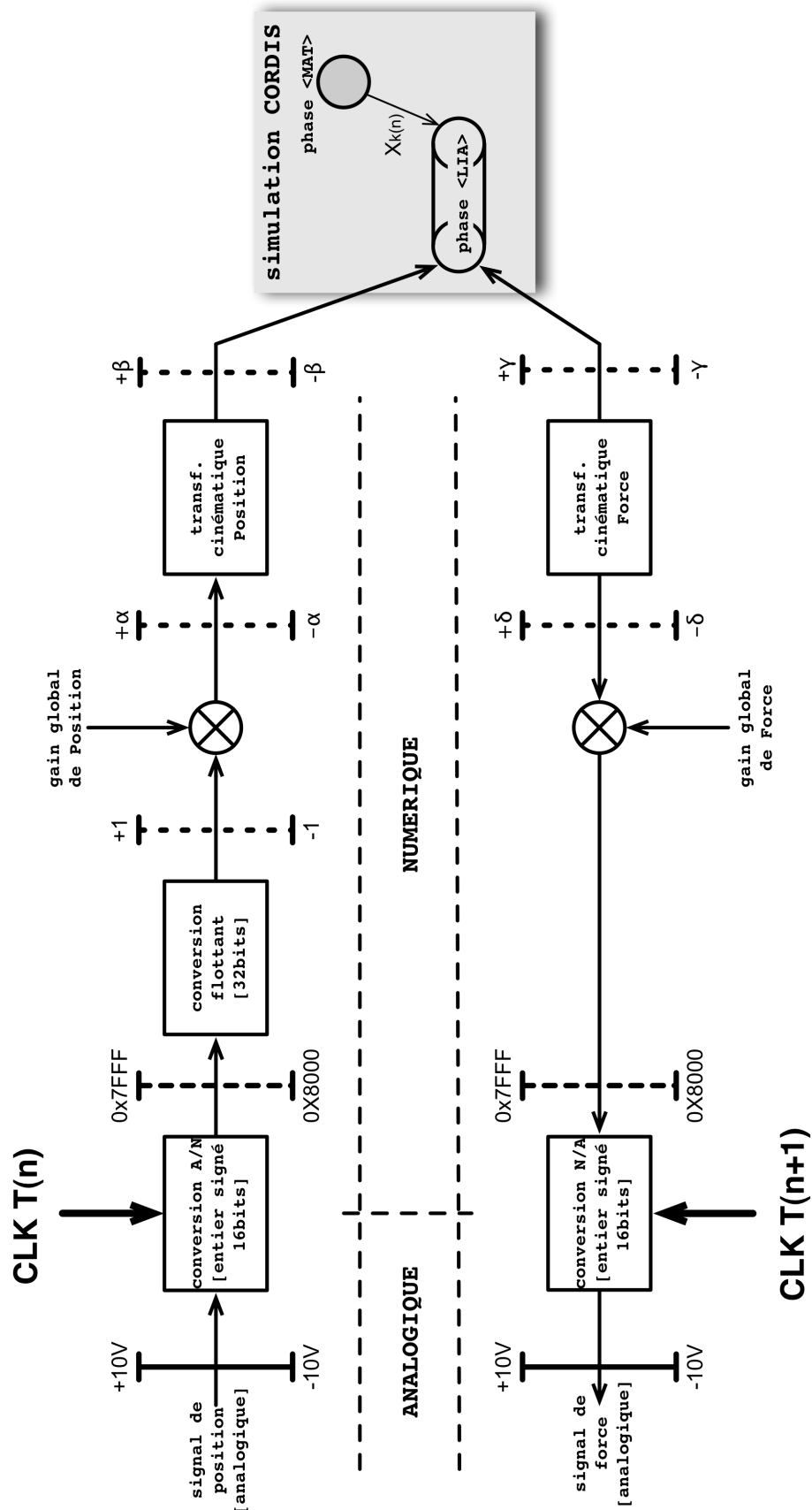


Figure 5.4. La chaîne des signaux mise en œuvre entre les entrées/sorties du TGR ERGOSet la simulation du modèle CORDIS

3.4 Communication hôte-DSP

C'est le PC hôte qui contrôle le chargement du programme sur le DSP du HPU, l'initialisation de la carte, le lancement du programme DSP. Une fois la simulation en cours, il est nécessaire que le processus simulation reste sous le contrôle du processus superviseur résidant sur le PC hôte. Cependant, si l'ensemble de la chaîne de simulation (conversion, transmission des données au simulateur, calcul d'un pas de simulation, transmission des données en sens inverse) est exécuté de manière complètement synchrone, la communication entre le DSP et le PC hôte peut être complètement asynchrone du processus simulation.

Afin de pouvoir faire communiquer le processus simulation et le processus superviseur, et plus généralement le programme exécuté sur le DSP du HPU et le processeur du PC hôte, nous avons besoin d'outils pour l'échange de données, et d'outils pour pouvoir synchroniser l'exécution des deux fils d'exécution sur le HPU et sur le PC hôte. Nous nous trouvons face à une architecture partagée, où les deux processeurs du PC hôte et du HPU se trouvent hébergés sur les architectures distinctes. Il ne nous est donc pas possible d'utiliser les mécanismes de synchronisation et de communication fournis par un système d'exploitation classique (rendez-vous, verrous, signaux, etc.). Le PC hôte étant cependant capable d'accéder en lecture comme en écriture à l'intégralité du bus interne du HPU, nous avons choisi d'implémenter un mécanisme de communication entre le PC hôte et le DSP par lecture/écriture dans une mémoire partagée. Cette mémoire partagée se trouve dans la mémoire interne du DSP, et est configurée pour que le DSP accède à cette zone mémoire sans passer par la mémoire cache. Le host peut accéder également à cette zone mémoire en lecture et en écriture par le port HPI. La synchronisation entre le processus simulation et le processus superviseur se fera alors par l'échange de drapeaux, et des phases d'attente active.

La carte TORO dispose de mémoires :

- Une SDRAM de 32 Mo (plage d'adresses 0xA000 0000 à 0xA1FF FFFF sur le bus DSP). L'accès à cette mémoire s'effectue au travers de la mémoire cache.
- Une SRAM de 64 ko (plage d'adresses 0x0000 à 0xFFFF). Il s'agit de la mémoire cache de niveau L2 du DSP, mais cette mémoire est configurable tout ou en partie soit en tant que mémoire cache de niveau L2 (le DSP possédant des mémoires caches plus petites de niveau L1 spécifiques pour le code et les données), soit en tant que zone de stockage de programme ou de données. L'accès à cette mémoire n'est pas caché. Cette mémoire se trouve accessible sur le bus d'adresse du DSP de la même manière que la SDRAM ; on peut également faire des accès à cette mémoire depuis le PC hôte via le port HPI.

Les cycles de lecture et écriture en SDRAM étant beaucoup plus lents que en SRAM, nous stockons sur la SDRAM les portions de code qui ne concernent pas la boucle temps-réel (initialisation et configuration de la carte TORO, initialisation des modules CORDIS, etc.), ainsi que les données de simulation (typiquement, paramètres algorithmiques et vecteurs d'état). Des tests ont en effet montré que lorsque la taille des données de simulation n'est pas trop importante, l'utilisation de cache compensait les faibles performances de la SDRAM. Ce même effet de cache ne peut pas compenser avec les faibles performances de la SDRAM pour les sections de code critiques, telles que le calcul de la boucle de simulation. Ces portions de code doivent donc être placées en SRAM, qui est beaucoup plus rapide.

L'espace mémoire SRAM est relativement limité, et ne permet pas d'y faire tenir les données de simulation (vecteurs d'état, paramètres algorithmiques, variables temporaires telles que les

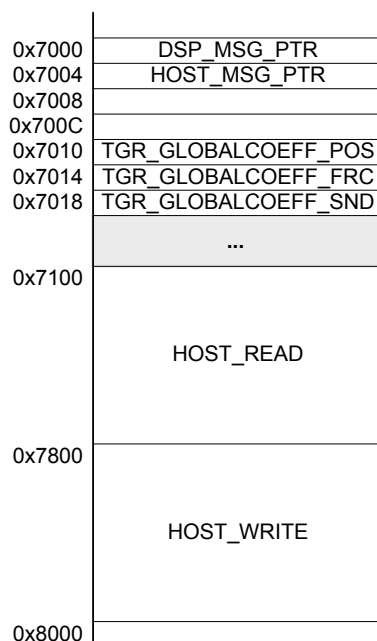


Figure 5.5. Zone mémoire de communication entre le PC hôte et le DSP

positions retardées, etc.). Par contre, son accès est rapide, puisqu'il s'agit d'une mémoire non cachée, et interne au DSP. Nous y plaçons donc les portions de code critiques, telles que la boucle de simulation en temps-réel afin de garantir les meilleures performances. La totalité de cette mémoire est répartie entre vecteurs d'interruptions [0x0000 ; 0x01FF], code pour la simulation temps-réel [0x0200 ; 0x6FFF] et mémoire cache [0x8000 ; 0x8FFF].

Une plage d'adresses spécifiques [0x7000 ; 0x7FFF] est réservée pour les échanges de données entre le processus simulation sur le DSP et le processus superviseur sur la machine hôte (Figure 5.5). Nous y réservons de manière permanente 4 octets pour l'envoi de messages du DSP vers le PC hôte (DSP_MSG_PTR — 0x7000) et du PC hôte vers le DSP (HOST_MSG_PTR — 0x7004). Sont également alloués de manière permanente les trois gains d'amplification globaux des données de position (TGR_GLOBALCOEFF_POS — 0x7010), de force (TGR_GLOBALCOEFF_FRC — 0x7014), et des sorties son (TGR_GLOBALCOEFF_SND — 0x7018). Enfin, deux espaces mémoires plus importants sont réservés pour l'envoi des paramètres algorithmiques du PC hôte vers le DSP [0x7100 ; 0x7FFF] et l'envoi des vecteurs d'état du DSP vers le PC hôte pour la sortie graphique [0x7800 ; 0x7FFF]. Ces deux espaces mémoires ne peuvent pas être structurés a priori, puisque leur contenu dépendra du modèle simulé (c'est-à-dire, du type et du nombre de modules CORDIS simulés).

3.5 Synchronisation du calcul de la simulation

Le programme exécuté sur le DSP comporte deux fils d'exécution indépendants.

A l'initialisation du DSP, le programme principal se charge de la configuration des convertisseurs A/N et N/A et de l'horloge, puis se met en attente de l'ordre de démarrage de la simulation par la machine hôte. Une fois que la simulation est en cours, le programme principal exécute en boucle

le rafraîchissement des paramètres algorithmiques envoyés par le processus superviseur, et la copie de l'état courant du modèle pour le processus de visualisation.

Le déclenchement du calcul d'un pas de simulation est asservi sur le signal d'horloge : le signal d'horloge, généré par un timer interne à la carte TORO ou obtenu depuis l'entrée horloge externe, est assigné au déclenchement des conversions A/N. Dès que le FIFO des convertisseurs A/N a atteint un niveau de remplissage égal au nombre de canaux de conversion utilisés, une interruption est levée, déclenchant le branchement du DSP sur le début du code de la fonction `II_EINT7_handler()`. Cette fonction contient le code pour le calcul d'un pas de simulation. En conséquence, ce fil d'exécution est indépendant du fil d'exécution du programme principal. A la fin de l'exécution de la fonction `II_EINT7_handler()`, le DSP branche à nouveau sur le point d'exécution dans le programme principal où il se trouvait auparavant.

3.6 Le calcul d'un pas de simulation

La séquence de calcul d'un pas de simulation se déroule comme il suit :

```
II_EINT7_IntHandler() {
    • Lecture du FIFO A/N, copie dans le tableau short_buff
    • Conversion en flottant 32 bits [float_buff]
    • Application du gain en position global
    • Transformation cinématique de position
    • Calcul des modules CORDIS

    • Transformation cinématique de force
    • Application du gain de force global
    • Conversion au format entier signé sur 16 bits [short_buff]
    • Écriture dans le FIFO N/A
}
```

Figure 5.6. Déroulement d'un pas de simulation

3.7 Automate d'états du simulateur

La Figure 5.7 décrit l'automate d'état du programme principal exécuté sur le DSP. Comme nous l'avons décrit plus haut, le calcul d'un pas de simulation est relativement indépendant de cet automate d'état, et n'apparaît donc pas sur la figure.

Le diagramme d'états du DSP est complètement esclave du processus superviseur exécuté sur l'application hôte. Chaque changement d'état comprend deux phases : la demande de changement d'état par le processus superviseur, et la confirmation du changement d'état par le processus simulation. Par exemple, une fois que le processus simulation se trouve dans l'état `READY`, le processus simulation se met en attente de la commande de lancement de la simulation ou de la commande de terminaison du programme de la part du processus superviseur. Le processus simulation confirme ensuite au processus superviseur que le changement d'état est bien effectif.

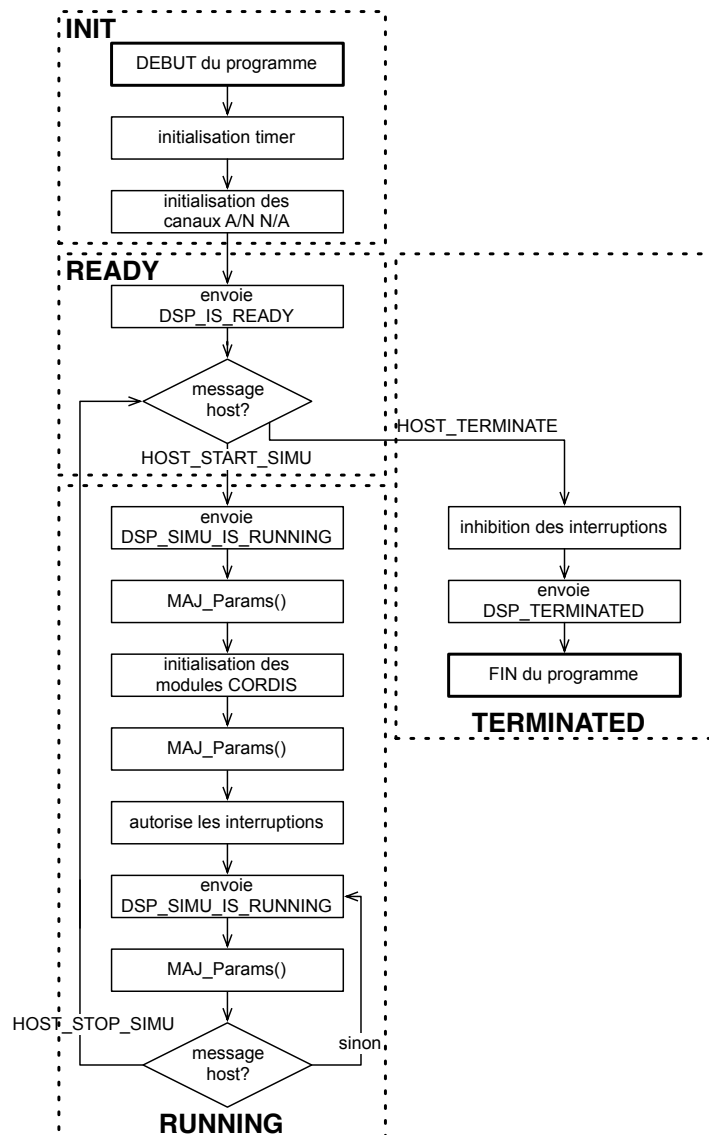


Figure 5.7. Automate d'états du programme principal exécuté sur le DSP

4 Conclusion

Ce chapitre présente l'architecture matérielle et logicielle du simulateur ERGON_X, dont le développement fut réalisé au cours de cette thèse. L'architecture du simulateur repose sur les trois composants matériels suivants : la gamme de TGR ERGOS, la carte d'acquisition et de génération TORO de Innovative Integration, et un ordinateur grand public bas de gamme.

La carte TORO est le cœur de cette architecture et constitue la mise en œuvre du composant que nous avons appelé HPU (chapitre 4). Nous avons fait le choix de tirer pleinement parti de la puissance du HPU, en exécutant le processus simulation sur le DSP de la carte TORO. Le processus simulation, exécuté sur le HPU est ainsi isolé de l'ordinateur hôte des contraintes de l'exécution de ce processus en temps réel. Un PC standard de bas de gamme est ainsi suffisant pour le simulateur ERGON_X, les éléments de puissance étant concentrés dans le HPU. D'autre part, la carte TORO fut choisie en grande partie pour la qualité de ses convertisseurs, qui présentent

des temps de conversion inférieurs à $5\ \mu\text{s}$ pour l'acquisition comme pour la génération. Le DSP se trouvant à proximité immédiate des FIFOs des convertisseurs, nous avons pu réduire les temps de communication entre le processus simulation et le TGR, ce qui nous a permis d'obtenir une boucle fermée synchrone rapide et de porter la fréquence de simulation jusqu'à 44,1 kHz.

Afin d'utiliser la carte TORO de Innovative Integration comme un HPU sur le système d'exploitation GNU/Linux, nous avons développé le pilote de périphérique de cette carte pour GNU/Linux. Ce pilote possède les fonctionnalités de base permettant de contrôler les convertisseurs numérique-analogique et d'exécuter du code sur le DSP de la carte. En raison de contraintes temporelles et de documentation technique insuffisante, le support du DMA entre l'ordinateur hôte et la carte TORO n'a pas encore été mis en œuvre. Cette étape de développement est cependant nécessaire pour que le HPU présente une interface simulateur qui permette l'export de tout ou partie du processus simulation à l'extérieur du HPU. En effet, le support du DMA permettra d'obtenir des transferts de données entre le DSP de la carte TORO et le ou les processeurs de l'ordinateur hôte avec une grande bande passante et des temps de transferts réduits.

Le travail logiciel pour l'utilisation de la carte TORO comme HPU a été envisagé selon deux axes. Le premier est le développement du simulateur autonome ERGON_X. Le second est l'utilisation de ce HPU dans d'autres architectures de simulation développées par le laboratoire. Ainsi, le HPU que nous avons développé est intégré dans la réalisation de la prochaine version du simulateur TELLURIS (thèse de J. Castet, en cours). Dans le cadre du simulateur ERGON_X, le HPU fut utilisé pour contrôler simultanément un TGR de la gamme ERGOS et le stimulateur tactile Tactos [Merlin, 2007]. Le Tactos est un stimulateur tactile composé de quatre matrices Braille, chacune étant composée de 8 points programmables. Le Tactos se contrôle par l'intermédiaire d'un port parallèle DC25. Dans le cadre du simulateur ERGON_X, nous utilisons le port I/O numérique de la carte TORO — i.e. sur l'interface geste du HPU — pour contrôler le Tactos. La boucle de contrôle du Tactos est intégrée à la boucle de simulation synchrone. Ce travail technologique a donné lieu à des expériences psychophysiques sur l'intégration du retour d'effort et de la stimulation tactile.

Chapitre 6

Tâches et Scenarii Emblématiques pour l'Interaction Instrumentale

Sommaire du Chapitre

1	Introduction	120
2	Le « E » : un démonstrateur pour ERGON_X et ERGOS	121
2.1	Un modèle-type pour la simulation haptique	121
2.2	Description du modèle	122
2.3	Transformation cinématique	124
2.4	Choix de la fréquence de simulation : 40 kHz	124
2.5	Raideur simulée	125
3	Simulation multisensorielle pour la psychophysique	125
3.1	Conditions nécessaires pour une démarche expérimentale	125
3.2	Perception de matériaux déformables	127
3.2.1	Objectifs	127
3.2.2	Description de la scène	127
3.2.3	Méthode et protocole expérimental	130
3.2.4	Résultats	131
3.3	Le tapping	133
3.3.1	Objectifs	133
3.3.2	Modèle	134
3.3.3	Résultats	135
4	Scénarios Emblématiques de l'Enaction	136
4.1	Le Scénario Emblématique « Pebble Box »	136
4.1.1	La PebbleBox de Sile O'Modhrain et Georg Essl, MediaLab Europe (PB1)	136
4.1.2	La PebbleBox de Charlotte Magnusson – ULUND (PB2)	137
4.1.3	La PebbleBox ACROE (PB3)	138
4.1.4	Le Scénario	140
4.1.5	Conclusion	143
4.2	Le Scénario Emblématique « Ergotic Sounds »	144
4.2.1	Historique du modèle	144

4.2.2	Description du modèle – version 2001	144
4.2.3	Description du modèle – version 2007	146
4.2.4	Intérêt de la scène et résultats	147
4.2.5	Le Scenario	148
5	Conclusion	152

1 Introduction

Nous avons proposé dans le chapitre 3 une analyse des dispositifs technologiques qui montre d'une part la difficulté des systèmes actuels à rendre compte des propriétés essentielles de l'énaction et de la situation instrumentale et au delà de l'énaction. Elle nous a donné un cadre conceptuel pour de nouveaux développements technologiques, qui ont été présentés dans les chapitres suivants. Dans ce chapitre, nous décrivons un ensemble d'expériences visant à mettre à la question nos hypothèses et technologies dérivées, tant du point de vue des exigences techniques que des propriétés que nous avons souhaitées qu'ils capables de restituer. Il s'agit bien entendu des propriétés majeures de la relation instrumentale et de la relation enactive telles qu'elles ont été exposées au chapitre 1 : émergence, présence, organicité de l'interaction, apprentissage, adaptivité, etc...

Le premier axe est l'exploitation des performances de la technologie ERGOS. Dans la précédente architecture de simulation TELLURIS, la fréquence de communication (synchrone) avec le TGR était limitée à 3 kHz. La nouvelle architecture de simulation ERGON_X permet de calculer des boucles de simulation synchrones à des fréquences élevées (jusqu'à 50 kHz). Le fait de pouvoir d'ouvrir le verrou technologique du « 3 kHz » a permis d'ouvrir un nouveau domaine de la modélisation multisensorielle. Cette nouvelle architecture de simulation permet d'exploiter pleinement les performances dynamiques de la dernière version du moteur en tranches ERGOS (2001). Nous avons conçu dans le cadre de cette thèse le modèle du « E » comme un modèle de démonstration pour la gamme de produits ERGOS. Il s'agit d'un modèle de test permettant à un utilisateur de tester les caractéristiques principales d'un transducteur gestuel : mouvement libre, rigidité maximale, contours précis... Ce modèle fut présenté pour la première fois à la conférence EuroHaptics 2006 à Paris dans le salon des exposants industriels.

Le deuxième axe de travail est celui de l'expérimentation psychophysique conventionnelle. La psychophysique exploite de plus en plus les technologies de la réalité virtuelle et les nouvelles technologies de médiation homme-ordinateur, puisque ces dernières présentent de nombreux avantages pour la construction de situations expérimentales. Nous montrons cependant qu'un outil pour l'expérimentation psychophysique doit posséder certaines qualités essentielles, et nous montrons comment ces qualités ont été utilisées dans des expériences psychophysiques. Dans le cadre de cette thèse, deux expériences ont été menées : (1) la première sur le geste de percussion d'un objet sonore, (2) la seconde sur la perception de l'élasticité d'objets compressibles, en collaboration avec Gunnar Jansson, professeur émérite au département de psychologie de l'université de Uppsala, Suède.

Le troisième axe de travail fut l'un des plus importants en ce qui concerne les usages du simulateur ERGON_X. Dans le cadre du réseau d'excellence Enactive Interfaces, le laboratoire a réalisé

cinq EES (*Enactive Emblematic Scenarii*) : « *Ergotic Sounds* » (frottement d'archet), « *Pebble Box* » (la boîte de cailloux), « *RVPC* » (Coopération Physique entre le Réel et le Virtuel), « *Nanomanipulation* » et « *Shapes and Contour* » (les Formes et le Contour). Ces EES, réalisés sous la direction de A. Luciani dans le cadre du projet européen Enactive Interfaces, ont été conçus pour permettre à un visiteur autonome (i.e. sans l'aide d'une assistance technique) de parcourir un scénario interactif l'amenant à explorer une situation instrumentale fortement ancrée dans le concept de l'enaction. Le travail mené dans le cadre de cette thèse fut essentiellement (1) un travail de développement de nouveaux modèles, ou de portage sur le simulateur ERGON_X des modèles développés antérieurement pour d'autres simulateurs (e.g. modèle de frottement d'archet) ; (2) un travail de développement logiciel pour permettre le fonctionnement du simulateur en autonomie complète afin que les scénarios puissent être présentés au grand public sans la présence permanente d'une assistance humaine ; (3) la participation au travail expérimental et à la validation des résultats. En particulier, les scenarii « *PebbleBox* » et « *Ergotic Sounds* » ont fait l'objet d'un travail expérimental plus approfondi auquel nous avons participé et que nous présentons également dans ce chapitre. La réalisation du scénario « *Pebble Box* » fut réalisée sous la direction de Annie Luciani en collaboration avec Charlotte Magnusson, du laboratoire CERTEC de l'université de Lund, Suède, et Sile O'Modhrain, de Queens University à Belfast, Irlande. Les cinq « EES » ont été présentés durant l'exposition pour le grand public « Toucher le Futur », organisée lors de la 4^e conférence scientifique Enactive/07 par le groupe de recherche ACROE & ICA à Grenoble en Novembre 2007, puis lors de la 4^e évaluation du réseau Enactive Interfaces, qui a eu lieu à Pise en mars 2008. Nous présentons dans ce mémoire la réalisation des deux scénarios « *Pebble Box* » et « *Ergotic Sounds* », pour lesquels notre investissement fut plus important. Dans le cadre de ce travail, la participation à la réalisation des scénarios « *RVPC* », « *Nanomanipulation* » et « *Shapes and Contour* » d'encadrement et d'assistance à la modélisation et à l'implantation des scénarios.

2 Le « E » : un démonstrateur pour ERGON_X et ERGOS

2.1 Un modèle-type pour la simulation haptique

Dans le chapitre présentant l'analyse fonctionnelle des systèmes haptiques, nous avons vu que deux des propriétés fondamentales d'un système haptique sont (1) sa capacité à restituer un mouvement libre avec le moins de friction possible et une inertie résiduelle faible, et (2) sa capacité à restituer des contacts durs [Couroussé, 2007b]. La difficulté de la conception d'un système haptique réside en partie dans la difficulté à concilier dans un même système ces deux propriétés orthogonales. Il existe un modèle de démonstration simple permettant d'illustrer ces deux propriétés fondamentales, communément appelé *corner-in-the-box* (« le coin dans la boîte »). L'utilisateur manipule par le biais du système haptique un cube rigide, qu'il peut déplacer dans une boîte tridimensionnelle (figure 6.2). Il est ainsi possible d'expérimenter la qualité de restitution du mouvement libre, la qualité de restitution des contacts contre une, deux ou trois surfaces, la qualité des mouvements de friction au contact d'une surface. Dans le cas d'un système à 6 degrés de liberté, il est aussi possible d'expérimenter la qualité de mouvement en rotation.



Figure 6.1. Démonstration du modèle lors de l'exposition grand public « Toucher le Futur » de la conférence Enactive/07 (Novembre 2007)

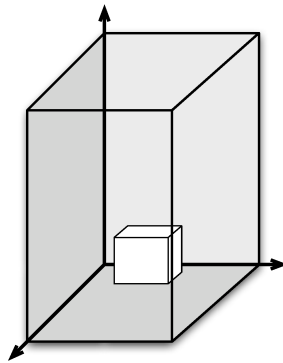


Figure 6.2. Représentation schématique d'un modèle conventionnel « corner-in-the-box ». L'utilisateur déplace le solide blanc dans une boîte tridimensionnelle, et peut expérimenter la qualité de restitution du mouvement libre et des contacts.

2.2 Description du modèle

Nous avons choisi d'implanter un modèle de type « corner-in-the-box » pour démontrer les excellentes propriétés dynamiques du moteur en tranches ERGOS. Il s'agit d'un modèle bidimensionnel permettant la restitution de coins convexes et concaves : l'utilisateur déplace le point manipulé dans un espace vide de la forme d'un « E ». La forme géométrique de la lettre « E » fut choisie parce qu'elle contient à la fois des surfaces lisses et des coins concaves pour l'expérimentation de la restitution de contacts, et des coins convexes pour la restitution de détails de surfaces précis.

Le système ERGOS fonctionne en mode impédance, et le point manipulé est donc assimilé à un module CORDIS de type <MAT>. Les surfaces du « E » étant fixes dans l'espace de modélisation, il est possible de modéliser l'interaction entre le point manipulé et les surfaces du « E » par une simple connexion entre un module SOL et la métaphore <MAT> du système haptique par des modules <LIA> spécialisés. Nous avons conçu un module <LIA> qui permet de représenter l'interaction avec une forme géométrique rectangulaire convexe (RECT convexe), et un module <LIA> l'interaction avec une forme rectangulaire concave (RECT concave). Nous choisissons

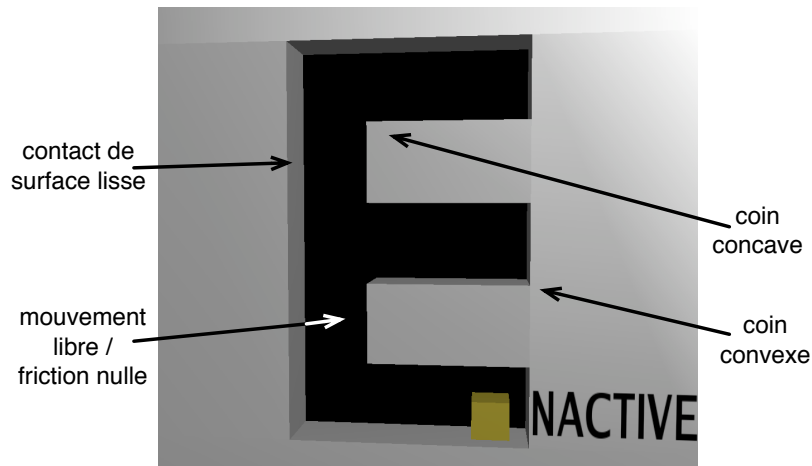


Figure 6.3. Capture d'écran de la simulation du « E » ; le cube jaune en bas à droite de la grille du « E » représente le point manipulé par l'utilisateur.

alors de représenter la forme du « E » par la combinaison d'un module <LIA> concave et de deux modules <LIA> convexes. La topologie du modèle CORDIS est alors la suivante : le point manipulé (<MAT>) est connecté à un module SOL par l'intermédiaire de trois modules <LIA> spécialisés : deux modules convexes et un module concave (figure 6.4).

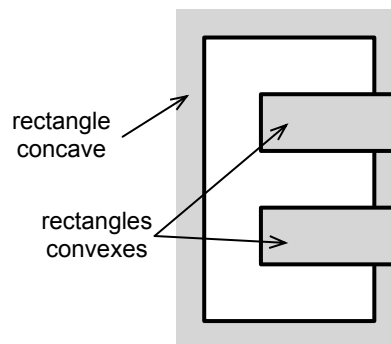


Figure 6.4. Modélisation de la forme géométrique du « E » à partir de deux formes rectangulaires élémentaires convexes et concaves

Le module RECT convexe établit une relation de visco-élasticité entre le point manipulé et le module SOL lorsque le point manipulé se trouve à l'extérieur de la forme rectangulaire. Le module RECT concave est le pendant du module convexe : il établit une relation de viscoélasticité entre le point manipulé et le module SOL lorsque le point manipulé se trouve à l'intérieur de la forme rectangulaire. Ces deux modules sont des modules 2D. Notons que ces modules entretiennent une grande similitude avec les modules BUT et BUL 2D (figure 6.5). La seule différence est que la projection de la force de contact est axiale suivant le vecteur formé par la position des deux modules <MAT> connectés aux <LIA> BUT ou BUL, alors que dans le cas des modules RECT, la force est projetée indépendamment sur chaque dimension du vecteur de position. Il est alors nécessaire de définir des paramètres de seuil indépendants pour chaque dimension d'un module RECT (S_x et S_y sur la figure 6.5).

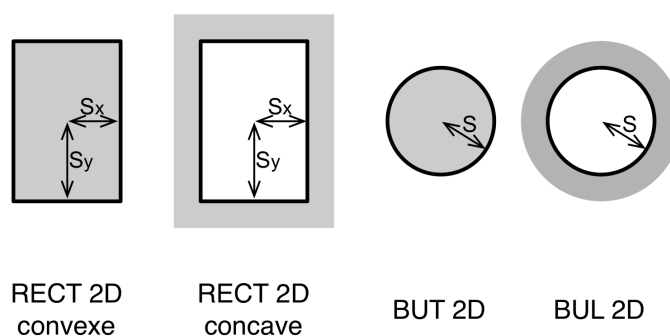


Figure 6.5. Comparaison entre les <LIA> RECT2D et les <LIA> BUT et BUL

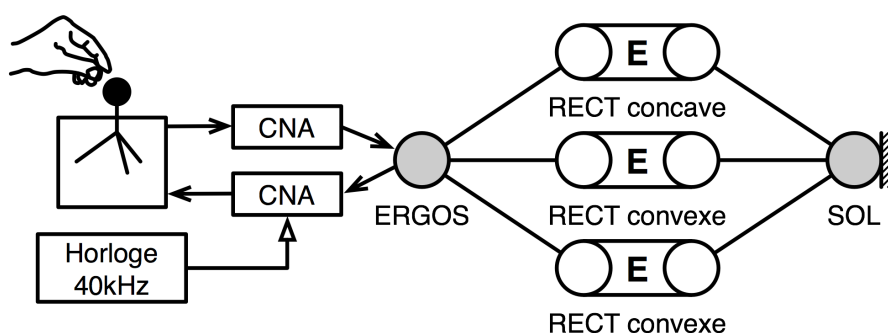


Figure 6.6. Topologie du modèle CORDIS « E »

2.3 Transformation cinématique

La manipulation se fait à l'aide d'un stick 3D ERGOS, et l'utilisation d'un transformateur cinématique dit « guidé plan » permet d'adapter l'espace de mouvement 3D du stick à l'espace de modélisation 2D. Les déplacements dans le modèle se font dans un plan parallèle au sol (X, Y), et l'axe vertical (Z) est bloqué par une liaison visco-élastique.

2.4 Choix de la fréquence de simulation : 40 kHz

Les positions initiales des trois modules SOL, ainsi que les paramètres de seuil des trois modules RECT sont choisis de manière à représenter un espace de manipulation en forme de « E ».

La fréquence de simulation est fixée à 40 kHz. Le choix d'une telle fréquence de simulation se justifie pour plusieurs raisons. En effet, la stabilité numérique des modèles CORDIS simulés est fonction de la fréquence de simulation. Augmenter la fréquence de simulation au-delà des 1 kHz traditionnels permet d'améliorer la stabilité numérique du modèle, et permet ainsi l'utilisation d'une plus grande plage de paramètres algorithmiques. Il est alors possible de simuler des raideurs de contact beaucoup plus importantes. Aussi, l'utilisation d'une fréquence de simulation importante permet [Campion et Hayward, 2005] d'améliorer la qualité du rebond lors de la simulation d'un mur [Chanclou, 1996]. Ceci est facilement vérifiable en évaluant des systèmes haptiques dont la fréquence de simulation est relativement faible. Par exemple, le SENSO-Wheel

est un volant à retour d'effort développé par DLR [Jean-Loup Florens, 2004]. Il s'agit d'un système à un seul degré de liberté en rotation, commandé en impédance. Le système présente d'excellentes qualités dynamiques et le rendu de la simulation d'une butée est excellent lorsque la vitesse de contact est relativement faible. Cependant, dès que la vitesse de contact est plus importante, la butée est perçue comme plus « molle ».

En outre, la bande passante des actionneurs du système TGR ERGOS est estimée supérieure à 10 kHz.

2.5 Raideur simulée

Après étalonnage de l'électronique ERGOS, et compte tenu des différents gains logiciels et matériels, la raideur des butées simulées dans ce modèle est de l'ordre de 135 N.mm^{-1} .

Cette mesure classe le système ERGOS parmi les systèmes présentant les meilleures performances à l'heure actuelle en termes d'intensité des forces générées par le système d'actionneurs. En effet, le modèle Phantom Premium HF (pour *High Force*), le plus puissant de la gamme de produits de Sensable, présente une raideur maximale de $3,5 \text{ N.mm}^{-1}$ [sen, 2007]. Le système Delta de ForceDimension présente une raideur maximale de $14,5 \text{ N.mm}^{-1}$ [Force Dimension, 2008] et le Haptic Master de Moog FCS, reconnu actuellement comme un des systèmes commercialisés présentant la meilleure rigidité, permet une rigidité maximale de 50 N.mm^{-1} [fcs, 2008]. Ce système, à l'inverse des systèmes cités précédemment et du système ERGOS, est contrôlé en admittance, ce qui favorise l'obtention de contacts rigides. Dans la catégorie des systèmes robotiques hybrides permettant une utilisation haptique, citons le cobot développé par Faulring et Colgate, qui permet la simulation de contacts rigides jusqu'à 100 N.mm^{-1} [Faulring *et al.*, 2006].

En outre, nous pouvons observer que, d'un point de vue subjectif, les contacts rigides obtenus avec ERGON_X sont supérieurs à ceux obtenus par exemple avec le Haptic Master, en raison d'une inertie résiduelle faible, alors que le Haptic Master présente une inertie résiduelle de 3 kg. En effet, Lawrence et al. [Lawrence *et al.*, 2000] ont montré que la perception de rigidité est liée au ratio entre la variation de force et la variation de la vitesse au moment de la pénétration plutôt qu'à la mesure de la rigidité simulée. Ce ratio est appelé *rate-hardness*. A rigidité équivalente, le *rate-hardness* sera meilleur si l'inertie résiduelle du système haptique est plus faible.

3 Simulation multisensorielle pour la psychophysique

3.1 Conditions nécessaires pour une démarche expérimentale

La démarche expérimentale en psychophysique est centrée sur l'humain en tant qu'objet d'étude. Comme toutes les sciences expérimentales, elle consiste en l'observation du système d'étude (ici, le *sujet* humain) soumis à une excitation (le *stimulus*). S'il existe une corrélation forte entre l'observation objective des modifications comportementales ou des retours subjectifs du sujet et le stimulus, alors il est possible de généraliser les résultats de l'étude dans un modèle. Pour que cette observation soit pertinente, il est nécessaire de pouvoir modéliser le stimulus le plus simplement possible, c'est-à-dire de réduire l'expression du stimulus à une ou plusieurs *variables indépendantes*.

Une des difficultés de l'expérimentation psychophysique réside alors dans le fait d'isoler une grandeur physique et de pouvoir la rendre indépendante des autres variables physiques de la situation expérimentale. De ce point de vue, il est aisé de comprendre le fort intérêt que l'on rencontre actuellement pour l'utilisation des systèmes haptiques et des plateformes de réalité virtuelle pour la psychologie expérimentale. En effet, ces outils permettent de disposer pour l'expérimentation d'un matériau expérimental modifiable et configurable à volonté beaucoup plus simplement qu'un matériau équivalent dans le monde réel. Il devient en effet possible de créer une palette presque infinie de nouvelles situations expérimentales à l'aide simplement d'un ordinateur et de périphériques adaptés. Puisque le matériau expérimental devient plus simple à modéliser, la construction de conditions expérimentales adéquates permettant en particulier l'extraction d'une ou plusieurs variables indépendantes devient a priori plus simple grâce à ces technologies.

Dans l'objectif de l'expérimentation psychophysique, la plateforme de simulation doit présenter deux caractéristiques fondamentales.

La première est, nous l'avons dit, le contrôle du stimulus, et plus particulièrement de la grandeur physique qui servira de variable indépendante dans l'expérience. L'intérêt de la simulation prend alors tout son sens, puisqu'il est possible de paramétrer, d'ajuster... bref de construire un simulacre d'un objet ou d'un environnement physique dans lequel on pourra modifier à volonté une seule variable indépendante en se gardant bien de modifier les autres grandeurs physiques de la situation expérimentale.

La deuxième caractéristique est la mesure. Il est en effet nécessaire de pouvoir décrire quantitativement comment la perception est modifiée en fonction de la grandeur physique observée. Là encore, la simulation est intéressante puisqu'elle permet d'observer la situation expérimentale *in situ*, soit en cours de simulation, soit *a posteriori* grâce à l'enregistrement du déroulement de l'expérience. La simulation présente là un réel avantage, puisque l'observation d'un objet *in situ* n'est pas possible dans le monde réel sans introduire de perturbations sur l'objet observé.

Tout ceci serait idéal s'il n'existait pas une limite fondamentale des techniques de simulation multisensorielle en temps réel, à savoir la frontière inévitable entre le monde réel et le monde virtuel. Dans le cadre de l'utilisation des technologies du virtuel pour la psychophysique, nous devons considérer deux approches. La première, restrictive, consiste à prendre pour objet l'humain en interaction avec un environnement ou un objet virtuel. Cette approche n'est pas confrontée à la frontière entre le réel et le virtuel, puisque l'objet d'étude devient l'humain *dans* le monde virtuel. La deuxième approche, plus générale, consiste à utiliser les technologies du virtuel pour la conception de matériaux expérimentaux, et il est supposé que les résultats pourront être généralisés à la connaissance de l'homme dans son environnement réel. Dans ce cas, la frontière entre le réel et le virtuel se retrouve au cœur de la situation expérimentale, et il devient nécessaire de pouvoir caractériser le matériau expérimental virtuel dans le monde réel. Ceci impose que l'objet virtuel soit vu comme constant dans le monde réel, c'est-à-dire que la durée d'un pas de simulation doit être constante et que les entrées et les sorties du processus de simulation soient synchronisées avec les entrées et les sorties des transducteurs (TGR, écran, haut-parleurs). Enfin, chaque composant de la chaîne de simulation doit pouvoir être calibré précisément. Cette démarche est fondamentale en particulier pour la psychophysique puisqu'elle permet d'appliquer les principes de la métrologie aux objets virtuels simulés.

Toutes les réalisations de simulateurs multisensoriels en temps réel de l'ACROE sont conformes à ces principes. D'une part, le processus de simulation temps-réel est synchronisé sur les entrées et les sorties des transducteurs et du TGR en particulier. D'autre part, le formalisme CORDIS, calculé avec un pas de simulation constant, permet une caractérisation des objets simulés avec les grandeurs de mesure du monde réel. La plateforme de simulation ERGON_X répond à ces critères, ce qui nous permet d'appliquer les principes de métrologie à l'objet physique que nous simulons et d'en faire un objet pour l'expérimentation psychophysique.

3.2 Perception de matériaux déformables

3.2.1 Objectifs

L'élasticité est une des propriétés rhéologiques élémentaires de la matière du point de vue de la perception, de même que la masse par exemple. La perception des propriétés rhéologiques des objets est un sujet largement couvert par la psychologie expérimentale. En particulier, Jones et Hunter ont établi les premiers seuils de perception de l'élasticité à l'aide d'un retour d'effort dès la fin des années 1980 [Jones, 1989, Jones et Hunter, 1990], puis Tan et al. ont montré que la perception de l'élasticité est liée à l'effort de mouvement et à l'intensité de la force terminale [Tan *et al.*, 1995]. Une bibliographie plus complète est disponible dans [Couroussé *et al.*, 2006b].

Cependant, nombre de ces études se sont basées sur l'utilisation d'une interaction viscoélastique simple réelle ou virtuelle (i.e. un ressort). Dans ce cas en effet, la relation entre déformation et force d'interaction est linéaire, alors que ce n'est que rarement le cas dans notre environnement quotidien. Nous avons donc voulu étudier la perception de l'élasticité dans le cas d'un objet déformable complexe, présentant une caractéristique force-déplacement non linéaire. Nous avons en outre exploité l'utilisation de la plateforme de simulation :

- En enregistrant une trace des mouvements et des forces d'interaction au niveau du TGR pour analyse postérieure,
- En étudiant les effets de l'ajout ou de la suppression de la modalité visuelle sur la perception de l'élasticité,
- En évaluant l'effet de la modalité visuelle sur les caractéristiques des mouvements exploratoires.

3.2.2 Description de la scène

Le modèle implanté avait pour objectif de simuler un matériau déformable, avec lequel un utilisateur pourrait interagir par l'intermédiaire du TGR, et visualiser à l'écran. La scène simulée représente donc un objet déformable, que nous appellerons dans la suite de ce chapitre la *pâte*. La pâte se déforme soit par l'interaction avec l'objet manipulé par l'utilisateur, soit par l'interaction avec d'autres objets du modèle.

Nous utilisons l'interface ERGOS équipée d'un stick 3D. Cette interface ne permet qu'une interaction outillée avec l'environnement simulé, et l'extrémité du stick 3D, manipulée par le sujet, est représentée par un module <MAT> dans le modèle. Les déplacements se font dans un plan 2D parallèle au plan transversal du sujet, et l'axe vertical de déplacement est bloqué par la simulation d'une raideur importante. L'espace de modélisation est donc 2D.

La tâche demandée à l'utilisateur était de juger de l'élasticité de la pâte, en déplaçant cette pâte au travers des deux obstacles afin de la déformer.

Notre modèles contient donc les catégories d'objets suivants :

1. La pâte : le ou les objets déformables dont l'élasticité était jugée.
2. L'objet manipulé par l'utilisateur (la métaphore de la main de l'utilisateur dans le modèle), grâce auquel ce dernier peut interagir avec la pâte : il s'agit d'une « boîte » circulaire dans laquelle la pâte est enfermée. L'utilisateur peut ressentir les interactions de contact entre cette boîte et la pâte.
3. Les obstacles : des objets fixes, en interaction avec les objets déformables. Deux obstacles sont utilisés ; ils sont disposés de telle sorte que la pâte se déforme lors de son passage entre les deux obstacles.
4. Les éléments d'interaction entre les objets des catégories (1), (2) et (3).

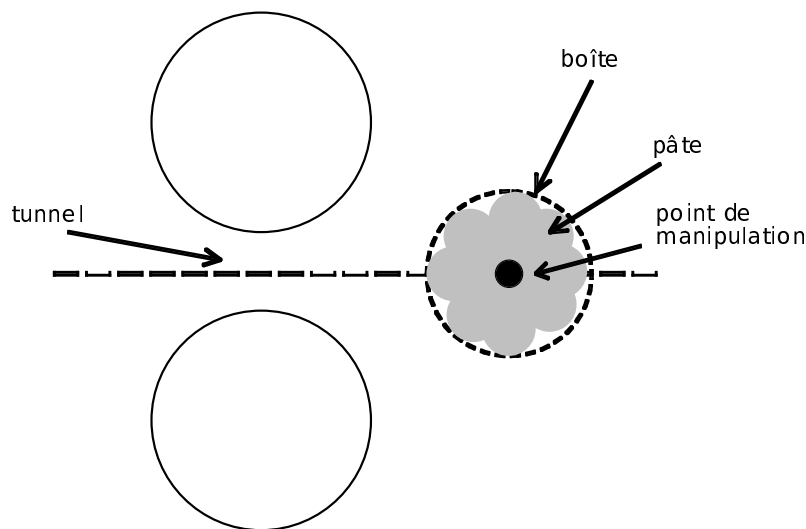


Figure 6.7. Représentation schématique du modèle simulé

La pâte est modélisée par un réseau de 20 modules MAS. Afin d'obtenir une représentation homogène et isotrope de la pâte, les modules MAS sont interconnectés par des modules REF de longueur au repos non nulle et identique. La masse totale de la pâte est la somme de la masse des 20 modules MAS ; elle est choisie de telle sorte que les forces d'inertie soient négligeables devant les forces d'interaction avec le sujet. Sept valeurs de rigidité différentes ont été choisies (K_1 à K_7). Dans l'objet pâte, tous les modules REF sont paramétrés avec une valeur de raideur égale ; il s'agira de la variable indépendante de l'expérience.

Les masses constituant l'objet pâte sont des éléments inertiels ponctuels, et ils ne sont donc pas déformables individuellement. C'est le réseau de ces masses qui permet de représenter un objet déformable, puisque les distances entre les éléments, calculées par les modules REF, varient en fonction des forces d'interaction. En conséquence, dans la représentation de la pâte, il fut choisi de représenter chaque masse individuellement par une forme circulaire pleine de taille constante. C'est seulement la distance entre ces éléments qui change au cours de la simulation, donnant à l'utilisateur l'impression d'un objet déformable (figure 6.8). Notre modèle de simulation étant en

deux dimensions, chacun des éléments « masse » de la pâte était représenté par des cercles pleins sans ombrage, dont le diamètre était suffisamment important pour que la représentation globale de la pâte ne comporte pas de vides.

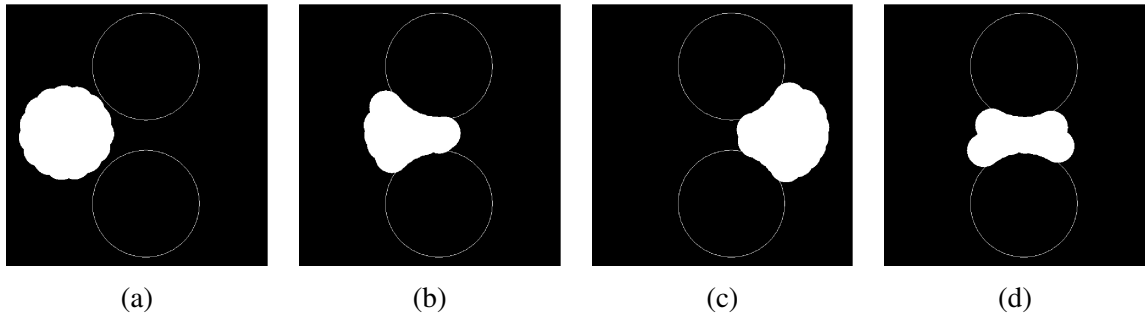


Figure 6.8. Captures d'écran de la simulation : la pâte passe entre les obstacles de gauche (a) à droite (d) (raideur faible)

Afin de permettre à l'utilisateur de contrôler les mouvements de la pâte et de ressentir les forces d'interaction entre la pâte et les obstacles, la pâte est enfermée dans une « boîte » circulaire dont la position est contrôlée par les mouvements du TGR. Le diamètre de cette boîte est légèrement inférieur au diamètre de la pâte au repos, de telle sorte que la boîte et la pâte sont toujours en contact ; la pâte ne peut donc pas se déplacer dans la boîte. Ainsi, aucune collision entre la pâte et la boîte n'est perçue par l'utilisateur, et les mouvements de la pâte suivent exactement ceux de la boîte. En outre, cette boîte n'est pas représentée graphiquement pendant la simulation. Ceci donne à l'utilisateur l'impression de déplacer la pâte plutôt qu'un objet intermédiaire. Le paradigme de manipulation ainsi obtenu reste « transparent » du point de vue de l'utilisateur, donnant au sujet l'impression qu'il/elle manipule directement la pâte sans intermédiaire. Pendant l'expérience pilote et pendant l'expérience finale, les participants n'ont jamais relevé cet artefact du modèle, ou ne se sont plaints du manque de contrôlabilité de la pâte ou d'une absence de transparence dans la manipulation.

Dans la condition expérimentale avec le retour haptique seul, l'expérience pilote a montré que le sujet perdait facilement ses repères spatiaux en mouvement libre, c'est-à-dire dès que la pâte n'était plus en contact avec les obstacles. Nous avons donc restreint les mouvements de l'utilisateur à un axe parallèle à l'axe transversal, passant par le milieu du goulet d'étranglement formé par les deux obstacles : en utilisant un guidage élastique avec une raideur importante (1 N/mm). Les forces appliquées par ce guidage étaient purement orthogonales aux mouvements de déplacement de la pâte au travers du goulet.

La figure 6.9 représente trois profils force/déplacement pour des valeurs de raideur K_1 , K_4 et K_7 . Il s'agit des valeurs de force et de déplacement mesurées dans l'espace réel de manipulation en cours d'expérience (figure 6.10). Les valeurs sont obtenues à partir de la trace enregistrée en temps réel à partir des variables de simulation. Les échelles de mesure sont obtenues par l'application à la trace enregistrée des coefficients de gain de l'électronique et des coefficients modificateurs introduits par la transformation géométrique MPVR. Cette figure montre bien la relation non linéaire établie entre le déplacement imprimé par le sujet, et la force d'interaction établie entre le stick 3D (la pâte) et les obstacles. L'intensité maximum de la force d'interaction dépend uniquement de la valeur du coefficient de raideur.

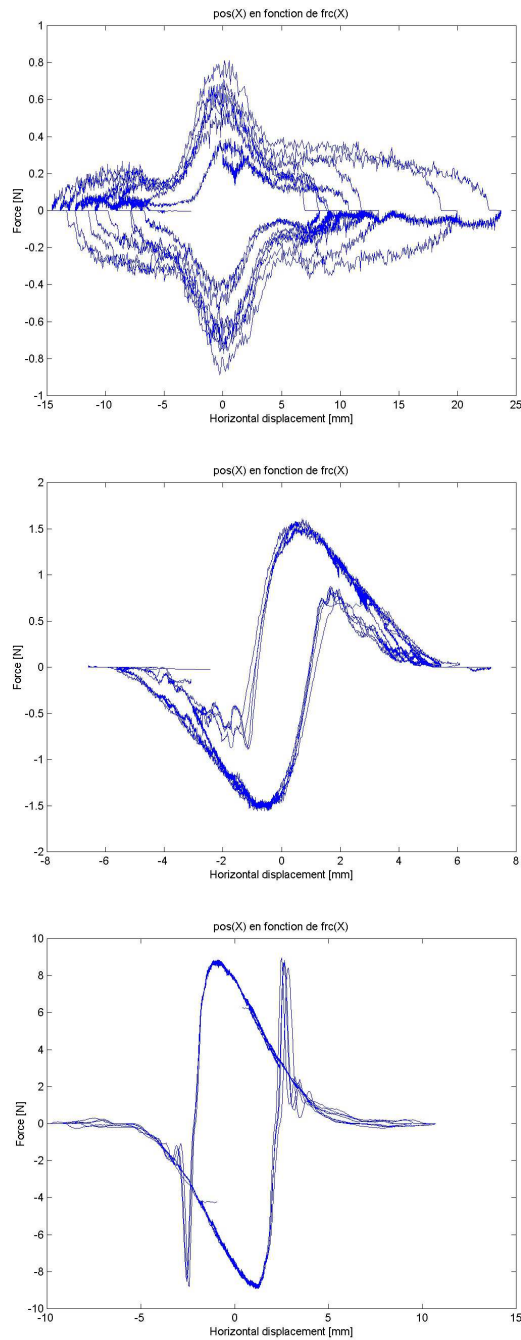


Figure 6.9. Courbes force d'interaction en fonction du déplacement obtenues à partir de l'enregistrement de traces expérimentales ; de haut en bas respectivement pour K_1 , K_4 et K_7

3.2.3 Méthode et protocole expérimental

Six sujets ont participé à l'expérience pilote, et douze sujets ont participé à l'expérience finale. Les sujets étaient âgés de 22 à 36 ans. Tous étaient droitiers et ne présentaient pas de handicap visuel ou moteur.

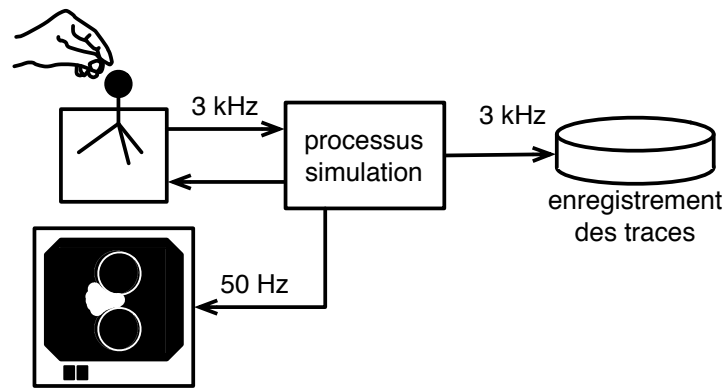


Figure 6.10. Schéma fonctionnel du modèle de pâtes

Deux conditions expérimentales ont été testées : (1) geste seul, et (2) geste et retour visuel. Après un bref apprentissage et une familiarisation avec le matériel expérimental, deux séries étaient présentées aux sujets, une pour chaque condition expérimentale, et l'expérience était renouvelée deux fois (dans un ordre différent) pour tester les effets de l'apprentissage.

Les sujets devaient pour chaque nouveau test juger de l'élasticité du modèle qui leur était présenté en déplaçant la pâte au travers des obstacles. Ils devaient attribuer une valeur numérique à chaque essai sur une échelle de 1 à 9. Les valeurs décimales étaient autorisées, une valeur numérique faible correspondant à une élasticité faible, une valeur numérique importante à une élasticité importante. Chaque série de tests comprenait sept valeurs différentes de raideurs, présentées dans un ordre aléatoire ; avant chaque série les sujets pouvaient manipuler la raideur la plus faible (K_1) et la plus importante (K_7), associées respectivement aux valeurs numériques 1 et 9.

Les positions et les forces de la manipulation sont enregistrées en temps réel à la fréquence de simulation (3 kHz). L'enregistrement des traces (figure 6.10) permet de compléter l'analyse par une observation objective du déroulement de l'expérience.

3.2.4 Résultats

Nous ne reportons pas dans ce document l'analyse complète et l'ensemble des conclusions tirées de l'expérience, mais on pourra se reporter à [Couroussé *et al.*, 2006b] pour un exposé plus complet.

Notre expérience met le sujet dans une situation d'interaction outillée avec un objet virtuel. L'interaction avec l'objet s'effectue par le biais d'un outil, matérialisé par le stick 3D utilisé pour l'expérience, de sorte qu'il n'est pas possible pour le sujet d'obtenir des informations sur l'élasticité de l'objet par les informations cutanées, comme les déformations de surface, et la relation entre les forces d'interaction outil-objet et le déplacement imprimé par le sujet sur l'outil est non linéaire. Les sujets ont parfaitement réussi à ordonner les différentes valeurs d'élasticité présentées sur une échelle numérique subjective bornée par deux modèles servant d'étalon, et ceci dans les deux conditions expérimentales : geste seul, ou geste combiné avec un retour visuel de la scène simulée.

Les résultats de cette expérience sont cohérents avec l'état de l'art sur la perception de l'élasticité : l'échelle perceptive de l'élasticité est une échelle ordonnable, et présente la même allure chez tous les sujets (figure 6.11). Le retour visuel n'a pas eu d'influence significative sur le résultat des évaluations. Les sujets ont par ailleurs confirmé dans l'entretien qui suivait l'expérience ne pas avoir porté beaucoup d'attention sur le retour visuel, lorsque celui-ci était disponible. L'enregistrement des traces de l'expérience a permis de montrer cependant que l'ajout de la vision modifie la dynamique des mouvements de la tâche : amplitude des mouvements et vitesse de déplacement du point manipulé en mouvement libre [Couroussé *et al.*, 2006b]. Il semble que la principale contribution du retour visuel soit de permettre de situer la manipulation dans l'espace.

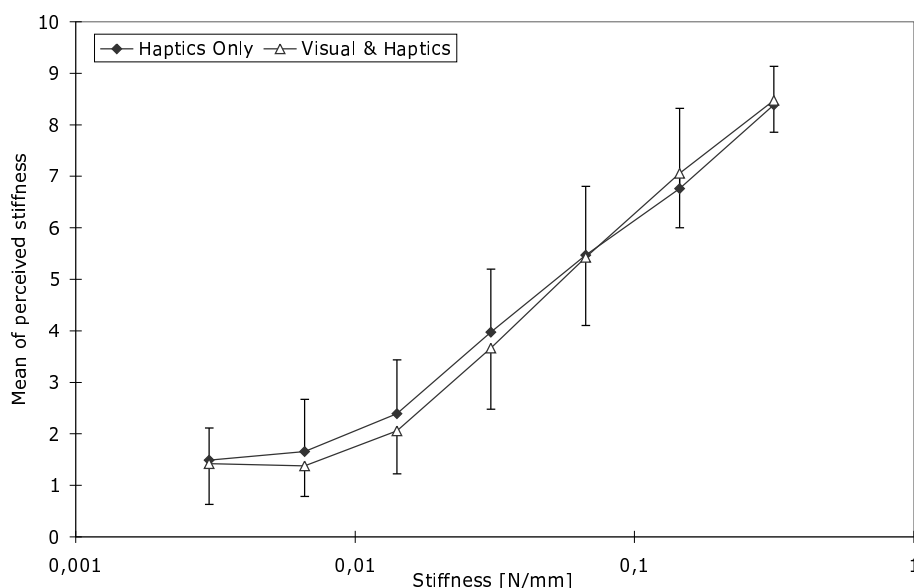


Figure 6.11. Echelle subjective moyenne attribuée par les sujets en fonction de la raideur

Srinivasan, Beauregard et Brook ont étudié en 1996 l'effet de la contradiction entre les stimuli visuels et les stimuli gestuels dans une tâche d'évaluation de l'élasticité [Srinivasan *et al.*, 1996]. Deux ressorts étaient simulés, et les sujets devaient estimer la raideur des deux ressorts à l'aide d'un dispositif à retour d'effort permettant de simuler la compression manuelle du ressort, et d'une représentation graphique de la déformation. Le dispositif expérimental permettait de modifier graduellement la relation entre déformation gestuelle et déformation visuelle, depuis une situation où la déformation visuelle était consistante avec la déformation gestuelle simulée, jusqu'à la situation inverse où la représentation visuelle du ressort le moins raide était associée à l'interaction gestuelle du ressort le plus raide et vice versa. Les résultats montrent une dominance significative du retour visuel par rapport aux informations gestuelles de déformation.

Lecuyer et al. ont affiné la notion de dominance visuelle en 2001 en introduisant le terme « boundary of illusion » [Lecuyer *et al.*, 2001], suggérant que le degré de la dominance des stimuli visuels sur les stimuli tactilo-kinesthésiques était dépendant de l'individu et de l'intensité des stimuli tactilo-kinesthésiques. En d'autres termes, plus les forces d'interaction étaient importantes, plus la contradiction entre vision et geste doit être importante pour que le jugement de l'élasticité se fasse sur la base des informations visuelles aux dépens des informations gestuelles. Les expériences étaient basées sur la comparaison entre des éléments déformables réels ou simulés

mais conformes à la réalité (déformation visuelle identique à la déformation gestuelle), et des interfaces gestuelles isométriques (comme des capteurs de pression), où la déformation visuelle du ressort simulé dépend de la pression gestuelle appliquée et non plus d'un déplacement. Lecuyer et al. ont montré que, avec de telles interfaces qui présentent une raideur propre constante (pour la main), il est tout de même possible de suggérer une élasticité variable en faisant varier la déformation visuelle.

Les deux études de Srinivasan, Beauregard et Brooks et de Lecuyer et al. sont intéressantes dans le sens où, fondées sur une inconsistance geste-vision, elles montrent une dominance de la vision sur le geste, alors que l'expérience présente, fondée au contraire sur le maintien de la consistance entre le geste et la vision, montre que le retour visuel était surtout utilisé pour situer la manipulation dans l'espace, et n'était que peu utilisé pour juger de l'élasticité.

3.3 Le tapping

3.3.1 Objectifs

La littérature en psychophysique abonde de références concernant l'utilisation du geste de battement pour l'étude de la synchronisation geste-son, et l'étude de la perception des délais geste-son [Couroussé *et al.*, 2006a]. Le geste de battement (tapping) est en effet un geste simple, qui peut être réduit à l'extrême lorsqu'il est effectué seulement avec l'extrémité de l'index.

La boucle de perception et d'action geste-son met en jeu des résolutions temporelles inférieures à la dizaine de milliseconde, voire inférieure à la milliseconde dans les gestes experts comme les gestes de musiciens professionnels. D'un point de vue technologique, il s'agit d'une situation particulièrement exigeante, puisque la résolution temporelle de la plateforme technologique utilisée pour une étude doit être meilleure que l'ordre de grandeur temporel du phénomène étudié. Il s'agit donc d'une situation expérimentale exigeante d'un point de vue général, mais d'autant plus dans le cas de la simulation haptique que les architectures de simulation ont une résolution temporelle du même ordre de grandeur que la résolution temporelle de la boucle geste-son, voire supérieure. C'est peut-être la raison pour laquelle jamais jusqu'à présent la simulation haptique ne fut utilisée pour l'étude du geste de tapping. En effet, la relation geste-son fut intensivement étudiée dans sa dimension temporelle (perception du délai geste-son, synchronisation du geste sur une référence auditive, etc.), mais peu dans sa dimension physique ou mécanique. Seuls les travaux de Aschersleben et Gehrke étudient l'influence de la perception tactile sur la performance de battement [Aschersleben *et al.*, 2000, Gehrke, 1995], en comparant la performance de sujets sains avec des sujets déafférentés (i.e. pas de perception cutanée ou kinesthésique). À notre connaissance, l'effet des propriétés rhéologiques de l'instrument avec lequel le percussionniste interagit ne fut jamais étudié jusqu'à présent.

Le modèle utilisé est un modèle de percussion simple : l'utilisateur manipule par l'intermédiaire du TGR une masse virtuelle qui vient percuter sur un oscillateur harmonique simple. Dans la tâche de percussion en rythme avec une source audio isochrone, un second oscillateur simple non amorti est utilisé pour donner la pulsation de référence.

3.3.2 Modèle

Le modèle de percussion est un modèle 1D comprenant deux éléments (figure 6.12) : la structure vibrante, constituée d'un oscillateur harmonique simple modélisé par un module CEL 1D (6), et le système percuteur, modélisé par un module MAT 1D que nous appelons le marteau (3), relié au TGR (1) par un module REF 1D (2).

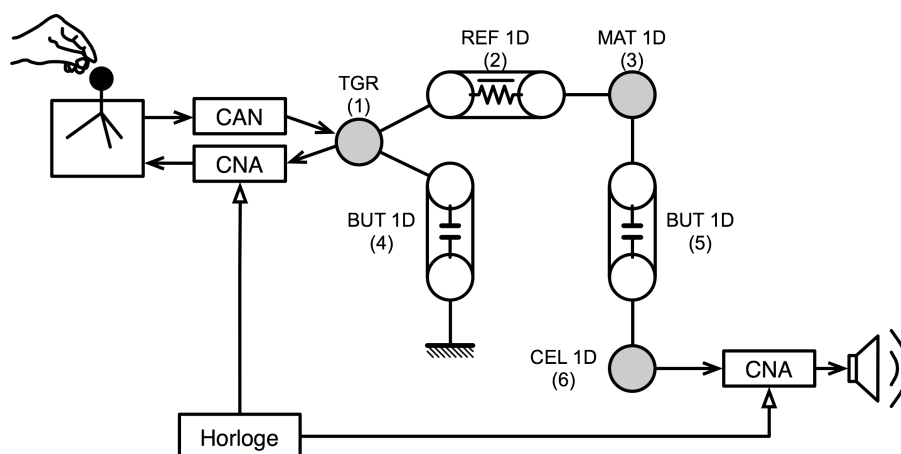


Figure 6.12. Le modèle CORDIS utilisé pour l'expérience de tapping

Ceci constitue un modèle classique de percussion tel qu'il est possible de le concevoir à l'aide de CORDIS. Afin de pouvoir construire une situation satisfaisante pour l'expérimentation sychophysique, quelques adaptations sont nécessaires.

Le module BUT 1D (1) ajoute une interaction de butée supplémentaire entre le module TGR et un point fixe (module SOL). En effet, du point de vue du geste de percussion, la structure vibrante percutée est assimilée à un point fixe ; mais les propriétés d'émission sonore de la structure vibrante sont liées à la dynamique du geste de percussion. L'introduction d'une butée supplémentaire entre le module TGR et un véritable point fixe permet de dissocier l'interaction de butée ressentie de la percussion marteau-structure vibrante, et en particulier de dissocier les propriétés physiques de la butée (raideur et viscosité) des propriétés d'émission sonore de la percussion. Ainsi, pour l'expérimentation la variable indépendante est la raideur de la butée. Cette variable devient en particulier indépendante de la qualité du son de percussion, ce qui n'est pas le cas dans le modèle de percussion constitué du triplet marteau, butée, structure vibrante.

L'introduction de cette butée supplémentaire permet également de rendre l'ordre de grandeur du système { marteau + structure vibrante } petit devant celui du système TGR + butée. Ainsi, l'interaction entre le module TGR et la structure vibrante, toujours présente par les liaisons REF 1D puis BUT 1D (5), devient négligeable et imperceptible devant l'interaction du système TGR-butée (BUT 1D no 4 sur la figure 6.12).

Le module TGR est connecté au moteur ERGOS, habillé d'un stick 1D ERGOS. Le geste de percussion est alors effectué par le sujet parallèlement à un axe vertical.



Figure 6.13. Photographie de l'expérience de tapping ; seuls le poignet et la main participent aux mouvements verticaux de percussion, le coude est au repos sur la table

3.3.3 Résultats

De même que pour la présentation de l'expérience sur la perception de l'élasticité, sont seulement reportés ici les résultats les plus significatifs. L'étude complète est publiée dans [Couroussé *et al.*, 2006a].

L'objectif principal de cette expérience était de déterminer si une modification des propriétés physiques de l'objet percuté pouvait avoir un effet sur la performance de tapping. En prenant comme mesure le délai entre l'instant du contact entre le point manipulé et la butée de contact d'une part, et l'instant de la percussion de la structure vibrante servant de métronome (i.e. le début de l'émission du son de référence) d'autre part, une asynchronie négative est mesurée dans la tâche de tapping synchronisé. Cette asynchronie négative présente un délai variant entre -10 et -80 ms, jusqu'à -200 ms avant l'émission du son de référence sur lequel le sujet doit synchroniser son geste de percussion, et nos résultats sont consistants avec les autres résultats présentés dans la littérature [Mates et Aschersleben, 2000, Aschersleben *et al.*, 2000, Aschersleben, 2002]. Aussi, l'asynchronie moyenne augmente lorsque la raideur de l'objet percuté diminue, de même que la dispersion des résultats diminue avec l'augmentation de la raideur de l'objet percuté. Dans la tâche de percussion rapide sans référence sonore, la fréquence maximale de percussion atteignable augmente significativement lorsque la raideur de l'objet percuté augmente.

Le geste de percussion est un des gestes humains les plus simples mettant en jeu un couplage fort, mais épisodique (le temps du contact seulement), entre la main et l'objet percuté. En ce sens, il s'agit d'un cas élémentaire de geste instrumental. Dans cette situation, les propriétés physique de l'objet percuté jouent un rôle essentiel dans cette relation, puisqu'elles modifient la performance d'une tâche basée sur le geste de percussion.

4 Scénarios Emblématiques de l'Enaction

4.1 Le Scénario Emblématique « Pebble Box »

Le cadre général de modélisation est basé sur le paradigme de la *PebbleBox*, qui signifie « boîte à cailloux » en anglais. Il fut l'objet de trois implantations complémentaires (1) par Sile O'Modhrain et Georg Essl anciennement au Media Lab Europe, (2) par Charlotte Magnusson au CERTEC à l'Université de Lund, Suède, et (3) par le groupe de recherche ACROE et ICA dans le cadre de cette thèse. Ces trois implantations présentent des caractéristiques techniques et des interfaces complémentaires : alors que (1) permet l'interaction manuelle directe avec des cailloux réels dont les sons sont recréés par ordinateur, (2) et (3) sont basées sur l'interaction outillée avec un environnement entièrement virtuel.

À partir de ces trois implantations, un travail de recherche fut mené dans le cadre du projet européen Enactive Interfaces, sous la direction de Annie Luciani. Un premier travail préparatoire d'exploration fut réalisé en Décembre 2005 [Luciani *et al.*, 2005b, Magnusson *et al.*, 2006]. Ce travail exploratoire préliminaire fut l'occasion de poser les bases exploratoires du futur travail expérimental. Puis une seconde session exploratoire fut réalisée en février 2007, cette fois mettant en parallèle les trois implémentations de la PebbleBox [Luciani *et al.*, 2007a]. L'objectif était de faire émerger des axes exploratoires pour la création d'un scénario emblématique destiné au grand public. Ce travail a mené à la réalisation d'un scénario interactif conçu pour le grand public (EES) qui fut présenté lors la conférence Enactive/07 en Novembre 2007, dans l'exposition « Toucher le Futur ».

4.1.1 La PebbleBox de Sile O'Modhrain et Georg Essl, MediaLab Europe (PB1)

Cette PebbleBox fut développée par Sile O'Modhrain et Georg Essl originellement au Media Lab Europe [O'Modhrain et Essl, 2004]. Il s'agit d'un coffre de dimensions suffisantes pour pouvoir y plonger les deux mains, dans lequel sont déposés un grand nombre de galets. L'intérieur du coffre est recouvert de mousse épaisse afin que les galets ne viennent pas s'entrechoquer contre les parois du coffre. Un microphone, fixé au fond du coffre et invisible de l'opérateur, capte le son des galets s'entrechoquant pendant la manipulation.

L'utilisateur se trouve en contact visuel direct avec la PebbleBox, et les sensations haptiques proviennent directement de la manipulation des cailloux dans le coffre, tandis que les retours auditifs proviennent du traitement en temps réel des sons captés de la manipulation, et modifiés par synthèse granulaire.

L'intérêt de ce dispositif est de créer des associations inhabituelles entre les sensations auditives et gestuelles de l'exploration de la PebbleBox.

Le procédé de synthèse granulaire permet d'associer la synthèse d'un son à la détection d'un événement sonore, dans le cas de la PebbleBox, la détection de l'entrechoquement de deux galets. À chaque détection d'un choc, l'algorithme de synthèse granulaire génère un nouveau son, en prenant en compte des paramètres comme l'amplitude du son détecté, la densité des chocs détectés à l'instant de la génération d'un nouveau son, etc. Les sons sont générés par l'algorithme de

synthèse granulaire sur la base de sons préenregistrés : chants d'oiseaux, tintement de pièces de monnaies, craquement d'une pomme, clapotis de l'eau...

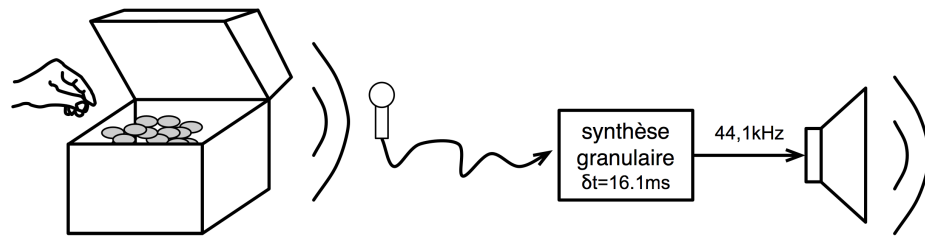


Figure 6.14. Schéma fonctionnel de la PebbleBox développé par O'Modhain et Essl (Media Lab Europe, 2004)

4.1.2 La PebbleBox de Charlotte Magnusson – ULUND (PB2)

Charlotte Magnusson a développé au CERTEC à l'université de Lund (Suède) un autre modèle de PebbleBox, basé sur l'utilisation du système haptique PhantomDesktop [Magnusson *et al.*, 2006]. Il s'agit d'un modèle purement virtuel, tridimensionnel, constitué d'une boîte rectangulaire dans laquelle peuvent se déplacer des objets sphériques (les cailloux). L'utilisateur peut interagir avec la PebbleBox par le biais du manche du système haptique : la position d'un des cailloux, de couleur distincte, est contrôlée par les mouvements appliqués au système haptique, et les forces d'interaction entre cet objet, les autres cailloux et la boîte sont retournées à l'utilisateur par le biais du système haptique. La représentation graphique et le modèle mécanique sont calculés à partir de la bibliothèque OpenHaptics de Sensable conjointement avec la bibliothèque OpenDynamicsEngine. L'ensemble de l'application est exécuté sur le système d'exploitation Windows XP sans patch pour le temps réel. La fréquence de simulation se situe autour de 1 kHz, et la vitesse de rafraîchissement de la boucle de simulation dépend donc fortement de la puissance de la machine utilisée. Les sons de la PebbleBox sont synthétisés en temps réel à partir d'une bibliothèque de sons préenregistrés et des paramètres de collision des cailloux (instant de collision et vitesse d'impact), à l'aide de la bibliothèque Direct3DSound permettant la spatialisation des sons synthétisés.

Cette implantation du modèle de PebbleBox présente des similarités fortes avec la PB1 : dans les deux cas, les sons générés sont obtenus à partir de sons préenregistrés, dont la synthèse est déclenchée à la détection d'un événement particulier : un pic dans le signal sonore enregistré attestant la présence de la collision entre deux cailloux pour la PB1, la collision entre deux cailloux virtuels dans la PB2. Le modèle de synthèse sonore est ensuite affiné pour apporter plus de crédibilité au son synthétisé : dans la PB1, l'amplitude du son synthétisé est proportionnelle à l'amplitude du son détecté et la fréquence de synthèse du son pré-enregistré, choisi parmi un dictionnaire de sons similaires, est modulée pseudo aléatoirement afin d'éviter un effet de monotonie qui serait rapidement perceptible. Dans la PB2, un mécanisme similaire est mis en place à partir des collisions détectées dans le modèle mécanique simulé. Ce mécanisme crée une rupture entre les parties mécaniques et acoustiques du modèle simulé : la génération des phénomènes acoustiques est obtenue à partir de la génération d'événements transmises du modèle mécanique

au modèle acoustique (figure 6.15). Il s'agit d'une situation de *contrôle* unidirectionnel basée sur le paradigme du signal : les événements acoustiques sont *déclenchés* selon les collisions détectées dans le modèle mécanique. L'intérêt de ce modèle est de permettre une plus grande liberté de choix dans les stimuli auditifs, puisque ceux-ci sont complètement dissociés du modèle mécanique. De la même manière que dans la PB1, il est alors possible d'associer des sons contre-nature au modèle mécanique simulé (par exemple, des bruits de craquement de pomme). La contre-partie est alors le fait que les phénomènes acoustiques n'ont plus aucune relation avec le comportement mécanique du modèle, et ce n'est que par le choix judicieux de stimuli audio adaptés qu'il est possible de recréer l'illusion d'un modèle multisensoriel cohérent.

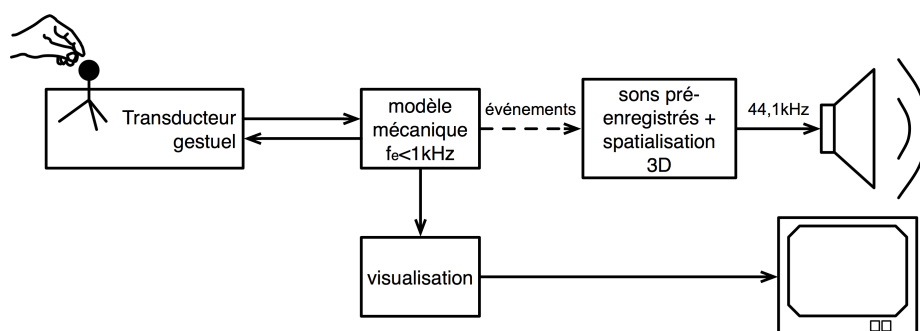


Figure 6.15. Diagramme fonctionnel de la PebbleBox développée par Charlotte Magnusson (ULUND)

Les deux modèles de PebbleBox présentés ci-dessus créent une situation d'interactivité riche en exploitant la rupture physique entre les galets manipulés et les retours sonores qui peuvent évoquer des situations impossibles avec des objets naturels. Au contraire, le modèle implanté sur le simulateur ERGON_X est basé sur le maintien d'une consistance physique étroite entre les stimuli gestuels, visuels et auditifs.

4.1.3 La PebbleBox ACROE (PB3)

Ce modèle de PebbleBox met l'accent sur la consistance énergétique entre le comportement mécanique simulé, retournée à l'utilisateur par le biais d'un TGR ERGOS, et le comportement acoustique simulé : un seul modèle acoustico-mécanique est calculé, permettant d'obtenir à la fois la synthèse du comportement mécanique et la synthèse du comportement acoustique (figure 6.16). La première différence fondamentale de ce modèle avec le modèle PB2 réside donc dans le choix de ne pas séparer la composante mécanique à basse fréquence (pour l'interaction manuelle) et la composante sonore du modèle.

Le modèle comprend une partie bidimensionnelle, pour la modélisation des objets mobiles, et une partie unidimensionnelle, pour la modélisation des structures vibrantes responsables de la production des phénomènes acoustiques. Il comprend une « boîte » qui permet de contraindre le mouvement des cailloux, mobiles, dans un espace restreint. La boîte est circulaire et fixe dans le repère spatial du modèle ; elle est donc modélisée par un point matériel fixe (SOL). Les éléments mobiles, les « cailloux », sont modélisés par des éléments matériels 2D (MAS) en interaction avec la boîte par le biais de butées simples (BUT). Le seuil de ces butées détermine le rayon de la boîte.

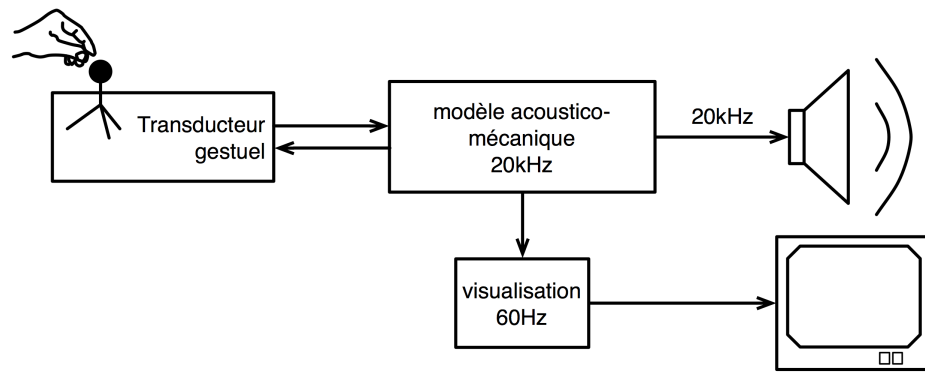


Figure 6.16. Schéma fonctionnel de la PebbleBox ACROE

Dans les PB1 et PB2, un événement sonore est créé à partir de la détection d'un événement « collision ». Dans ce modèle, chaque objet mobile caillou est associé à une structure vibrante : il s'agit d'un oscillateur harmonique du second ordre modélisé par un module CEL 1D. La collision entre deux cailloux mettra alors en mouvement les deux structures vibrantes qui leur sont associées : une collision est donc susceptible de créer deux événements sonores simultanés mais distincts. Ceci signifie que dans le modèle de cette PebbleBox, chaque caillou possède son identité sonore propre, alors que dans les PB1 et PB2, pour une collision donnée, le son associé est recréé à partir d'un échantillon sonore choisi aléatoirement dans une base de sons pré-enregistrées.

Pour produire du son, un principe de projection simple est nécessaire, étant donné que les cailloux sont modélisés par des éléments 2D, mais que les signaux sonores sont unidimensionnels. De plus, la représentation des objets mobiles (les cailloux) par des éléments matériels ponctuels simples tels que les modules MAS ne permet pas de représenter les phénomènes vibratoires. Les cailloux sont donc modélisés comme des sphères pulsantes : leur déplacement dans l'espace du modèle est représenté par la position du module MAS 2D, et le mouvement radial, correspondant à la déformation de l'objet susceptible de générer un son, est représenté par le mouvement du module CEL 1D (figure 6.17).

Nous utilisons donc un module physique qui permet de conserver les propriétés énergétiques des signaux vibratoires 2D par l'extraction des mouvements radiaux : le LPR. Le module LPR est décrit dans [Florens *et al.*, 2005]. Sans plus entrer dans les détails, mentionnons simplement que ce module permet d'associer les modules MAS 2D modélisant les cailloux à leur structure vibrante modélisée par un module CEL 1D, tout en conservant la relation de collision entre chacun des cailloux.

Nous calculons dans une même boucle de simulation à la fois le calcul des comportements mécaniques pour la relation avec le système ERGOS ainsi que le calcul des comportements acoustiques. Il est donc nécessaire de choisir une fréquence de simulation suffisamment importante pour pouvoir générer des signaux dans le domaine fréquentiel acoustique, c'est-à-dire au-delà de 10 kHz. Outre les questions de consistance mécanique, le fait de simuler un objet acoustico-mécanique dans une seule boucle de simulation permet de réduire la latence geste-son. En effet, dans les PebbleBox PB1 et PB2, une latence non négligeable est introduite entre la détection de la collision de deux cailloux et la génération de l'événement sonore associé, en raison de la séparation des composants et de la complexité logicielle du processus de génération du son. Du point de vue

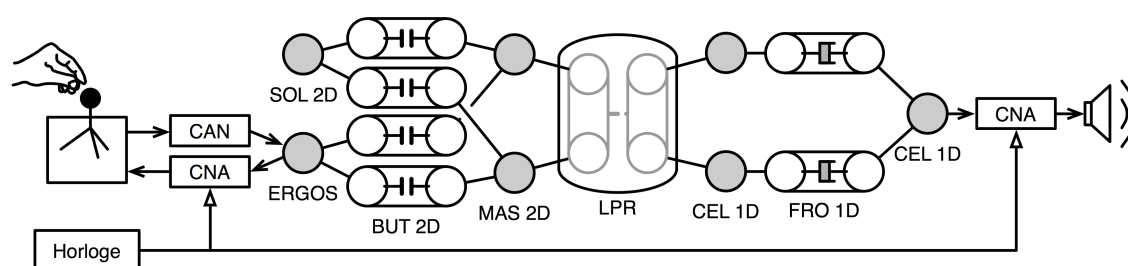


Figure 6.17. Topologie du modèle de PebbleBox ; le modèle est représenté pour $n = 2$ cailloux (MAT 2D) seulement ; la visualisation est créée à partir des positions des modules MAS 2D.

signal, dans ce modèle la latence geste-son est strictement égale à une période d'échantillonnage, soit 0,1 ms pour une fréquence de simulation à 10 kHz. Du point de vue des phénomènes physiques générés, la latence geste-son sera égale à une période d'échantillonnage additionnée au temps de génération des phénomènes physiques par aux transducteurs utilisés (TGR ERGOS pour le geste, haut-parleurs pour les sons). Dans le cas du son, il faudrait en toute rigueur tenir également compte de la propagation du son entre les haut-parleurs et les oreilles de l'utilisateur. Cette latence reste dans tous les cas inférieure à celle des PB1 et PB2.

Le modèle PB3 est relativement complexe, et comporte de nombreux modules d'interaction. Pour pouvoir simuler n cailloux, le modèle comportera n BUT 2D pour l'interaction caillou-système ERGOS, n BUT 2D pour l'interaction caillou-boîte, $n \times (n - 1)/2$ modules LPR pour les interaction entre cailloux, soit un total de $n \times (n + 3)/2$ modules d'interaction 2D. Les modules d'interaction 2D sont relativement lourds à calculer au regard des modules de type <MAT> 2D ou des modules 1D en général, puisque leur algorithme comporte une racine carrée pour le calcul de la distance scalaire, puis une division flottante pour le calcul de la force résultante sur chaque dimension.

La première passe d'implantation de ce modèle de PebbleBox a permis de calculer seulement 8 cailloux en temps réel à la fréquence de 10 kHz. La fréquence de 10 kHz était relativement faible et présentait un effet de repliement spectral important sur les signaux sonores produits par le modèle. Cet effet de repliement spectral fut utilisé à profit, en conférant un timbre métallique au son produit par les collisions au lieu d'un son harmonique pur moins riche, produit habituellement par des modules CEL 1D.

À la suite de plusieurs passes d'optimisation successives, il est maintenant possible de calculer un modèle de PebbleBox comportant une petite dizaine de cailloux à la fréquence de 44,1 kHz.

4.1.4 Le Scénario

Les trois plateformes présentent, pour le même paradigme de la « boîte à cailloux » plusieurs axes de complémentarité :

- Manipulation directe / manipulation outillée : PB1 permet la manipulation à main nue des objets, alors que PB2 et PB3 présentent une situation d'interaction outillée : l'utilisateur se trouve

en interaction avec les objets (simulés) par l'intermédiaire du manche du TGR, métaphore du manche d'un outil permettant de « touiller » dans la boîte de cailloux simulée.

- Qualité du retour d'effort : PB2 utilise un TGR Phantom Omni, c'est-à-dire un système à retour d'effort de moindre qualité que le TGR ERGOS utilisé dans la PB3. Par « qualité », il faut comprendre la fidélité du système à représenter le phénomène simulé ; les caractéristiques pertinentes sont l'intensité des forces de friction en mouvement libre, la bande passante des capteurs et des actionneurs. L'espace de travail influe sur les membres mis en œuvre dans la manipulation (manipulation des doigts dans un petit espace de travail, ou mouvements impliquant le coude voire l'avant bras pour un espace de travail plus grand).
- Espace de modélisation : les objets sont complètement simulés dans PB2 et PB3 ; cependant, dans PB2 l'espace de modélisation est tridimensionnel, alors qu'il n'est que bidimensionnel dans PB3.
- Spatialisation du son : PB1 et PB3 utilisent une sortie son monophonique, alors que PB2 implante un système de spatialisation stéréophonique du son.
- Choix de la synthèse sonore : PB1 et PB2 mettent en œuvre une synthèse granulaire sur la base des collisions détectées par microphone (PB1) ou par simulation du modèle mécanique (PB2), alors que dans PB3 les sons synthétisés sont issus des phénomènes vibratoires acoustico-mécaniques simulés.

Un premier travail préparatoire d'exploration fut réalisé en Décembre 2005 dans le cadre du réseau d'excellence Enactive sous la direction de A. Luciani [Luciani *et al.*, 2005b, Magnusson *et al.*, 2006]. Ce travail exploratoire préliminaire fut l'occasion de poser les bases exploratoires du futur travail expérimental. En particulier furent dégagés les axes suivants :

- La causalité des actions, et la perception des effets sur l'objet manipulé. Nous avons relevé en effet qu'il était parfois de déterminer la cause de certains phénomènes — sonores en particulier. La raison est que la boîte de cailloux étant remplie d'objets mobiles, les sons entendus sont parfois le résultat de collisions en chaîne, et il est alors difficile pour l'utilisateur de déterminer la relation entre les phénomènes sonores produits et les mouvements effectués.
- L'émergence de la notion d'objet : si dans les trois situations expérimentales il existe toujours objectivement plusieurs objets sonores mobiles, les retours subjectifs évoquaient parfois des situations ou des objets éloignés de la réalité objective de la simulation, voire même irréalistes (« *The surface of the water is everywhere in the free space.* ») ou en contradiction avec le sentiment même de believabilité de la scène (« *It is completely different, it is too much abstract. It is impossible to imagine something. I can't believe !* »).

Une seconde session exploratoire fut réalisée en Février 2007 cette fois mettant en parallèle les trois implémentations de la PebbleBox. L'objectif était de faire émerger des axes exploratoires pour la création d'un scénario emblématique destiné au grand public et qui serait présenté à la conférence Enactive/07 en Novembre 2007.

Le premier résultat remarquable de ce travail expérimental est que, dans toutes les situations et pour les trois plateformes expérimentales utilisées, les sujets essaient d'inférer une scène ou un objet, même lorsque a priori les percepts sensoriels peuvent, pris indépendamment, être considérés comme complètement différents (typiquement, la sensation tactile et kinesthésique de la manipulation de galets secs, associée à des sonorités de chants d'oiseaux).

Le deuxième résultat remarquable est le fait que des stratégies exploratoires qui sont apparues naturellement au cours de la manipulation des sujets. Dans le cas de la PB1 et de la PB2, nous avons appelé EEP (Emergent Exploratory Procedures – Procédures Exploratoires Emergentes) ces stratégies exploratoires. Les EEP sont le résultat du processus de construction de l'interprétation d'une scène à partir des stimuli sensoriels qui sont présentés au sujet. Par exemple, dans le cas de PB1, avec les chants d'oiseaux : mimiques de pas de marche ; avec les sons d'eau : geste de jets de cailloux, de « nager » avec les mains ; avec les sons de croquement de pomme : geste de gratter le contenu de la boîte dans laquelle se trouverait un animal, gestes prudents d'approche comme si un animal se trouvait dans la boîte, etc.

Nous avons retrouvé les EEP dans le cas de PB2, dans l'alternance entre les scènes sans visualisation et les scènes avec visualisation.

Dans le cas de la PB3, plusieurs modèles furent testés en jouant sur la rigidité et la taille des cailloux, et la rigidité du manipulateur. Pour PB3, les différentes situations présentées jouaient sur les propriétés physiques des objets de la scène, mais tout en conservant une consistance physique complète de toute la chaîne instrumentale (de la main du sujet jusqu'aux structures sonores des cailloux), à l'inverse des PB1 et PB2 où la variation provenait d'une association voulue comme contradictoire ou inhabituelle entre les différents stimuli sensoriels. Nous avons aussi relevé une adaptation des stratégies de manipulation des sujets à la scène présentée, mais qui se traduit par des DMA (Dynamic Manipulation Adaptativity – Adaption Dynamique de la Manipulation) ou des AEL (Adaptive Experimentale Learning – Apprentissage Experimental Adaptatif). Les DMA sont des stratégies d'adaptation en ligne de la manipulation qui émergent en fonction des propriétés des objets manipulés : par exemple, quand les objets sont rigides, les sujets exercent une prise ferme sur le stick du TGR et font des mouvements énergiques ; à l'inverse, quand les objets sont moins rigides, les sujets font des mouvements plus souples avec une prise en main plus délicate du stick du TGR. Les AEL concernent l'évolution des stratégies de manipulation au fur et à mesure que le sujet se construit une représentation de la scène : quand les objets manipulés sont mous, les sujets rapportent premièrement ne rien sentir, puis décrivent progressivement un champ visqueux ou résistant, puis enfin un champ visqueux avec des objets mous (des « grumeaux »). À l'inverse, quand les cailloux sont rigides, les sujets décrivent le plus souvent un seul « gros objet », puis raffinent leur exploration en décrivant d'autres objets.

Lors de la présentation du scénario « PebbleBox » pour l'exposition grand public « Toucher le futur » de la conférence Enactive/07, les commentaires laissés par les visiteurs lors de leur exploration du scénario ont été enregistrés (annexe B). Ces commentaires confirment les observations faites lors de la deuxième session exploratoire en Février 2007. Aux stratégies de type EEP observées dans la session exploratoire pour l'exploration des PB1 et PB2, l'association d'une dimension narrative ou métaphorique est ici confirmée.

Pour la PB1, les suggestions de la scène évoquée sont : « *Impression de marcher dans l'eau, le mouvement des doigts se substitue au mouvement des jambes dans l'eau* », « *Galets au bord d'une rivière* » pour les sons d'eau ; « *There is something being disturbed by moving the stones* », « *manipulating little animals... ?* », « *seagulls singing the sines* » pour les sons d'oiseau ; « *Eating an apple* », « *marcher sur des insectes* » pour le craquement de pomme.

Pour la PB2, les suggestions possèdent toujours une dimension narrative, mais les descriptions se font plus précises. Sans visualisation : « *Une sorte de barrière* », « *Parcours de trajectoires à l'intérieur d'une sphère* » ; avec visualisation : « *déplacer une boule* », « *champ de forces* ».

Enfin, pour la PB3, la dimension narrative des descriptions s'efface pour laisser place à la description relativement précise d'objets. En particulier, on trouve fréquemment l'utilisation d'adjectifs qualificatifs des propriétés physiques des objets manipulés : « *A box of metallic marbles.* », « *jeux avec des boules élastiques* », « *un matériau granulaire* », « *j'ai l'impression de me déplacer sur un plan avec une légère rugosité* ».



Figure 6.18. Le Scénario Emblématique « Pebble Box »

4.1.5 Conclusion

Les trois implantations du paradigme de la « boîte de cailloux » que nous avons explorées dans ce travail expérimental sont une parfaite illustration de l'enaction : l'utilisateur se construit une représentation du monde simulé qu'il explore à partir de ses percepts. Cette connaissance nouvelle ne tire son origine ni dans les connaissances de l'utilisateur antérieures à la manipulation, puisque les objets manipulés ne possèdent aucun corrélat dans le monde réel, ni dans l'objet simulé lui-même, parce que la représentation construite par l'utilisateur est souvent différente de la réalité objective de ce qui est inscrit dans le programme de simulation. Par exemple, le modèle temps réel simulé dans PB2 et PB3 est un modèle d'objets mobiles sonores, alors que suivant les paramètres choisis, l'utilisateur peut se représenter une scène différente de sa réalité objective (« *J'ai l'impression de me déplacer sur un plan avec une légère rugosité au centre* »). La scène est construite subjectivement à partir de l'intégration de tous les percepts sensoriels, puisque dans le cas de PB1, le fait de modifier les retours sonores, dans le cas de PB2 le fait de modifier les retours visuels, et dans le cas de PB3 le fait de modifier les retours tactilo-kinesthésiques modifient complètement la nature de la scène que le sujet se représente.

Comme le suggère A. Luciani [Luciani *et al.*, 2007b], il y a un lien direct entre l'ergotique de la situation instrumentale et les effets d'adaptation dynamique ou d'apprentissage des mouvements exploratoires effectués par le sujet. Lorsque seule la relation entre le geste et les objets manipulés est ergotique et que la relation entre les gestes et les phénomènes sonores est purement épistémique et sémiotique (PB1 et PB2), nous observons davantage des procédures exploratoires (EEP), alors que lorsque la relation ergotique est continue jusqu'à la source sonore (PB3), nous observons davantage des adaptations dynamiques des mouvements (DMA, AEL). Au regard des résultats de la visite du scénario PebbleBox par plusieurs centaines de visiteurs, nous pouvons étendre cette observation de la dimension « ergotique » ou « non-ergotique » de la relation geste-son

sur le contenu des retours subjectifs que nous avons enregistrés pendant les visites ouvertes au grand public. En situation de relation geste-son épistémique ou sémiotique seulement (PB1, PB2), l'évocation de la scène porte une dimension narrative dominante devant la dimension descriptive, alors que lorsque la relation geste-son est complètement ergotique (PB3), la dimension narrative de la description s'efface devant une description détaillée (tableau 6.1).

Tableau 6.1. Stratégies d'exploration et contenu de la description subjective de la scène en fonction de la relation geste-son

	Non ergotique	Ergotique
Stratégie d'exploration	EEP	DMA / AEL
Description subjective	Dimension narrative forte	Dimension descriptive forte

4.2 Le Scénario Emblématique « Ergotic Sounds »

4.2.1 Historique du modèle

Le premier modèle de corde frottée fut implanté dès 1985 à l'ACROE par J.L. Florens. Ce modèle était implanté sur une architecture AP120/LS11, la partie sonore était simulée à 20 kHz et signaux gestuels échantillonnés à 300 Hz. Ce modèle ne comportait pas de retour d'effort.

En 1990, une nouvelle version fut implantée, toujours sur AP120, mais bénéficiant du clavier à retour d'effort CRM conçu en 1988, et d'une boucle geste synchrone à 1 kHz. La simulation de la partie sonore du modèle était toujours effectuée à 20 kHz. Le retour d'effort sur l'axe de friction restait limité et présentait une allure plutôt visqueuse [Flo08], mais plusieurs situations plus grossières typiques du phénomène de stick-slip (la « craie sur le tableau ») ont été réalisées grâce à cette configuration.

En 2001, une nouvelle version fut implantée en *ad hoc* sur l'architecture TELLURIS avec la version actuelle du clavier à retour d'effort ERGOS [Florens, 2003]. La puissance de cette architecture permettait l'implantation de la boucle geste à 3 kHz et de la boucle sonore à 30 kHz, ainsi que la modélisation d'un instrument à deux cordes. En outre, la qualité accrue du TGR ERGOS et l'amélioration de la fréquence de la boucle geste permettaient une plus grande plage de possibilités de jeu instrumental.

Le modèle développé dans le cadre de cette thèse (2007) est largement inspiré de la version développée en 2001. Nous allons donc dans un premier temps présenter le modèle de 2001 en détails avant de présenter la nouvelle version.

4.2.2 Description du modèle – version 2001

Le modèle de frottement d'archet comporte relativement peu de composants, et est essentiellement centré autour de l'implantation du phénomène de stick-slip qui a lieu dans l'interaction entre l'archet et la corde d'un violon. Ce phénomène consiste en la succession rapide de collages et de décollages entre l'archet et la corde : l'archet adhère à la corde (*stick*) et le mouvement de la corde et de l'archet est conjoint (la vitesse de mouvement différentiel est alors nulle). Un

brusque relâchement a lieu lorsque la force d'interaction entre l'archet et la corde dépasse une certaine intensité : la vitesse de mouvement différentiel n'est plus nulle, et la force d'interaction entre la corde et l'archet est décrite par une relation hyperbolique à la vitesse de mouvement. La corde glisse alors contre l'archet (*slip*) jusqu'à un nouveau collage. C'est ce phénomène répétitif de collages puis décollages microscopiques qui met en vibration la corde et lui donne son timbre caractéristique. En particulier, le seuil de décollement et l'intensité de la force d'interaction dépendent de la pression exercée par l'archet sur la corde.

Le modèle est bi-dimensionnel et est réduit à deux degrés de liberté juxtaposés : l'axe de glissement sur lequel s'effectue l'interaction avec la corde non linéaire de type stick-slip, et l'axe de mise en pression de l'archet contre la corde. Le stick 2D de la gamme ERGOS permet de coupler les axes de pression et de glissement dans un plan vertical à l'utilisateur (figure 6.19).

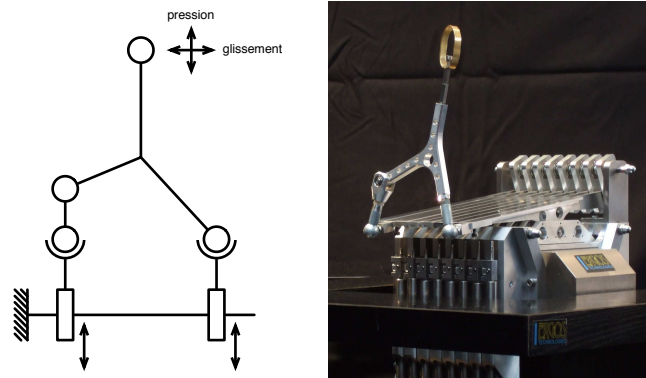


Figure 6.19. A gauche, la cinématique du stick 2D ERGOS monté sur le clavier ; à droite, une photo du clavier ERGOS équipé d'un stick 2D

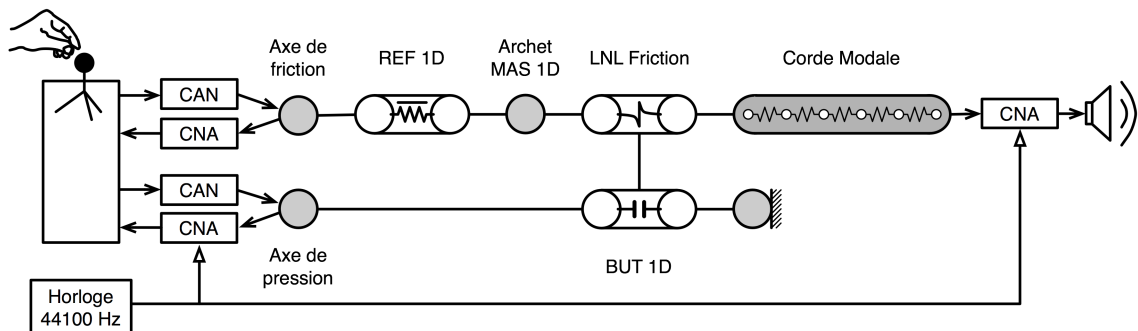


Figure 6.20. Le modèle CORDIS de frottement d'archet

L'axe de pression est l'axe vertical dans le plan frontal de l'utilisateur. Dans le modèle (figure 6.20), l'axe de pression est modélisé par une interaction de butée contre un module SOL. La rigidité de la butée représente la rigidité de la corde lors d'un effort de pression orthogonal à l'axe de la corde. La force d'interaction de cette butée représente la pression utilisée pour faire varier la force de friction dans le module de friction LNL. L'axe de glissement est l'axe horizontal dans le plan frontal de l'utilisateur. Dans le modèle (figure 6.20), l'axe de glissement comporte une

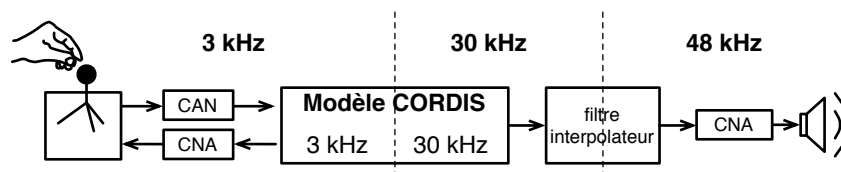


Figure 6.21. Diagramme fonctionnel du modèle de frottement d'archet – version 2001

caractéristique d'interaction non linéaire entre l'archet et la corde représentant le phénomène de stick-slip.

La corde est modélisée par un macro-module de corde modale (figure 6.20). Le modèle modal permet d'ajuster librement les fréquences de chaque mode vibratoire de la corde indépendamment, et ce de manière continue, ce qui permet en particulier d'obtenir une configuration quasi-harmonique même avec peu de modes. Ceci ne serait pas possible avec un chapelet discret de type CORDIS. De la même manière, le modèle de corde modale permet de choisir continûment sur toute la longueur de la corde la position du point de jeu (c'est-à-dire le point de couplage avec l'archet) et la position du point d'écoute.

Sur l'architecture TELLURIS, la fréquence de synchronisation avec les entrées/sorties du TGR est limitée à 3 kHz (chapitre 3 page 76). En outre, la partie du modèle responsable de la production des signaux sonores doit être calculée à une fréquence suffisamment haute pour que les phénomènes sonores modélisés ne soient pas déformés par un sous-échantillonnage. Dans la version 2001 du modèle de frottement d'archet, la partie du modèle en relation directe avec le TGR est calculée à 3 kHz et le calcul est synchrone avec la communication avec le TGR ; la partie sonore du modèle est calculée à 30 kHz, mais la synchronisation s'effectue à 3 kHz. Un filtre interpolateur restitue le signal sonore échantillonné à 48 kHz au format AES pour les sorties sonores (figure 6.21).

4.2.3 Description du modèle – version 2007

L'enjeu de l'implantation du modèle de frottement d'archet sur l'architecture ERGON_X était de valider le développement d'un modèle geste-son avec une seule boucle de simulation mono-fréquence à 44,1 kHz (figure 6.22). Jusqu'à présent en effet, la simulation en temps-réel d'objets sonores nécessitait d'utiliser une technique de simulation multi-fréquence, la synchronisation du simulateur avec les entrées/sorties s'effectuant à la fréquence de simulation « geste », c'est-à-dire la fréquence de simulation égale à la fréquence d'échantillonnage des signaux du TGR [Uhl, 1996, Giraud, 1999].

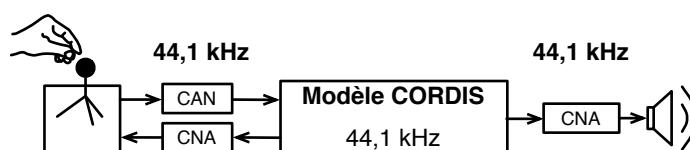


Figure 6.22. Diagramme fonctionnel du modèle de frottement d'archet – version 2007

Pour pouvoir augmenter la fréquence de simulation de la totalité du modèle jusqu'à 44,1 kHz, et pouvoir synchroniser le calcul de la simulation sur la communication avec le TGR, nous avons implanté le calcul de la simulation sur le HPU du simulateur ERGON_X. La majeure partie du travail d'implantation du modèle consistait donc à porter le modèle développé sous TELLURIS dans sa version 2001 sur la nouvelle plateforme ERGON_X. Les modules CORDIS nécessaires furent portés sur la nouvelle architecture, puis un travail important sur les paramètres algorithmiques du modèle fut mis en oeuvre afin de déterminer les plages de fonctionnement du modèle et de caractériser plusieurs situations de jeu intéressantes.

L'implantation du modèle de frottement d'archet en monofréquence présente plusieurs avantages par rapport à la version multi-fréquence de 2001. Premièrement, elle permet de synchroniser l'ensemble du modèle simulé sur la boucle des entrées/sorties du TGR. Le signal sonore est obtenu du modèle directement à la fréquence d'échantillonnage de 44,1 kHz. En réservant deux canaux de conversion N/A du HPU, il est alors possible d'alimenter deux haut-parleurs avec un signal sonore analogique de retard égal à la durée de 1 pas de simulation seulement (soit une période d'échantillonnage). Les conversions A/N et N/A de la communication avec le TGR et N/A pour les sorties sonores sont toutes exécutées simultanément. Par rapport à la version 2001, nous faisons l'économie du filtre interpolateur pour le signal sonore, et le délai geste-son est réduit au strict minimum¹.

D'autre part, la simulation multi-fréquence, même en situation synchrone est de manière générale source d'instabilités supplémentaires et rend le paramétrage du modèle plus délicat. À l'inverse, dans le cas du frottement d'archet, la simulation mono-fréquence permet d'obtenir plus facilement un comportement acceptable du modèle au point de couplage entre la structure vibrante (HF) et l'archet (BF). En effet, la modification de la fréquence de simulation modifie le comportement physique d'un modèle pour un même jeu de paramètres algorithmiques.

La configuration mono-fréquence rend possible l'étude des phénomènes haute-fréquence qui ont lieu au niveau de l'archet. Les phénomènes vibratoires ne sont plus filtrés que par les propriétés mécaniques du modèle d'archet (et, du point de vue de l'utilisateur, par les propriétés du TGR), alors que la réduction de la fréquence de simulation pour la partie geste dans les versions précédentes ne permettait pas l'émergence de phénomènes HF au niveau de l'archet. Le scénario expérimental mis en place à partir de ce modèle explore en particulier cette particularité du nouveau modèle.

Enfin, l'utilisation d'une fréquence de simulation élevée est rendue nécessaire non seulement en raison de la nature des signaux, mais aussi en raison de la nature des phénomènes mécaniques modélisés. En effet, le phénomène de stick-slip implique le calcul d'une non-linéarité dans le module de LNL (figure 6.20), et c'est une des raisons pour lesquelles la fréquence de simulation de la partie HF était portée à 30 kHz dans la version de 2001.

4.2.4 Intérêt de la scène et résultats

Charles Nichols a développé au CCRMA un violon virtuel haptique [Nichols, 2002], destiné à combler les lacunes du jeu de violon sur des instruments électroniques de type MIDI. Les

¹ Nous ne reviendrons pas plus en détails sur ce point, déjà discuté dans le cas du modèle de PebbleBox, page 140.

actionneurs sont des moteurs électriques Maxon combinés à un système de cabestans. La structure mécanique permet de prendre tous les degrés de liberté nécessaires dans le jeu de violon (mouvement latéral de l'archet, mais aussi déplacement longitudinal de l'archet pour le jeu *sul tasto*² ou *sul ponticello*³, jeu simultané sur plusieurs cordes, etc.). L'algorithme de simulation de la corde est basé sur le Synthesis Toolkit (STK) de Cook & Scavone, contrôlé par le signal de vitesse obtenu des capteurs de rotation. Une vibration aléatoirement variable de la corde est transmise au servomoteur de l'axe latéral pour simuler l'interaction archet-corde.

Stefania Serafin a développé une application permettant le jeu de violon virtuel à l'aide d'une tablette graphique [Serafin, 2004]. La force appliquée sur le stylet et la vitesse de mouvement sont transmises à un modèle de corde de type guide d'onde.

Les deux implantations de violon virtuel de C. Nichols et de S. Serafin sont basées sur un paradigme de type « signal » : les caractéristiques dynamiques du geste (vitesse, pression) sont utilisées pour le contrôle de l'algorithme de synthèse sonore, mais les principes de couplage mécanique ne sont pas respectés. Notre modèle, à l'inverse, permet un couplage mécanique fort entre la corde et l'archet, et permet de retrouver tous les modes de jeu permis par la situation instrumentale réelle (piqué, sauté, jeté... excitation des harmoniques naturels). Ainsi, ce modèle permet à l'instrumentiste de sentir la vibration de la corde simulée au bout des doigts, et de sentir la variation de la vibration lorsque le fondamental de la corde est modifié ou lorsque la corde est excitée sur un de ses harmoniques naturels.

Le couplage mécanique entre la corde et l'archet est particulièrement important afin de faire émerger les différents modes d'excitation de la corde à partir du geste d'archet seulement. Aussi, il est possible d'exciter la corde sur ses premiers harmoniques naturels simplement en adaptant le geste d'excitation afin de trouver le couple de vitesse et de pression adéquat, ou encore d'obtenir des sons de « crissements » par l'application d'une pression exagérée.

La différence majeure entre la situation virtuelle médiatisée et la situation instrumentale naturelle réside dans l'ambitus du mouvement d'archet. Dans la situation virtuelle, l'ambitus est limité par les caractéristiques mécaniques du stick 2D ERGOS, qui est d'environ 15 cm, alors que l'archet de violon (qui est le plus petit archet des instruments à corde) mesure entre 40 et 75 cm⁴. Cependant, la différence entre les ambitus de mouvements n'a été relevée que par les musiciens confirmés ayant une bonne connaissance théorique (compositeurs) ou pratique (instrumentistes professionnels ou bons amateurs) de l'instrument. Toutes les autres personnes ayant joué du présent violon virtuel, et n'ayant qu'une connaissance approximative des conditions de jeu avec un instrument réel n'ont pas relevé cette limitation sur l'ambitus des mouvements d'archet.

4.2.5 Le Scénario

Comme dans le cas du scénario « Pebble Box », nous avons procédé à une séance d'exploration du matériau expérimental, ainsi qu'à quelques premiers tests pilotes avec plusieurs des participants

²Jeu sur la touche : l'archet vient frotter le corde à hauteur de la touche.

³Jeu sur le chevalet

⁴Aux XVI^e et XVII^e siècles, la longueur des archets varie entre 40 et 75 cm. L'archet actuel se stabilise au cours du XIX^e siècle à une longueur de 73 à 74 cm.

à la séance de travail. Ont participé à la séance de travail les chercheurs du réseau Enactive Charlotte Magnusson (CERTEC, ULUND) et Sile O'Modhrain (Queens University Belfast), les compositeurs Peter Torvik, Periklis Douvitsas ainsi que plusieurs membres de l'ACROE et du laboratoire ICA.

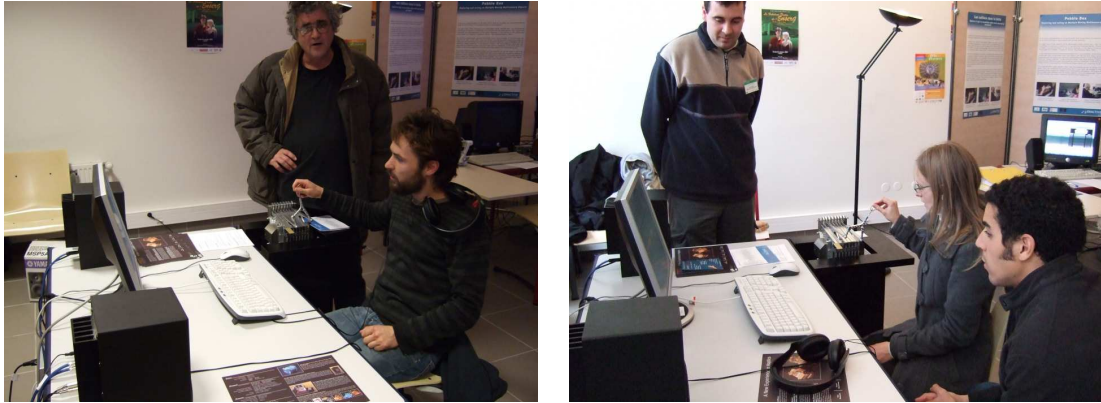


Figure 6.23. Le scénario emblématique « Ergotic Sounds », lors de l'exposition « Toucher le Futur » de la conférence Enactive/07 (Novembre 2007)

Pendant la première partie du travail exploratoire, les participants ont exploré la situation en l'absence de retour sonore. Ils ont décrit la situation en termes de propriétés de forme ou de surface : « un objet rugueux », « un objet sphérique », « des engrenages », « la sensation d'une fermeture éclair », et seul un participant a décrit une corde. Avec le retour sonore, l'identification de l'objet virtuel à un instrument de la famille des cordes frottées (violin, alto, violoncelle, contrebasse) fut immédiate et unanime. Les premiers résultats de ce travail exploratoire confirment l'hypothèse éactive : sans a priori de la part de l'utilisateur (ou en d'autres termes, de structures cognitives pré-existantes), l'objet inféré est différent selon les sensorialités impliquées dans la tâche. La construction de la représentation de l'objet est un processus évolutif qui se forme dynamiquement au cours de l'exploration.

Dans un deuxième temps, le travail exploratoire s'est focalisé sur les conditions nécessaires pour l'obtention de la continuité du son lorsque le mouvement de l'archet est inversé. Avec un instrument réel, cette tâche demande un long apprentissage et seuls les instrumentistes confirmés parviennent à une exécution sûre de ce geste. Il s'agit donc d'un geste instrumental demandant une grande expertise, et nous supposons que le couplage mécanique entre la main et la corde, par l'intermédiaire de l'archet, est une condition essentielle à sa réalisation.

Plusieurs niveaux de friction différents sont testés, pour deux hauteurs de cordes différentes. En particulier, la hauteur de corde joue un rôle important dans la sensation de « grain » perçue : pour la corde la plus grave (fondamental à 78 Hz), la sensation de frottement est nettement perceptible, alors que pour la corde plus aigue (fondamental à 110 Hz) cette sensation est plus diffuse. Ce dernier cas se rapproche en particulier de la situation du jeu de violon, où la sensation de friction de l'archet contre la corde n'est pas perceptible par le geste.

Les conditions de réussite de la tâche sont observées objectivement (les signaux de position de la corde, de position de l'archet, de la pression de l'archet appliquée sur la force et de la

force sur l'axe de friction sont observés à l'oscilloscope), et subjectivement par les personnes de l'assistance.

Le temps d'exploration des différentes conditions expérimentales et de test de la tâche varie entre 15 et 30 minutes selon les participants. Tous les participants sans exception parviennent au moins une fois à réussir la tâche de continuité du son. Les temps d'apprentissage et les aptitudes à la réussite de la tâche sont variables suivant les individus, mais d'une manière générale, les conditions de friction déclarées par les participants et permettant objectivement d'atteindre la meilleure performance sont le cas nominal et le cas présentant des forces de friction plus faibles que le cas nominal. Lorsque la friction est absente ou exagérée, tous les participants montrent beaucoup plus de difficultés à réussir la tâche. Cependant, certains participants parviennent à obtenir la continuité du son lorsque la friction est nulle, après un apprentissage avec le cas nominal ou avec le cas à friction faible.

Ce travail préparatoire permet de sélectionner plus précisément les situations qui seront présentées aux visiteurs du Scénario : il ne faut pas que le parcours du scénario interactif soit trop long par une surcharge du nombre de situations présentées, mais le parcours doit tout de même pouvoir donner une vision d'ensemble relativement complète.

Le scénario est divisé en deux parties. La première partie est composée de deux « expériences » où le visiteur explore la situation instrumentale du jeu sur la corde virtuelle. La situation est d'abord présentée sans retour sonore (expérience 1), puis présentée en présence de retour sonore (expérience 2), dans le cas présentant les forces de friction nominales. La deuxième partie du scénario (expérience 3) propose au visiteur d'essayer d'obtenir un son continu à l'inversion du mouvement de l'archet, en testant plusieurs cas de friction différents : friction faible, friction nominale, et friction exagérée. A chacune des étapes de la première comme de la deuxième partie, le visiteur est amené à décrire par écrit ses sensations ou ses impressions. Les retours subjectifs des visiteurs et les statistiques enregistrées sont rapportés en Annexe C de ce mémoire.

Pendant les cinq jours de l'exposition « Toucher le Futur », le scénario « Ergotic Sounds » a été visité par 200 personnes environ, parmi lesquelles respectivement 128, 121 et 123 personnes ont passé plus de 30 s par expérience pour un temps moyen de parcours total du scénario de 6 minutes 30 s environ.

L'analyse des résultats se base sur les enregistrements des questionnaires remplis par les visiteurs, présentés en Annexe C.

De même que pendant l'expérience pilote, dans la première expérience les visiteurs décrivent en l'absence de son des objets en relation avec la friction (« *a rubber surface* », « *frottement* », « *sensation de toucher une matière rugueuse et vibrante* »,) et décrivent les propriétés géométriques de surface (« *segment of a circle as virtual wall* », « *surface of a ball* »). L'identification n'est pas toujours possible ou évidente (« *étrange et bizarre je n'arrive pas à deviner* », « *first time nothing... half of sphere* »). Dans la deuxième expérience, l'identification d'un instrument à corde frottée est unanime (« *un violoncelle ? une contrebasse ?* », « *string of a cello* », « *un instrument à cordes frottées* », etc.), et certains visiteurs mentionnent l'aspect particulièrement évocateur du son (« *une contrebasse — surtout pour le son —* »).

Plusieurs visiteurs ont attribué dès la première expérience une sensation de « corde », de « violon ». Il est à supposer que cela est dû aux conditions de visite, différentes de celles d'une expérience

conventionnelle, où les visiteurs ont pu voir d'autres personnes parcourir le scénario, ou entendre au loin les sons provenant de l'espace du scénario « Ergotic Sounds » préalablement à leur parcours du scénario, ce qui les a incités à décrire une corde frottée dès la première expérience.

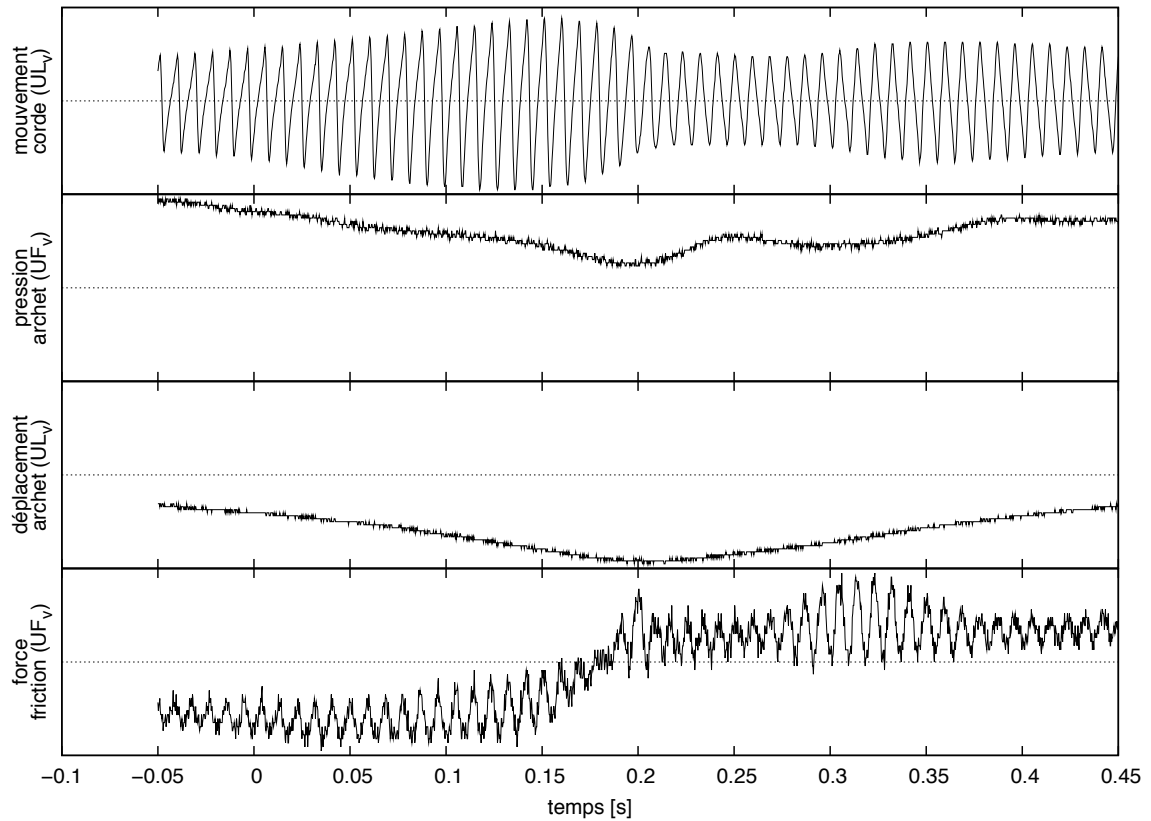


Figure 6.24. Observation objective de la tâche d'inversion du mouvement d'archet en maintenant la continuité du son. Les signaux sont extraits du modèle (unités : UL_v Unité de Longueur virtuelle, UF_v Unité de Force virtuelle) L'oscillation de la corde est maintenue pendant l'inversion du mouvement de l'archet. On peut observer que la pression appliquée ne s'annule pas pendant l'inversion, et que la force de friction change de signe au moment de l'inversion du mouvement.

Cette expérience est particulièrement significative de la « signature » que porte le son d'un instrument à corde frottée. La signature est significativement évocatrice, puisque si les visiteurs décrivent un objet différent — parfois éloigné — d'une corde frottée en l'absence de retour sonore, l'ajout du retour sonore les invite à se représenter un violon ou un violoncelle. L'identification d'un instrument à corde frottée en présence de son est unanime. À l'inverse, et de l'aveu même d'instrumentistes de haut niveau, la sensation de frottement de l'archet contre la corde n'est pas présente dans le jeu l'instrument réel pour le violon ou l'alto, et n'a que peu d'importance dans le jeu de violoncelle ou de contrebasse. En d'autres termes, la friction corde-archet est une caractéristique particulière de la famille instrumentale des cordes frottées qui leur donne ce son caractéristique, mais ne participe pas ou seulement peu à des conditions adéquates de jeu instrumental.

À ce stade de la réflexion, il serait tentant de conclure que c'est le son uniquement qui porte

la signature d'un instrument à corde frottée. En effet, l'identification de la situation instrumentale virtuelle à un instrument à corde frottées se fait en présence du son seulement, et cette identification est quasiment unanime parmi toutes les personnes ayant pu explorer le scénario. Inversement, en l'absence de son, la situation instrumentale n'est que rarement associée à l'interaction entre un archet et une corde.

De plus il est à supposer que la sensation de frottement n'est que peu pertinente dans le jeu de violon, même si la majorité des participants portent une préférence au cas de frottement « nominal », où les forces de friction sont supérieures au cas de friction faible. Cependant, le son des instruments à cordes frottées a ceci de particulier que la signature du geste même est inscrite dans le son. C'est une des raisons pour lesquelles les modèles de synthèse sonore en temps différé (i.e. sans excitation gestuelle en temps réel par un opérateur humain) se montrent encore peu convaincants à l'heure actuelle.

Il serait intéressant de continuer les investigations sur la signature du geste qui se trouve inscrite au cœur même du son d'un instrument à corde frottée. À ce titre, le modèle de corde frottée qui a été implanté sur le simulateur ERGON_X permet une exploration fine *in situ* du jeu des instruments à corde frottée et du phénomène de stick-slip. Il s'agit d'un outil pertinent et solide pour la construction d'une nouvelle démarche expérimentale sur la base des premières hypothèses avancées.

Pour plus de 50 % des visiteurs qui se sont exprimés, le cas de friction nominale était celui qui permettait le mieux la réussite de la tâche de continuité du son, et était aussi celui qui était le plus agréable à jouer. Le cas à faible friction a reçu 30 % des préférences, contre moins de 10 % pour le cas de friction exagérée. 9 % des visiteurs qui se sont exprimé ont jugé qu'aucune des frictions présentées ne permettait de réussir la tâche ; on peut supposer que ces personnes n'ont pas estimé avoir réussi à effectuer la tâche demandée, ce qui signifie en d'autres termes, que 90 % des visiteurs qui se sont exprimés ont jugé avoir réussi la tâche.

À la différence de l'expérience pilote où les observations subjective (d'autres expérimentateurs) et objective (à l'oscilloscope) permettaient de valider la réussite de la tâche, dans les conditions de l'exposition grand public, seuls les retours entrés dans l'ordinateur par les visiteurs étaient disponibles pour l'analyse de l'expérience. Les retours apportés par les visiteurs constituent un point de vue partiel sur les retours de visites puisque la participation aux formulaires de réponses n'était pas imposé lors de la visite. Cependant, les retours obtenus (présentés en annexe C) confirment les préférences exprimées par les participants à l'expérience pilote préliminaire.

5 Conclusion

Ce chapitre est centré sur les usages du simulateur ERGON_X. Ce travail a été abordé sous plusieurs axes complémentaires.

Le premier axe consiste en la validation de la plateforme de simulation et en la démonstration de ses capacités techniques. Dans ce but un modèle de démonstration a été réalisé, le « E », qui permet d'exploiter pleinement les capacités de la technologie à retour d'effort de la gamme ERGOS. L'utilisation d'une fréquence de simulation de 40 kHz a permis :

- De valider l'architecture ERGON_X pour la simulation de phénomènes dynamiques à grande bande passante ;
- D'élargir la plage de stabilité numérique du modèle, et d'accéder à des nouvelles valeurs de rigidité numérique du modèle simulé ;
- D'exploiter pleinement la bande passante du système ERGOS.

Ce modèle permet d'exploiter les performances dynamiques des systèmes ERGOS, et de démontrer leurs excellentes capacités, tant dans les mouvements libres, dans la simulation de contacts rigides, que dans la simulation d'objets de grande précision (que l'on peut expérimenter sur les arêtes convexes).

Le second axe consiste en l'utilisation de la plateforme ERGOS selon les critères conventionnels de la psychophysique. Nous avons montré que certaines caractéristiques technologiques sont nécessaires si l'on souhaite utiliser le retour d'effort pour l'expérimentation psychophysique, et que le simulateur ERGON_X possède ces qualités. Les deux expériences réalisées, l'une sur une tâche de percussion d'un objet sonore, l'autre sur la compression d'un objet déformable, mettent en avant l'importance des propriétés physiques de l'objet manipulé.

L'expérience sur la perception de l'élasticité, outre ses résultats expérimentaux, valide le concept de plateforme de simulation interactive comme un outil pour l'expérimentation psychophysique. En particulier, l'observation in situ, par l'enregistrement de traces, a permis d'analyser plus finement la situation expérimentale [Couroussé *et al.*, 2006b].

L'expérience sur la percussion d'un objet sonore montre l'intérêt de la simulation pour la construction d'un environnement expérimental. Dans ce contexte, il est possible de modifier une propriété d'un objet indépendamment —ici, la raideur de l'objet percuté sans modifier son timbre— alors que ce n'est pas possible dans le monde réel.

Les modèles implantés dans le cadre de cette thèse permettent d'ouvrir de nouvelles perspectives dans le champ de la simulation de matériaux crédibles ou belivable, et en particulier sur la notion de *présence* dans les réalités virtuelles. Par exemple, le modèle de « E » permet l'ouverture du domaine de la recherche haptique à la simulation de tout un nouvel espace de comportements mécaniques rigides. En guise d'anecdote, citons la remarque de Abderrahmane Kheddar, professeur à l'université d'Evry et reconnu comme parmi les plus grands spécialistes par la communauté de la recherche haptique, lors de sa première manipulation du modèle « E » : à peine sa main se fut-elle posée sur le stick ERGOS qu'il demanda si les contacts qu'il ressentait correspondaient aux butées mécaniques du système, ou bien aux contacts simulés ! De la même manière, la version 2001 du modèle de frottement d'archet présentait une avancée importante dans la simulation du jeu de violon puisqu'il permettait de retrouver tous les modes de jeu de la situation naturelle grâce à une restitution du couplage mécanique qui a lieu entre la main de l'instrumentiste et la corde de violon par l'intermédiaire de l'archet et de l'interaction collophanée. Le modèle de frottement d'archet développé dans le cadre de cette thèse constitue une étape supplémentaire dans l'émergence de la notion d'objet et de sa présence dans une situation instrumentale médiatisée.

Le troisième axe consiste donc au développement de nouveaux modèles susceptibles de faire émerger la sensation de présence, et en l'ouverture des démarches expérimentales vers des approches alternatives à l'expérimentation psychophysique classique. Dans ce chapitre nous avons

présenté la mise en oeuvre de scénarios interactifs à partir desquels des pistes pour l'exploration expérimentale de ces nouvelles situations médiatisées ont été proposées. Dans ce cadre ont été mises en évidence des stratégies dynamiques par lesquelles un être humain prend connaissance d'un objet inconnu et s'en construit une représentation. Ces stratégies sont particulièrement illustratives des théories de l'énaction.

Conclusion

L'objectif de cette thèse est multiple. Dans le cadre du réseau d'excellence Enactive Interfaces, il s'agit dans un premier temps de contribuer à une analyse exhaustive de l'état de l'art pour la simulation instrumentale multisensorielle. Cette approche est pluridisciplinaire : sciences de l'ingénieur (mécatronique, informatique...), sciences de l'homme (psychophysique, cognition, philosophie) et arts. Les trois premiers chapitres constituent une partie de notre collaboration au travail d'analyse effectué dans le cadre du réseau Enactive Interfaces. Le chapitre 2 en particulier propose un état de l'art de la technologie pour le retour d'effort par le biais de l'analyse fonctionnelle du couplage mécanique entre l'homme et l'objet manipulé.

* * *

Les chapitres 3 et 4 sont le résultat de l'analyse des différentes approches technologiques que l'on trouve dans les plateformes pour la simulation instrumentale multisensorielle. Le chapitre 3 se focalise sur le point de jonction entre les deux parties d'une architecture de simulation multisensorielle que sont la commande du TGR à un bas niveau d'une part, les architectures pour la simulation par modèle physique d'autre part. Il s'agit encore aujourd'hui d'un point peu abordé dans la littérature, où pourtant se trouvent concentrés plusieurs points critiques d'une architecture pour la simulation multisensorielle. Le chapitre 4 présente le HPU (*Haptic Processor Unit*) ; il s'agit du composant matériel qui permet la connexion d'un Transducteur Gestuel Rétroactif au reste de l'architecture du simulateur. Ce composant doit pouvoir exécuter un processus de simulation en temps-réel, synchronisé sur les flux de communication avec le TGR ; il s'agit donc d'un composant de connexion doté de capacités de calcul évoluées.

Derrière l'acronyme « HPU » se cachent à la fois un concept et un composant ; le chapitre 4 est consacré au détail des propriétés et des caractéristiques du concept HPU, et le chapitre 5 décrit la mise en œuvre du composant HPU au sein d'une nouvelle architecture pour la simulation instrumentale multisensorielle. Au cours de ce travail, le HPU est en effet né conjointement du travail d'analyse effectué sur les architectures de simulation, en particulier dans le cadre du réseau d'excellence Enactive [Jean-Loup Florens, 2004, Couroussé et Florens, 2007b, Florens et Couroussé, 2007], et des étapes successives des développements techniques qui ont abouti à la réalisation du simulateur ERGON_X.

* * *

Le simulateur ERGON_X est un nouveau simulateur multisensoriel qui fut réalisé dans le cadre de cette thèse. Il est présenté dans le chapitre 5. ERGON_X possède d'excellentes caractéristiques

en termes de réactivité, de compacité et de transportabilité. Sa mise en œuvre constitue la mise en application du concept de HPU. Le HPU du simulateur ERGON_X embarque un DSP C6711 qui permet le calcul flottant, et sur lequel est exécuté le processus de simulation. Il s'agit d'une unité de calcul relativement modeste au regard des processeurs grand public actuels, mais qui a l'avantage de se trouver à proximité immédiate des convertisseurs A/N et N/A du HPU. Il fut ainsi possible d'obtenir des boucles de simulation synchrones avec une période de simulation extrêmement élevée, jusqu'à plus de 44,1 kHz. La réalisation d'un simulateur adapté à la simulation de grands modèles est actuellement en cours à l'ACROE, dans le cadre de la thèse de J. Castet. Ce simulateur intègre le composant HPU réalisé dans le cadre de cette thèse. À terme, le groupe de recherche ACROE & ICA disposera d'une palette d'outils complète et compétitive pour la simulation instrumentale multisensorielle.

Le simulateur ERGON_X est compact et transportable : il se compose d'un TGR de la gamme ERGOS, d'un ordinateur grand public et d'un HPU sous la forme d'une carte d'acquisition TORO de Innovative Integration. Une nouvelle version de l'électronique ERGOS fut développée en 2006–2007 afin d'améliorer en particulier sa compacité et sa portabilité, ce qui a permis de créer un simulateur compact et facilement transportable. En juillet 2006, le simulateur ERGON_X fut présenté pour la première fois dans le salon des exposants industriels de la conférence EuroHaptics 2006, à Paris. Fin 2006, l'ACROE a réalisé un modèle de démonstration multisensorielle de nanomanipulation. Ce modèle a été conçu pour une exposition grand public sur les nanotechnologies. D'une durée totale de cinq ans, cette exposition est itinérante — CCSTI de Grenoble, La Villette Paris, Genève, Bordeaux, etc. — et reçoit environ 1000 visiteurs/mois. En 2007, quatre simulateurs ERGON_X ont été réalisés. Ils ont été présentés à l'exposition grand public « Toucher le Futur » de la conférence scientifique Enactive/07, organisée par le laboratoire ICA et l'ACROE en Novembre 2007 à Grenoble. Six scénarios interactifs y ont été présentés : « Real to Virtual Physical Cooperation », « Hands in the nanoworld », « PebbleBox », le « E », « Ergotic Sounds », « Shapes and Contour ». À ce jour, six simulateurs ERGON_X ont été réalisés : trois se trouvent au laboratoire, deux ont été vendus au laboratoire SPCL de l'université de McGill à Montréal, le sixième se trouvant dans l'exposition itinérante sur les nanotechnologies (achat du CCSTI de Grenoble). Le simulateur ERGON_X est donc un outil qui se trouve au cœur des problématiques et des usages dans le cadre de la recherche sur la simulation instrumentale multisensorielle.

Le simulateur ERGON_X présente une excellente réactivité. La fréquence de simulation en temps réel et complètement synchrone a pu ainsi être portée jusqu'à 44,1 kHz. Cette fréquence permet de simuler dans un même modèle et à fréquence de simulation unique les déformations acoustico-mécaniques d'un objet sonore, et de les manipuler directement par l'intermédiaire d'un TGR. L'utilisation d'une fréquence unique pour la simulation d'un modèle d'objet sonore est particulièrement importante pour la modélisation d'instruments de musique, dans lesquels les vibrations sonores peuvent se communiquer jusqu'à la partie mécanique en contact avec l'instrumentiste. Ces vibrations sonores font alors partie du « ressenti » de l'instrument et semblent nécessaires pour certaines techniques de jeu. Dans le scénario « Ergotic Sounds », il a été en effet montré que, dans le cas du frottement d'archet, le ressenti des vibrations de la corde dans l'archet était qualifié comme un paramètre de jeu important.

La modularité est une notion fondatrice dans les développements scientifiques et techniques du groupe de recherche ACROE & ICA. L'architecture de ERGON_X fut conçue en ce sens, et permet

la connexion avec les autres outils de recherche de l'ACROE. Le nouveau simulateur TELLURIS en cours de réalisation dans la thèse de J. Castet, intègre le HPU pour la réalisation de la connexion avec les TGR de la gamme ERGOS.

* * *

Le travail de développement réalisé dans le cadre de cette thèse a permis de poser les bases technologiques et conceptuelles du simulateur ERGON_X et du composant HPU. La réalisation technique a atteint un stade suffisamment achevé pour permettre de présenter des démonstrations autonomes au grand public (i.e. sans assistance technique d'un utilisateur). Dans l'état actuel, les travaux de développement ont été essentiellement orientés vers le support de la communication avec le TGR et le développement d'un squelette efficace pour le processus simulation, supporté par le HPU.

Un modeleur supportant d'une manière générique le formalisme CORDIS demande à être implanté. En effet, le travail de modélisation s'effectue actuellement à partir du développement de modèles ad hoc, ce qui impose au concepteur du modèle la connaissance intime de l'architecture de simulation et l'écriture de nouveau code source, c'est-à-dire endosser le rôle de développeur. L'interface utilisateur est elle aussi rudimentaire. Le simulateur ERGON_X doit pouvoir prétendre à une utilisation large, à la fois dans le laboratoire mais aussi à l'extérieur — par la commercialisation ou par le biais de collaborations —, par des utilisateurs de profils variés — et pas seulement par des informaticiens. Dans ce but, la prochaine étape de développement devra améliorer les composantes modeleur et interface de ERGON_X : cela suppose le support d'un noyau de simulation CORDIS-ANIMA générique mais modulaire, et l'existence d'une interface de modélisation conviviale.

Utiliser un noyau de simulation générique pour CORDIS-ANIMA signifie avoir à sa disposition, en tant qu'utilisateur, un modeleur qui permette le développement de tous les modèles supportés par le formalisme du langage de modélisation. Dans le cadre de la simulation en temps réel, le développement d'un noyau de simulation générique n'est pas une tâche aisée. En comparaison avec le développement de code de simulation ad hoc, il suppose l'intégration de mécanismes d'indirection supplémentaires nécessaires à la gestion de la genericité et de la modularité. Cependant, les performances d'exécution temporelles du noyau de simulation sont critiques pour le temps réel, et l'ajout d'un niveau de genericité supplémentaire peut se montrer coûteux au regard des performances de simulation actuelles. Le développement d'un noyau de simulation générique devra donc trouver le bon compromis entre efficacité calculatoire et genericité du noyau de simulation. Il existe cependant une voie d'issue qui permettrait d'obtenir un noyau de simulation générique sans introduire de niveau d'indirection supplémentaire : la génération de code en ligne. Il est en effet possible de faire générer du code ad hoc par un programme de haut niveau, la structure du code généré dépendant de la topologie du modèle décidé par l'utilisateur. Ce code, optimisé pour le modèle à simuler, est ensuite compilé, chargé puis exécuté sur l'architecture de simulation sous le contrôle du processus superviseur. Cette solution nécessite le développement d'un logiciel capable de générer du code source, et que ce code soit aussi efficace que le code ad hoc écrit par un programmeur au fait des subtilités de l'architecture de simulation. Il sera intéressant d'étudier sérieusement cette voie, car elle semble une issue prometteuse qui permettrait de résoudre la perte de performance induite par l'utilisation d'un noyau de simulation générique. Dans tous les cas, le

développement d'un modèle ad hoc en réécrivant le code de la boucle de simulation doit rester une alternative possible à l'utilisation de l'interface générique. L'expérience des usages du simulateur TELLURIS montre en effet que, pour certains modèles, le développement ad hoc s'avère être la meilleure issue lorsque les performances requises sont proches des limites matérielles du simulateur. Plus généralement, le remaniement du squelette de simulation ou du code de bas niveau de la simulation doit rester une option possible dans les usages du simulateur. Il s'agit d'une étape incontournable dans une problématique de recherche sur les architectures pour la simulation en temps réel et de développement des technologies pour les Transducteurs Gestuels Rétroactifs.

Le groupe de recherche ACROE & ICA a beaucoup investi ces cinq dernières années dans le portage des interfaces de modélisation GENESIS et MIMESIS. GENESIS est un environnement de modélisation physique pour la synthèse sonore, et MIMESIS un environnement de modélisation physique pour l'image animée. Il s'agit d'un projet de développement important qui a pour objectif la refonte complète des interfaces et des outils de modélisation, ainsi que leur portage sur les systèmes d'exploitation les plus utilisés par le grand public (Linux, Apple OSX et Windows). L'équipe en charge de ce projet a développé un important savoir-faire sur la conception d'interfaces d'un point de vue technique (développement logiciel) et d'un point de vue utilisateur (IHM, ergonomie, usabilité, etc.). Le développement d'interfaces de modélisation pour le temps réel, en particulier dans le cadre du projet ERGON_X, doit pouvoir bénéficier de ce savoir faire. La collaboration avec cette équipe sera nécessaire lorsque le développement d'une interface utilisateur sera envisagé pour ERGON_X.

* * *

Le simulateur ERGON_X est un outil de recherche sur l'interaction instrumentale, à partir duquel il est possible de définir deux orientations. La première est l'utilisation de ERGON_X comme un outil pour l'expérimentation sur la perception humaine. Le chapitre 6 présente l'utilisation de ERGON_X à la fois pour l'expérimentation psychophysique conventionnelle, mais aussi pour la mise en œuvre de nouvelles méthodes expérimentales. Si nous reprenons la terminologie introduite dans notre analyse des TGR (chapitre 2), les deux expériences sur la percussion d'un objet sonore et la perception de l'élasticité se situent dans l'approche « homme », même si l'étude se focalise sur les relations entre les propriétés physico-mécaniques de l'objet manipulé, la « performance » du geste et la perception. En augmentant la réactivité générale du simulateur ERGON_X tout en garantissant des conditions d'exécution temps-réel, une forte sensation de présence de l'objet simulé émerge des situations instrumentales développées : les modèles présentés dans le chapitre 6 permettent le développement d'une instrumentalité forte, et c'est à partir de cette propriété qu'ont pu être envisagées de nouvelles méthodes d'expérimentation sur la perception humaine. Cette nouvelle démarche, exemplaire du travail mené sur l'enaction dans le cadre du projet Enactive Interfaces, s'inscrit dans une approche « interaction » du couplage homme-objet. Les résultats concernent l'homme en situation d'interaction avec son environnement, plutôt que l'homme et ses capacités perceptives ou motrices comme un objet d'observation fermé.

Dans le cadre des réalités virtuelles et en particulier des technologies pour le retour d'effort, les capacités de la perception humaine sont souvent utilisées comme un critère permettant de définir les caractéristiques et les performances des nouvelles technologies développées. À l'inverse, les nouveaux outils technologiques développés à partir de ces critères servent à leur tour d'outil pour

l'expérimentation et la production de connaissances sur la perception humaine. Il s'agit d'une situation cyclique potentiellement dangereuse, où la connaissance de l'humain sert de référent à la technologie qui à son tour est utilisée comme référent pour la recherche sur la connaissance de l'humain. Dans le cadre des développements technologiques entrepris par le groupe de recherche ACROE & ICA pour la simulation instrumentale multisensorielle, et en particulier dans le cadre du développement du simulateur ERGON_X, nous avons volontairement fait le choix de nous détacher des critères technologiques de performance communément admis par la communauté « haptique », en particulier l'utilisation d'une fréquence de simulation pour la boucle gestuelle à retour d'effort de 1 kHz. ERGON_X est un outil qui permet d'exploiter pleinement les excellentes performances dynamiques des TGR de la gamme ERGOS, et d'un simulateur qui se rapproche du simulateur idéal — à temps continu — en termes de bande passante et en termes de réactivité.

La deuxième orientation de recherche consiste alors à questionner la première orientation, c'est-à-dire à analyser dans quelle mesure la technologie du retour d'effort et des réalités virtuelles peut être un outil pertinent pour l'étude de la perception humaine, et inversement, dans quelle mesure les connaissances sur la perception humaine apportent une mesure pertinente pour les développements technologiques. Cette démarche est encore minoritaire dans les travaux de recherche actuels.

ERGON_X est le candidat idéal pour entreprendre cette démarche. Il possède des caractéristiques techniques, notamment en matière de réactivité, qui le placent dans une situation de « référence » par rapport aux autres solutions technologiques disponibles aujourd'hui. Nous avons alors à notre disposition un outil modulable qui permet de restituer les conditions de fonctionnement d'autres systèmes, et d'analyser dans quelle mesure les caractéristiques technologiques des systèmes actuels satisfont aux exigences des usages et de leurs usagers. Cette nouvelle orientation pour la recherche sur la simulation instrumentale multisensorielle est susceptible d'apporter de nouvelles réponses déterminantes à la fois pour les usages de la technologie de retour d'effort et de la simulation instrumentale multisensorielle, mais aussi pour les orientations de recherche à privilégier pour les futurs développements technologiques.

Bibliographie

- [DBL, 2005] (2005). *World Haptics Conference (WHC 2005), 18-20 March 2005, Pisa, Italy*. IEEE Computer Society.
- [sen, 2007] (2007). Sensable technologies, <http://www.sensable.com>.
- [erg, 2008] (2008). Ergos Technologies, <http://ergos-technologies.com>.
- [imm, 2008] (2008). Immersion corporation, <http://www.immersion.com/>.
- [fcs, 2008] (2008). The Haptic Master, <http://www.fcs-cs.com/robotics/products/hapticmaster>.
- [Adachi *et al.*, 1995] Adachi, Y., Kumano, T. et Ogino, K. (1995). Intermediate representation for stiff virtual objects. In *Proceedings of the Virtual Reality Annual International Symposium (VRAIS '95)*, pages 203–210, Washington, DC, USA. IEEE Computer Society.
- [Akahane *et al.*, 2005] Akahane, K., Hasegawa, S., Koike, Y. et Sato, M. (18-20 March 2005). A development of high definition haptic controller. *Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint*, pages 576–577.
- [Akahane *et al.*, 2006] Akahane, K., Hasegawa, S., Koike, Y. et Sato, M. (Nov. 2006). A proposal of a high definition haptic rendering for stability and fidelity. *Artificial Reality and Telexistence—Workshops, 2006. ICAT '06. 16th International Conference on*, pages 162–167.
- [Aschersleben, 2002] Aschersleben, G. (2002). Temporal control of movements in sensorimotor synchronization. *Brain and Cognition*, 1(48):66–79.
- [Aschersleben *et al.*, 2000] Aschersleben, G., Gehrke, J. et Prinz, W. (2000). Tapping with peripheral nerve block - a role for tactile feedback in the timing of movements. *Experimental Brain Research*, 136(3):331–339.
- [Astley et Hayward, 1998] Astley, O. R. et Hayward, V. (1998). Multirate haptic simulation achieved by coupling finite element meshes through Norton equivalents. *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, 2:989–994.
- [Batter et Brooks, 1971] Batter, J. J. et Brooks, F. P. (1971). GROPE-I : A computer display to the sense of feel. In *IFIP congress 71*, pages 188–192 (TA-4), Ljubljana, Slovenia.
- [Bergamasco *et al.*, 1994] Bergamasco, M., Allotta, B., Bosio, L., Ferretti, L., Parrini, G., Prisco, G. M., Salsedo, F. et Sartini, G. (1994). An arm exoskeleton system for teleoperation and virtual environments applications. In *IEEE International Conference on Robotics and Automation*, pages 1449–1454, San Diego, CA.
- [Berkelman *et al.*, 1996] Berkelman, P. J., Butler, Z. J. et Hollis, R. L. (1996). Design of a hemispherical magnetic levitation haptic interface device. In *ASME International Mechanical Engineering Congress and Exposition*, pages 483–448, Atlanta.

- [Berkelman et Hollis, 1997] Berkelman, P. J. et Hollis, R. L. (1997). Dynamic performance of a magnetic levitation haptic device. *In Proceedings of SPIE Telemanipulation & Telepresence*.
- [Bi et al., 2005] Bi, D., Zhang, J. et Wang, G. L. (2005). Fuzzy virtual coupling design for high performance haptic display. *In Wang, L. et Jin, Y., éditeurs : FSKD (1)*, volume 3613 de *Lecture Notes in Computer Science*, pages 942–949, Changsha, China. Springer.
- [Bongers, 1994] Bongers, B. (1994). The use of active tactile and force feedback in timbre controlling musical instruments. *In Proceedings of the International Computer Music Conference (ICMC)*, rhus, Denmark.
- [Borro et al., 2004] Borro, D., Savall, J., Amundarain, A., Gil, J. J., Garcia-Alonso, A. et Matey, L. (2004). A large haptic device for aircraft engine maintainability. *IEEE Comput. Graph. Appl.*, 24(6):70–74.
- [Bouzit et al., 2002a] Bouzit, M., Burdea, G. C., Popescu, G. et Boian, R. (2002a). The Rutgers Master II–New Design Force Feedback Glove. *IEEE/ASME Transactions on Mechatronics*, 7(2):256–263.
- [Bouzit et al., 2002b] Bouzit, M., Popescu, G., Burdea, G. C. et Boian, R. (2002b). The Rutgers Master II-ND Force Feedback Glove. *In Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, page 145. IEEE Computer Society.
- [Burdea, 1996] Burdea, G. C. (1996). *Force and touch feedback for virtual reality*. John Wiley & Sons, Inc.
- [Burdea, 1999] Burdea, G. C. (1999). Invited review : The synergy between virtual reality and robotics. *IEEE Transactions on Robotics and Automation*, 15(3):400–410, 415.
- [Burdea et Coiffet, 1993] Burdea, G. C. et Coiffet, P. (1993). *La réalité virtuelle*. Hermès Paris.
- [Burdea et Speeter, 1989] Burdea, G. C. et Speeter, T. (1989). Portable dextrous force feedback master for robot telemanipulation. *In Proceedings 1989 NASA Conference on Space Telerobotics*, volume 2, pages 153–161, Pasadena CA.
- [Cadoz, 1994] Cadoz, C. (1994). Le geste, canal de communication homme/machine. la communication ”instrumentale”. *Technique et Science de l’Information*, 13(1):31–61.
- [Cadoz, 2004] Cadoz, C. (2004). Enactive interfaces? *Enactive Virtual Workshop*. <http://www.interdisciplines.org/enaction>.
- [Cadoz et al., 1988] Cadoz, C., Lisowski, L. et Florens, J.-L. (1988). Clavier rétroactif modulaire et actionneur modulaire plat – Brevet Français. Déposé le 13/10/1988. France n°88 14064, US n°07/420 242 - Europe n°0 365 441 - Titulaire : ACROE.
- [Cadoz et al., 1984] Cadoz, C., Luciani, A. et Florens, J.-L. (1984). Responsive input devices and sound synthesis by simulation of instrumental mechanisms : The Cordis System. *Computer Music Journal*, 8(3):60–73.
- [Cadoz et al., 1990] Cadoz, C., Luciani, A. et Florens, J.-L. (1990). CORDIS-ANIMA : système de modélisation et de simulation d’instruments et d’objets physiques pour la création musicale et l’image animée. *In Modèles physiques, création musicale et ordinateur*, volume II, pages 583–626. ACROE.

- [Cadoz *et al.*, 1982] Cadoz, C., Luciani, A., Florens, J.-L. et Dars-Berbeyran, T. (1982). The control channels of instrumental playing in computer music - real time in computer music, incidence of basic models. *In Proceedings of ICMC 1982*, Venice.
- [Campion et Hayward, 2005] Campion, G. et Hayward, V. (2005). Fundamental limits in the rendering of virtual haptic textures. *Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint*, pages 263–270.
- [Castagné *et al.*, 2005] Castagné, N., Florens, J.-L., Luciani, A. et Kergadallan, G. (2005). Computer platforms for hard-real time and high quality ergotic multisensory systems. *In Camurri, A. et Frisoli, A., éditeurs : Proceedings of Enactive05, 2nd International Conference on Enactive Interfaces*, Genoa, Italy. Enactive Interfaces.
- [Castet *et al.*, 2007] Castet, J., Couroussé, D., Florens, J.-L. et Luciani, A. (2007). A real-time simulator for virtual reality conceived around haptic hard constraints. *In Luciani, A. et Cadoz, C., éditeurs : Proceedings of Enactive/07, 4th International Conference on Enactive Interfaces*, pages 49–52, Grenoble, France. ACROE & ICA.
- [Cavusoglu et Tendick, 2000] Cavusoglu, M. C. et Tendick, F. (2000). Multirate simulation for high fidelity haptic interaction with deformable objects in virtual environments. *In ICRA*, pages 2458–2465. IEEE.
- [Çavuşoğlu *et al.*, 2002] Çavuşoğlu, M. C., Feygin, D. et Tendick, F. (2002). A critical study of the mechanical and electrical properties of the PHANTOM haptic interface and improvements for high-performance control. *Presence*, 11(6):555–568.
- [Chanclou, 1996] Chanclou, B. (1996). *Modélisation physique de véhicules tout terrain — Application à la Simulation Dynamique de Robots Mobiles d'Intervention en Milieu Naturel*. Thèse de doctorat, Université Claude Bernard - Lyon I, Grenoble, France.
- [Chu, 2002] Chu, L. L. (2002). Haptic design for digital audio. *In Proceedings of the IEEE International Conference on Multimedia and Expo, ICME '02*, volume 2, pages 441–444.
- [Colgate et Brown, 1994] Colgate, J. E. et Brown, J. M. (1994). Factors affecting the z-width of a haptic display. *In Proceedings of the IEEE 1994 International Conference on Robotics & Automation*, pages 3205–3210, San Diego, CA.
- [Cotin *et al.*, 1999] Cotin, S., Delingette, H. et Ayache, N. (1999). Real-Time Elastic Deformations of Soft Tissues for Surgery Simulation. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):62–73.
- [Couroussé, 2002] Couroussé, D. (2002). Le codage du geste – spécification et implémentation pour le traitement du geste dans un système de représentation par modèle physique. Rapport de stage de fin d'études, département Electronique et Systèmes de Communication, INSA de Rennes, 96 pages, Laboratoire ACROE-ICA.
- [Couroussé, 2007a] Couroussé, D. (2007a). Haptic board. *In Luciani, A. et Cadoz, C., éditeurs : Enaction and Enactive Interfaces : A Handbook of Terms*, pages 126–127. Enactive Interfaces Books, Grenoble, France.
- [Couroussé, 2007b] Couroussé, D. (2007b). Mechanical impedance. *In Luciani, A. et Cadoz, C., éditeurs : Enaction and Enactive Interfaces : A Handbook of Terms*, pages 194–196. Enactive Interfaces Books, Grenoble, France.

- [Couroussé et Florens, 2005] Couroussé, D. et Florens, J. L. (2005). Issues on reactivity of the computer interface and computer architecture. In Florens, J. L. et Couroussé, D., éditeurs : *Preliminary Roadmap*, numéro D.RD1.1.1, pages 26–36. Enactive Interfaces.
- [Couroussé et Florens, 2007a] Couroussé, D. et Florens, J.-L. (2007a). Functional analysis of haptic devices. In Luciani, A. et Cadoz, C., éditeurs : *Proceedings of Enactive/07, 4th International Conference on Enactive Interfaces*, pages 57–60, Grenoble, France. ACROE & ICA.
- [Couroussé et Florens, 2007b] Couroussé, D. et Florens, J. L. (2007b). Issues on computer architecture and reactivity. In Florens, J.-L. et Couroussé, D., éditeurs : *Synthesis Document on Bottlenecks and Roadmap*, numéro Deliverable D.RD.2.1, pages 166–173. Enactive Interfaces.
- [Couroussé et al., 2006a] Couroussé, D., Florens, J.-L. et Luciani, A. (2006a). Effects of stiffness on tapping performance. In *Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'06)*, pages 65–72, Arlington, USA. Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'06).
- [Couroussé et Gil, 2007] Couroussé, D. et Gil, J. J. (2007). Contact interaction. In Luciani, A. et Cadoz, C., éditeurs : *Enaction and Enactive Interfaces : A Handbook of Terms*, pages 55–56. Enactive Interfaces Books, Grenoble, France.
- [Couroussé et al., 2006b] Couroussé, D., Jansson, G., Florens, J.-L. et Luciani, A. (2006b). Visual and haptic perception of object elasticity in a squeezing virtual event. In *Proceedings of the EuroHaptics 2006 Conference*, pages 283–288, Paris, France.
- [de Boeck, 2006] de Boeck, J. (2006). *A User and Designer Perspective on Multimodal Interaction in 3D Environments*. thèse de doctorat - Kennistechnologie, Informatica, Wiskunde, ICT, Universiteit Maastricht, School voor Informatietechnologie.
- [de Götzen et Marogna, 2007] de Götzen, A. et Marogna, R. (2007). Sound algorithms – sound synthesis and sound models. In Luciani, A. et Cadoz, C., éditeurs : *Enaction and Enactive Interfaces : A Handbook of Terms*, pages 273–274. Enactive Interfaces Books, Grenoble, France.
- [de Götzen et al., 2007] de Götzen, A., Mion, L. et Tache, O. (2007). Sound algorithms. In Luciani, A. et Cadoz, C., éditeurs : *Enaction and Enactive Interfaces : A Handbook of Terms*, pages 271–272. Enactive Interfaces Books, Grenoble, France.
- [der Linde et al., 2002] der Linde, R. Q. V., Mammerste, P., Frederiksen, E. et b. Ruiter (2002). The hapticmaster, a new high-performance haptic interface. In Wall, S. A., Riedel, B., Crossan, A. et McGee, M. R., éditeurs : *Proceedings of the EuroHaptics conference*, pages 1–5, Edinburgh, UK.
- [Duriez et al., 2004] Duriez, C., Andriot, C. et Kheddar, A. (2004). A multi-threaded approach for deformable rigid contacts with haptic feedback. *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS '04. Proceedings. 12th International Symposium on*, pages 272–279.
- [Duriez et al., 2006] Duriez, C., Dubois, F., Keddar, A. et Andriot, C. (2006). Realistic Haptic Rendering of Interacting Deformable Objects in Virtual Environments. *Visualization and Computer Graphics, IEEE Transactions on*, 12(1):36–47.
- [Ellis et al., 1996] Ellis, R. E., Ismaeil, O. et Lipsett, M. (1996). Design and evaluation of a high-performance haptic interface. *Robotica*, 4:321–327.

- [Ellis *et al.*, 2004] Ellis, S. R., Mania, K., Adelstein, B. D. et Hill, M. (2004). Generalizeability of latency detection in a variety of virtual environments. *In Human Factors and Ergonomics Society, 48th Annual Meeting*, New Orleans, USA.
- [Ellis *et al.*, 1999] Ellis, S. R., Young, M. J., Adelstein, B. D. et Ehrlich, S. M. (1999). Discrimination of changes in latency during head movement. *In Proceedings of the HCI International '99 (the 8th International Conference on Human-Computer Interaction) on Human-Computer Interaction*, pages 1129–1133. Lawrence Erlbaum Associates, Inc.
- [ENACTIVE, 2004] ENACTIVE (2004). Theoretical and conceptual definitions. Public Deliverable D7.1, European Network of Excellence Enactive Interfaces.
- [ENACTIVE, 2007a] ENACTIVE (2007a). Basic theoretical chapters. Deliverable D.IA2.5, European Network of Excellence Enactive Interfaces.
- [ENACTIVE, 2007b] ENACTIVE (2007b). Kinesthetic model of the user. Deliverable D.RD2.2, European Network of Excellence Enactive Interfaces.
- [Engelbart, 1970] Engelbart, D. C. (1970). X-Y position indicator for a display system. US patent 3,541,541, Palo Alto, California.
- [Faulring *et al.*, 2006] Faulring, E. L., Colgate, J. E. et Peshkin, M. A. (2006). The cobotic hand controller : Design, control and performance of a novel haptic display. *International Journal of Robotics Research*, 25(11):1099–1119.
- [Fisch *et al.*, 2003] Fisch, A., Mavroidis, C., Bar-Cohen, Y. et Melli-Huber, J. (2003). *Biologically-Inspired Intelligent Robots*, chapitre 4 : Haptic devices for virtual reality, telepresence and human-assistive robotics. SPIE Press.
- [Florens, 1978] Florens, J.-L. (1978). *Coupleur Gestuel Retroactif pour la Commande et le controle de Sons Synthetisés en Temps Réel*. Thèse de doctorat, Institut National Polytechnique de Grenoble, Grenoble, France.
- [Florens, 2003] Florens, J.-L. (2003). Expressive bowing on a virtual string instrument. *In* Camurri, A. et Volpe, G., éditeurs : *Gesture-Based Communication in Human-Computer Interaction ; 5th International Gesture Workshop*, pages 487–496, Genova, Italy. Springer-Verlag Berlin Heidelberg 2004.
- [Florens, 2008] Florens, J.-L. (2008). *Technologies de la Simulation Interactive de Modèles Physiques Multisensoriels*. Mémoire d'Habilitation à Diriger les Recherches, ACROE & Laboratoire ICA, Grenoble, France.
- [Florens et Couroussé, 2005] Florens, J.-L. et Couroussé, D., éditeurs (2005). *Preliminary Roadmap*. Numéro D.RD.1.1.1. Enactive Interfaces.
- [Florens et Couroussé, 2007] Florens, J.-L. et Couroussé, D., éditeurs (2007). *Synthesis Document on Bottlenecks and Roadmap, deliverable D.RD.2.1*. Enactive Interfaces.
- [Florens *et al.*, 2004] Florens, J.-L., Luciani, A., Cadoz, C. et Castagné, N. (2004). ERGOS : A multi-degrees of freedom and versatile force-feedback panoply. *In* Buss, M. et Frittschi, M., éditeurs : *Proceedings of Eurohaptics 2004*, pages 356–360, Munich, Germany.
- [Florens *et al.*, 2005] Florens, J.-L., Luciani, A. et Castagné, N. (2005). Sharp colliding of multiple sound objects with haptic feedback. Rapport technique, ACROE & ICA laboratory.

- [Florens et Urma, 2006] Florens, J.-L. et Urma, D. (2006). Dynamical issues at the low level of human / virtual object interaction. In *HAPTICS*, page 47. IEEE Computer Society.
- [Force Dimension, 2008] Force Dimension (2008). the Delta Haptic Device. Datasheet, <http://www.forcedimension.com/downloads/specs/specsheet-delta.6.pdf>.
- [ForceDimension, 2007] ForceDimension (2007). DHD API 3.0.
- [Frisoli *et al.*, 2005] Frisoli, A., Rocchi, F., Marcheschi, S., Dettori, A., Salsedo, F. et Bergamasco, M. (2005). A new force-feedback arm exoskeleton for haptic interaction in Virtual Environments. In *Proceedings of the First Joint Eurohaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*.
- [Gehrke, 1995] Gehrke, J. (1995). Sensorimotor synchronization : the intensity of afferent feedback affects the timing of movements. Rapport technique 15/1995, Max Planck Institute for Psychological Research.
- [Gibet, 1987] Gibet, S. (1987). *Codage, représentation et traitement du geste instrumental – application à la synthèse de sons musicaux par simulation de mécanismes instrumentaux*. thèse de doctorat, spécialité informatique, Institut National Polytechnique de Grenoble, Grenoble, France.
- [Giraud, 1999] Giraud, O. (1999). *Architecture logicielle haute performance pour la simulation temps-réel synchrone d'objets physiques multisensoriels : retour d'effort, synthèse de sons, synthèse d'images*. Spécialité informatique système et communication, Institut National Polytechnique de Grenoble, Grenoble, France.
- [Goertz, 1952] Goertz, R. C. (1952). Fundamentals of general-purpose remote manipulator. *Nucleonics*, 10,11:36–45.
- [Goertz et Thompson, 1954] Goertz, R. C. et Thompson, W. M. (1954). Electronically controlled manipulator. *Nucleonics*, 12(11):46–47.
- [Grange et Conti, 2001] Grange, S. et Conti, F. (2001). The delta haptic device as a nanomanipulator. In *SPIE Microrobotics and Microassembly III*, Boston MA.
- [Grange *et al.*, 2001a] Grange, S., Conti, F., Helmer, P., Rouiller, P. et Baur, C. (2001a). The delta haptic device. In *Mecatronics 2001*, Besançon, France.
- [Grange *et al.*, 2001b] Grange, S., Conti, F., Helmer, P., Rouiller, P. et Baur, C. (2001b). Overview of the delta haptic device. In *Eurohaptics '01*, Birmingham, England.
- [Hayward et Astley, 1996] Hayward, V. et Astley, O. R. (1996). Performance measures for haptic interfaces. In Giralt, G. et Hirzinger, G., éditeurs : *Robotics Research : The 7th International Symposium*, pages 195–207. Springer Verlag.
- [Hayward *et al.*, 1994] Hayward, V., Choksi, J., Lanvin, G. et Ramstein, C. (1994). Design and multi-objective optimization of a linkage for a haptic interface. In *Advances in Robot Kinematics and Computationed Geometry*, pages 359–368. Kluwer Academic Publishers.
- [Hollis et Salcudean, 1992] Hollis, R. L. et Salcudean, S. E. (1992). Input/output system for computer user interface using magnetic levitation. US patent 5,146,566, Yorktown Heights, NY.
- [Hollis et Salcudean, 1993] Hollis, R. L. et Salcudean, S. E. (1993). Lorentz levitation technology : A new approach to fine motion robotics. In *International Symposium of Robotics Research*, Hidden Valley, PA.

- [Hunt *et al.*, 2002] Hunt, A. D., Wanderley, M. et Paradis, M. (2002). The importance of parameter mapping in electronic instrument design. In *Proceedings of the 2002 Conference on New Interfaces for Musical Expression (NIME-02)*, Dublin, Ireland.
- [Hurmuzlu *et al.*, 1998] Hurmuzlu, Y., Ephanov, A. et Stoianovici, D. (1998). Effect of pneumatically driven haptic interface on the perceptual capabilities of human operators. *Presence*, 7(3):290–307.
- [Iwata *et al.*, 2001] Iwata, H., Yano, H., Nakaizumi, F. et Kawamura, R. (2001). Project FEELEX : adding haptic surface to graphics. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 469–476. ACM Press.
- [Jacobsen *et al.*, 1986] Jacobsen, S. C., Iversen, E. K., Knutti, D. K., Johnson, R. T. et Biggers, K. B. (1986). Design of the Utah/MIT Dextrous Hand. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1520–1532, San Francisco.
- [Jean-Loup Florens, 2004] Jean-Loup Florens, I., éditeur (2004). *Report on the State of the Art on Existing sensors/actuators technologies for haptic interfaces, deliverable D3.1*. Enactive Interfaces.
- [Jensenius *et al.*, 2007] Jensenius, A., Camurri, A., Castagné, N., Maestre, E., Malloch, J., McGilvray, D. et Schwarz, D. (2007). Panel : The need of formats for streaming and storing music-related movement and gesture data. In *Proceedings of the International Computer Music Conference (ICMC)*, volume II, pages 13–16, Copenhagen, Denmark.
- [Jöhl *et al.*, 2005] Jöhl, K., Preusche, C. et Reintsema, D. (2005). High Fidelity USB Force Feedback Joystick. In *Proceedings of the World Haptics Conference (WHC 2005)*, Pisa, Italy.
- [Jones, 1989] Jones, L. A. (1989). Matching forces : constant efforts and differential thresholds. *Perception*, 18(5):681–687.
- [Jones, 1997] Jones, L. A. (1997). Dextrous hands : Human, prosthetic, and robotic. *Presence*, 6(1):29–56.
- [Jones et Hunter, 1990] Jones, L. A. et Hunter, I. W. (1990). A perceptual analysis of stiffness. *Experimental Brain Research*, 79(1):150–156.
- [Katz, 1989] Katz, D. (1989). *The world of Touch*. Lawrence Erlbaum Associates.
- [Lawrence, 1993] Lawrence, D. A. (1993). Stability and transparency in bilateral teleoperation. *IEEE Transactions on Robotics and Automation*, 9(5):624–637.
- [Lawrence *et al.*, 2000] Lawrence, D. A., Pao, L. Y., Dougherty, A. M., Salada, M. A. et Pavlou, Y. (2000). Rate-hardness : a new performance metric for haptic interfaces. *IEEE Transactions on Robotics and Automation*, 16(4):357–371.
- [Lawrence *et al.*, 1996] Lawrence, D. A., Pao, L. Y., Salada, M. A. et Dougherty, A. M. (1996). Quantitative experimental analysis of transparency and stability in haptic interfaces. In *Proceedings of the Fifth Annual Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, held at the ASME Winter Annual Meeting, Atlanta, GA. ASME Winter Annual Meeting.
- [Laycock et Day, 2003] Laycock, S. et Day, A. (2003). Recent developments and applications of haptic devices. *Computer Graphics Forum*, 22(2):117–132.

- [Lecarme, 1971] Lecarme, O. (1971). A system for interactive graphic programming. *In IFIP congress 71*, Ljubljana, Slovenia.
- [Lécuyer, 2001] Lécuyer, A. (2001). *Contribution à l'étude des retours haptique et pseudo-haptique et de leur impact sur les simulations d'opérations de montage / démontage en aéronautique*. Thèse de doctorat en informatique, Université de Paris XI Orsay.
- [Lécuyer et al., 2003] Lécuyer, A., Andriot, C. et Crosnier, A. (2003). Interfaces haptiques et pseudo-haptique. *In Proceedings of JNRR'03 (4e Journées Nationales de la Recherche en Robotique)*.
- [Lecuyer et al., 2001] Lecuyer, A., Burkhart, J. M. et Coquillart, S. (2001). "boundary of illusion" : an experiment of sensory integration with a pseudo-haptic system. *In IEEE International Conference on Virtual Reality (IEEE VR)*, pages 115–122, Yokohoma, Japan.
- [Leuschke et al., 2005] Leuschke, R., Kurihara, E. K. T., Doshier, J. et Hannaford, B. (2005). High fidelity multi finger haptic display. *In Proceedings of the First Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*.
- [Lin et Salisbury, 2004] Lin, M. C. et Salisbury, K. (2004). Haptic rendering - beyond visual computing. *IEEE Computer Graphics and Applications*, 24(2):22–23.
- [Luciani, 2004] Luciani, A. (2004). *State of the Art on current interaction paradigms based on action and vision*, chapitre 2- Action-Vision fusion un CG, HCI, VR, teleoperation, telecommunication : Novels Issues, pages 15–25. ENACTIVE.
- [Luciani et al., 2006a] Luciani, A., Evrard, M., Castagné, N., Couroussé, D., Florens, J.-L. et Cadoz, C. (2006a). A basic gesture and motion format for virtual reality multisensory applications. *In Proceedings of the 1st international Conference on Computer Graphics Theory and Applications (GRAPP)*, Setubal, Portugal.
- [Luciani et al., 2006b] Luciani, A., Evrard, M., Couroussé, D., Castagné, N., Cadoz, C. et Florens, J.-L. (2006b). A basic gesture and motion format for virtual reality multisensory applications. 2nd Enactive Workshop, McGill University, Montreal, Canada.
- [Luciani et al., 2005a] Luciani, A., Florens, J.-L. et Castagné, N. (2005a). From action to sound : a challenging perspective for haptics. *In WHC'05 : Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 592–595.
- [Luciani et al., 2007a] Luciani, A., Florens, J.-L., Magnusson, C., O'modhrain, S. et Couroussé, D. (2007a). Emblematic enactive scenario "creativity : Pebble box". Deliverable D.EES.4, NoE Enactive Interfaces IST-2004-002114-ENACTIVE.
- [Luciani et al., 2005b] Luciani, A., Magnusson, C., Davies, R., Florens, J.-L., Couroussé, D., Castagné, N., Tache, O., Castet, J. et Cadoz, C. (2005b). Virtual pebble box experiments INPG-ULUND 5-6 December 2005. Experiment Report EI_ULUND_INPG_VPebbleB_051204.doc, Enactive Interfaces IST-2004-002114-ENACTIVE.
- [Luciani et al., 2007b] Luciani, A., O'modhrain, S., Magnusson, C., Florens, J.-L. et Couroussé, D. (2007b). Perception of virtual multisensory mobile objects – wandering around the enactive assumption. *In Luciani, A. et Cadoz, C., éditeurs : Proceedings of Enactive/07, 4th International Conference on Enactive Interfaces*, pages 153–156, Grenoble, France. ACROE & ICA.

- [MacLean *et al.*, 1999] MacLean, K. E., Snibbe, S. S. et Shaw, R. S. (1999). An architecture for haptic control of media. In *Proceedings of the 8th Annual Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems, ASME / IMECE*, volume DSC-5B-3, Nashville, TN.
- [Magnusson *et al.*, 2006] Magnusson, C., Luciani, A., Couroussé, D., Davies, R. et Florens, J.-L. (2006). Preliminary test in a complex virtual dynamic haptic audio environment. 2nd Enactive Workshop, McGill University, Montreal, Canada.
- [Mäki-Patola et Hämäläinen, 2004] Mäki-Patola, T. et Hämäläinen, P. (2004). Latency tolerance for gesture controlled continuous sound instrument without tactile feedback. In *Proceedings of the International Computer Music Conference (ICMC 2004)*, Miami, USA.
- [Mania *et al.*, 2004] Mania, K., Adelstein, B. D., Ellis, S. R. et Hill, M. I. (2004). Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. In *Proceedings of the 1st Symposium on Applied perception in graphics and visualization*, pages 39–47. ACM Press.
- [Mark *et al.*, 1996] Mark, W. R., Randolph, S. C., Finch, M., Verth, J. M. V. et Russell M. Taylor, I. (1996). Adding force feedback to graphics systems : issues and solutions. In *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 447–452, New York, NY, USA. ACM.
- [Massie et Salisbury, 1994] Massie, T. H. et Salisbury, J. K. (1994). The PHANTOM haptic interface : A device for probing virtual objects. In *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Chicago, IL.
- [Mates et Aschersleben, 2000] Mates, J. et Aschersleben, G. (2000). Sensorimotor synchronization : the impact of temporally displaced auditory feedback. *Acta psychologica*, 104:29–44.
- [Mavroidis *et al.*, 2000a] Mavroidis, C., Pfeiffer, C., Celestino, J. et Bah-Cohen, Y. (2000a). Controlled compliance haptic interface using electrorheological fluids. In *Proceedings of the 2000 SPIE Conference on Electro-Active Polymer Actuators and Devices (EAPAD 2000), SPIE 7th Annual International Symposium on Smart Materials and Structures*, Newport Beach, CA.
- [Mavroidis *et al.*, 2000b] Mavroidis, C., Pfeiffer, C., Celestino, J. et Bah-Cohen, Y. (2000b). Design and modeling of an electro-rheological fluid based haptic interface. In *Proceedings of DETC'00 : 26th Biennial Mechanism and Robotics Conference*, Baltimore, Maryland.
- [Mazzella *et al.*, 2002] Mazzella, F., Montgomery, K. et Latombe, J.-C. (2002). The Forcegrid : A Buffer Structure for Haptic Interaction with Virtual Elastic Objects. In *ICRA*, pages 939–946. IEEE.
- [Mendès, 2002] Mendès, D. M. (2002). Implantation sur une architecture multiprocesseur d'un serveur pour la simulation temps réel interactive des modèles physiques. Mémoire de D.E.A., ENSIMAG, Grenoble, France.
- [Merlin, 2007] Merlin, D. (2007). Etude et réalisation d'un module électronique permettant d'interfacer un stimulateur tactile "TACTOS" (afficheur braille) avec une carte DSP supportant un simulateur d'objets physiques temps réel. Master 1 – stage ingénieur ENSERG, INP Grenoble, INPG, Grenoble.

- [Murayama *et al.*, 2004] Murayama, J., Bouguila, L., Luo, Y., Akahane, K., Hasegawa, S., Hirsbrunner, B. et Sato, M. (2004). SPIDAR G&G : A two-handed haptic interface for bimanual VR interaction. *In Proceedings of the EuroHaptics conference*, Technische Universität in Munich, Germany.
- [Nichols, 2002] Nichols, C. (2002). The vBow : Development of a virtual violin bow haptic human computer interface. *In Proceedings of International Conference on New Interfaces For Musical Expression*, Dublin, Ireland.
- [Nikolakis *et al.*, 2004] Nikolakis, G., Tzovaras, D., Moustakidis, S. et Strintzis, M. G. (2004). Cybergrasp and Phantom Integration : Enhanced Haptic Access for Visually Impaired Users. *In SPECOM-2004*, pages 507–513.
- [O’Modhrain, 2000] O’Modhrain, M. S. (2000). *Playing by Feel : Incorporating Haptic Feedback into Computer-Based Musical Instruments*. Thèse de doctorat, Stanford University.
- [O’Modhrain et Essl, 2004] O’Modhrain, S. et Essl, G. (2004). Pebblebox and crumblebag : Tactile interfaces for granular synthesis. *In Proceedings of NIME’04*, Hamamatsu, Japan.
- [Pang *et al.*, 1991] Pang, X.-D., Tan, H. Z. et Durlach, N. I. (1991). Manual discrimination of force using active finger motion. *Perception & Psychophysics*, 49(6):534–540.
- [Patterson, 2004] Patterson, D. A. (2004). Latency lags bandwidth. *Communications of the ACM*, 47(10):71–75.
- [Popescu *et al.*, 1999] Popescu, G., Burdea, G. C. et Bouzit, M. (1999). Virtual reality simulation modeling for a haptic glove. *Computer Animation, 1999. Proceedings*, pages 195–200.
- [Prévo, 2004a] Prévo, C. (2004a). Description technique et architectures. *In Conversions analogique-numérique et numérique-analogique*, numéro E371. Techniques de l’Ingénieur.
- [Prévo, 2004b] Prévo, C. (2004b). Introduction. *In Conversions analogique-numérique et numérique-analogique*, numéro E370. Techniques de l’Ingénieur.
- [Prévo, 2004c] Prévo, C. (2004c). Marché, technologie et applications. *In Conversions analogique-numérique et numérique-analogique*, numéro E372. Techniques de l’Ingénieur.
- [Ramstein et Hayward, 1994] Ramstein, C. et Hayward, V. (1994). The PANTOGRAPH : a Large Workspace Haptic Device for Multi-Modal Human-Computer Interaction. *In Conference Companion on Human Factors in Computing Systems (CHI’94)*, pages 57–58. ACM Press.
- [Rosenberg et Brave, 1996] Rosenberg, L. B. et Brave, S. (1996). Using force feedback to enhance human performance in graphical user interfaces. *In Proceedings of CHI’96 : Conference on human Factors in Computing Systems*.
- [Ruspini *et al.*, 1997] Ruspini, D. C., Kolarov, K. et Khatib, O. (1997). The haptic display of complex graphical environments. *In SIGGRAPH ’97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 345–352, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.
- [Salisbury *et al.*, 1995] Salisbury, J. K., Brock, D. L., Massie, T., Swarup, N. et Zilles, C. B. (1995). Haptic rendering : Programming touch interaction with virtual objects. *In Symposium on Interactive 3D Graphics*, pages 123–130, Monterey, California, United States.
- [Salisbury *et al.*, 2004] Salisbury, K., Conti, F. et Barbagli, F. (2004). Haptic rendering : Introductory concepts. *IEEE Computer Graphics and Applications*, 24(2):24–32.

- [Savall *et al.*, 2002] Savall, J., Borro, D., Gil, J. J. et Matey, L. (2002). Description of a haptic system for virtual maintainability in aeronautics. In *Intelligent Robots and System, 2002. IEEE/RSJ International Conference on*, volume 3, pages 2887–2892, EPFL, Lausanne, Switzerland.
- [Schmandt et Hulteen, 1982] Schmandt, C. et Hulteen, E. A. (1982). The intelligent voice-interactive interface. In *Proceedings of the 1982 conference on Human factors in computing systems*, pages 363–366, New York, NY, USA. ACM Press.
- [Scilingo *et al.*, 2000] Scilingo, E. P., Bicchi, A., De Rossi, D. et Scotto, A. (2000). A magnetorheological fluid as a haptic display to replicate perceived biological tissues compliance. In *1st Annual International Conference On Microtechnologies in Medicine and Biology*, pages 229–233, Lyon, France.
- [Scilingo *et al.*, 2003a] Scilingo, E. P., Sgambelluri, N., De Rossi, D. et Bicchi, A. (2003a). Haptic displays based on magnetorheological fluids : design, realization and psychophysical validation. In *11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTIC 2003)*, pages 10–15.
- [Scilingo *et al.*, 2003b] Scilingo, E. P., Sgambelluri, N., De Rossi, D. et Bicchi, A. (2003b). Towards a haptic black box for free-hand softness and shape exploration. In *IEEE International Conference on Robotics and Automation (ICRA'03)*, volume 2, pages 2412–2417, Taipei, Taiwan.
- [SensableTechnologies, 2005] SensableTechnologies (2005). OpenHaptics™ Toolkit Programmer's Guide, Version 2.0.
- [Serafin, 2004] Serafin, S. (2004). *The sound of friction : real-time models, playability and musical applications*. PhD Thesis, Stanford University, CCRMA.
- [Sheridan, 1992] Sheridan, T. B. (1992). *Telerobotics, Automation, and Human Supervisory Control*. MIT Press.
- [Shimoga, 1992] Shimoga, K. B. (1992). Finger force and touch feedback issues in dexterous telemanipulation. In *Proceedings of the Intelligent Robotic Systems for Space Exploration*, pages 159–178.
- [Shimoga et Khosla, 1994] Shimoga, K. B. et Khosla, P. (1994). Touch and force reflection for telepresence surgery. In *Proceedings of the 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Engineering Advances : New Opportunities for Biomedical Engineers*, volume 2, pages 1049–1050.
- [Sinclair et Wanderley, 2007] Sinclair, S. et Wanderley, M. (2007). Defining a control standard for easily integrating haptic virtual environments with existing audio/visal systems. In *Proceedings of the 2007 International Conference on New Interfaces for Musical Expression (NIME07)*, pages 209–212, New York City, USA.
- [Smart *et al.*, 2002] Smart, L. J., Stoffregen, T. A. et Bardy, B. (2002). Visually induced motion sickness predicted by postural instability. *Human Factors*, 44(3).
- [Smyth et Wing, 1984] Smyth, M. M. et Wing, A. M. (1984). *The psychology of Human Movement*, chapitre 1. Movement, action and skill, pages 1–15. Academic Press.
- [Sreng *et al.*, 2007] Sreng, J., Bergez, F., Legarrec, J., Lécuyer, A. et Andriot, C. (2007). Using an event-based approach to improve the multimodal rendering of 6dof virtual contact. In *VRST*

- '07 : *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, pages 165–173, New York, NY, USA. ACM.
- [Srinivasan *et al.*, 1996] Srinivasan, M. A., Beauregard, G. L. et Brock, D. L. (1996). The impact of visual information on the haptic perception of stiffness in virtual environments. *In Proceedings of the ASME Dynamic Systems and Control Division*, volume 58, pages 555–559, Atlanta, GA, USA. ASME.
- [Steelman, 1968] Steelman, H. S. (1968). The GROPE-1 system : analysis of friction and backlash problems. Mémoire de D.E.A., University of North Carolina, Chapel Hill.
- [Stone, 2000] Stone, R. J. (2000). Haptic feedback : A potted history, from telepresence to virtual reality. *In Haptic Human-Computer Interaction Workshop*, volume First International Workshop de *Lecture Notes in Computer Science*.
- [Summers *et al.*, 2005] Summers, I. R., Brady, A. C., Syed, M. K. et Chanter, C. M. (2005). Design of array stimulators for synthetic tactile sensations. *In [DBL, 2005]*, pages 586–587.
- [Sutherland, 1965] Sutherland, I. (1965). The ultimate display. *In Kalendich, W. A., éditeur : Proceedings of IFIP Congress 65*, volume 2, pages 506–508, New York City. Spartan Books, Washington, D.C.
- [Tan *et al.*, 2006] Tan, H. Z., Adelstein, B. D., Taylor, R., Kocsis, M. et Hirleman, E. D. (2006). Discrimination of real and virtual high-definition textured surfaces. *In Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'06)*, pages 3–9, Alexandria, Virginia, USA.
- [Tan *et al.*, 1995] Tan, H. Z., Durlach, N. I., Beauregard, G. L. et Srinivasan, M. A. (1995). Manual discrimination of compliance using active pinch grasp : The rols of force and work cues. *Perception & Psychophysics*, 57(4):495–510.
- [Tan *et al.*, 1993] Tan, H. Z., Durlach, N. I., Shao, Y. et Wei, M. (1993). Manual resolution of compliance when work and force cues are minimized. *In Kazerooni, H. et Colgate, J. E., éditeurs : DSC–Vol 49, Advances in Robotics, Mechatronics and Haptic Interfaces*, numéro H00851, pages 99–104.
- [Tan *et al.*, 1992] Tan, H. Z., Pand, X. D. et Durlach, N. I. (1992). Manual resolution of length, force, and compliance. *In Kazerooni, H., éditeur : DSC–Vol 42, Advances in Robotics*, numéro G00751, pages 13–18.
- [Tan *et al.*, 1994] Tan, H. Z., Srinivasan, M. A., Eberman, B. et Cheng, B. (1994). Human factors for the design of force-reflecting haptic interfaces. *Dynamic Systems and Control, DSC–Vol.55–1*, pages 353–359.
- [Technology, 2000a] Technology, L. (2000a). LTC1591/LTC1597 14-Bit and 16-Bit Parallel Low Glitch Multiplying DACs with 4-Quadrant Resistors. Data sheet.
- [Technology, 2000b] Technology, L. (2000b). LTC1606 16-bit, 250ksps, Single Supply ADC. Data sheet.
- [Traylor *et al.*, 2005] Traylor, R. M., Wilhelm, D., Adelstein, B. D. et Tan, H. Z. (2005). Design Considerations for Stand-Alone Haptic Interfaces Communicating via UDP Protocol. *In [DBL, 2005]*, pages 563–564.

- [Uhl, 1996] Uhl, C. (1996). *Architecture de machine pour la simulation d'objets physiques en temps-réel*. Thèse de doctorat, spécialité signal, image, parole, Institut National Polytechnique de Grenoble, Grenoble, France.
- [Unger *et al.*, 2003] Unger, B. J., Klatzky, R. L. et Hollis, R. L. (2003). A telemanipulation system for psychophysical investigation of haptic interaction. *In Int'l Conf. on Robotics and Automation*, pages 1253–1258, Taipei, Taiwan.
- [Varela *et al.*, 1993] Varela, F., Thompson, E. et Rosch, E. (1993). *L'inscription corporelle de l'esprit*. Seuil, Paris.
- [Vertut et Coiffet, 1984] Vertut, J. et Coiffet, P. (1984). *Les Robots*, volume 3A : Teleoperation, Evolution des Technologies. Hermès Publishing, Paris, France.
- [Wanderley et Depalle, 1999] Wanderley, M. et Depalle, P. (1999). *Interfaces homme-machine et création musicale*, chapitre 7- Contrôle gestuel de la synthèse sonore, pages 145–163. Hermès, Paris.
- [Wessel et Wright, 2002] Wessel, D. et Wright, M. (2002). Problems and prospects for intimate musical control of computers. *Computer Music Journal*, 26(3):11–22.
- [Zarudiansky, 1981] Zarudiansky, A. (1981). Remote handling devices. US patent 4,302,138, Velizy, France.
- [Zilles et Salisbury, 1995] Zilles, C. et Salisbury, J. K. (1995). A constraint-based god-object method for haptic display. *Proceedings of the IEE/RSJ International Conference on Intelligent Robots and Systems, Human Robot Interaction, and Cooperative Robots*, 03:146–151.

Annexe A

La Technologie ERGOS

La première version du moteur en tranche utilisé actuellement dans la technologie ERGOS (Transducteur Gestuel Rétroactif), appelé *Clavier Rétroactif Modulaire*, fut présentée en 1988 [Cadoz *et al.*, 1988]. Il s'agissait du premier système haptique permettant de disposer d'un nombre illimité de degrés de liberté, par la mise en parallèle de « tranches » indépendantes. Le système fut successivement amélioré en 2001, puis 2007.

La technologie ERGOS actuelle constitue la 3^{ème} version de la technologie dite « moteur en tranches » (figure A.1). Les actionneurs sont des bobines électromagnétiques alimentées par une amplificateurs de courant linéaires, enroulées autour d'une base électromagnétique de ferrite. Les tranches sont guidées verticalement par des glissières à billes présentant des forces de friction résiduelles inférieures à 5×10^{-3} N. La plage de déplacement des tranches est de 20 mm, et la mesure de la position s'effectue grâce à des capteurs LVDT situés à l'arrière de la structure.

La technologie électromagnétique permet aux actionneurs de présenter une bande passante supérieure à 10 kHz, pour une force maximale de plus de 200 N et une vitesse maximale de $1,8 \text{ m.s}^{-1}$. Le tableau A.1 résume les principales caractéristiques de la base ERGOS.

Il est possible d'adapter à la base ERGOS différents habillages mécaniques, permettant d'obtenir des morphologies variées suivant l'application gestuelle ciblée : sticks à 3 ou 6 degrés de liberté, clavier, archet. Les caractéristiques des principaux habillages mécaniques disponibles sont résumées dans le tableau A.2.

Tableau A.1. Caractéristiques de la base ERGOS

Général	
Type de contrôle	Impédance
Puissance requise	1 000 W
Inertie de la partie mobile	300 g par tranche
Mécanique	
Déplacement	20 mm par tranche
Espace de travail	Configurable — cf. habillages mécaniques
Nombre de degrés de liberté	Illimité
Force de friction max.	5×10^{-3} N
Actionneurs	
Principe physique	Bobines électromagnétiques
Force max. (pic)	200 N
Force max. (continu)	60 N
Vitesse max.	$1,8 \text{ m.s}^{-1}$
Accélération max.	60 m.s^{-2}
Capteurs	
Technologie	LVDT
Résolution	$1 \mu\text{m}$
Conditionnement signaux	
Entrée	± 10 V analogique
Sortie	± 10 V analogique

Tableau A.2. Caractéristiques des habillages mécaniques

Clavier	Nombre illimité de degrés de liberté
	Déplacement vertical de 50 mm par touche
Stick 3D	3 degrés de liberté
	Espace de travail : $60 \times 60 \times 25$ mm
Stick 6D	6 degrés de liberté
	Translations : $50 \times 50 \times 25$ mm
	Rotations : 60° sur les 3 axes
Archet	2 degrés de liberté
	Espace de travail : 150×150 mm



Figure A.1. Trois systèmes ERGOS équipés de différents habillages mécaniques : à gauche, un stick 6D, au milieu un stick 3D, à droite un archet

Annexe B

Données pour le Scénario « PebbleBox »

Cette annexe rapporte les réponses des visiteurs du scénario « Pebble Box », l'analyse des résultats et quelques statistiques de visites. Nous présentons également en fin d'annexe les captures d'écran du scénario (page B-6).

1 Retours de visite

1.1 Retours de visite pour l'expérience 1 (PebbleBox de QUB)

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
PB1 — Eau	« The connection between interaction with water and pebbles is natural » « Liquification of solids » « Impression de marcher dans l'eau, le mouvement des doigts se substitue au mouvement des jambes dans l'eau » « moving my hands in the water » « fontaine » « toucher l'eau » « plage, lieu relaxant » « River side, fountain »	P MP : « it gives the sensation of fluent movement of the pebles » I : « Problème de dimension (doigts-jambes) et de quantité d'eau bougée » MP : « don't know really, but there is not a complete correspondence between the manipulation and the sound, maybe because of the dimensions of the stones » MP : « manipuler des galets et obtenir des "bruits d'eau" » P : « because of the stimulation of the senses seems almost possible »
Suite page suivante (PB1 — Eau)		

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
	« The impression is that the sound is related to movement only in-so-far as it happens at the same time as the action. However, I find that there is little relation between the transients in the action-sound relationship. » « It's so funny !! » « Un caillou qui tombe dans l'eau » « Extranse feeling : I hear water, but I felt stones !!! » « La plage » « Au bord de l'eau. » « Galets au bord d'une rivière » « Rivière ; sensation de froid ; impression de couler » « Wine pouring » « À une bac d'eau » « Mettre la main dans l'eau, lac... » « Pouring gravelly substance through the fingers sometimes, nearly like water »	MP MP : « The sound that I imagined is completely different from that one in actual. » P : « situation que l'on peut facilement rencontrer dans la réalité » MP : « The sound is synchronised with the hands movement » P : « cailloux + eau = plage » MP : « Il est possible - ou pas forcément - possible de toucher des cailloux en étant au bord de l'eau. » P P : « galets et plage » MP : « c'est sec » MP

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
PB1 — Oiseaux	« There is something being disturbed by moving the stones » « Écrabouillage de poussins » « manipulating little animals ? »	P : « there could be some live creatures under the rocks being disturbed by their movement » MP : « Quantité, il faudrait écraser des centaines de poussins à la pelle mécanique » I : « i should expect a different tactile sensation »
Suite page suivante (PB1 — Oiseaux)		

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
	« être dans un zoo, dans la jungle »	I : « parce que ces bruits d'oiseaux ne sont pas cohérents avec la manipulation des galets »
	« bird cage, beach »	MP
	« Animals »	I : « The sound continues even when the action stops. I am only pressing "play". »
	« jet de pierre a des oiseaux »	MP
	« Cris de jeunes oiseaux. »	I : « La manipulation de cailloux n'a pas grand rapport avec des cris d'oiseaux. »
	« seagulls singing the sines »	MP
	« nid d'oiseau »	P

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
PB1 — Croquement de pomme	« Temporal extention of the plausible natural sound »	I : « sounds artificial; there is no narrative association »
	« marcher sur des insectes »	P
	« grasping on something »	I
	« croquer une pomme, un biscuit »	MP : « c'est un son de "toucher", "casser" qui ressemble a la manipulation »
	« no sure, folding paper ? »	MP : « because of the manipulation of the stones and the association with the sound seems quite possible »
	« Eating an apple »	MP : « I felt the relation to sound more consistent because in this sound the attack is less important. Also it sounds more granular which works better with the pebbles. »
	« croquer dans une pomme »	MP : « le toucher correspond moyennement à la perception auditive »
	« croqueur de chocolat; casseur de cailloux »	
	« grinding friction »	MP

Suite page suivante (PB1 — Croquement de pomme)

Expérience	Description de la scène suggérée	Évaluation et commentaires (P : Plausible, MP : Moyennement Plausible, I : Impossible)
	« croquer dans une pomme »	I : « Le son est trop identique lors de la répétition »

1.2 Retours de visite pour l'expérience 2 (PebbleBox ULUND)

Expérience	Retour subjectif : que suggère l'expérience ?
PB2 — sans visu	« un champ qui empêche de sortir d'une boîte » « Une résistance » « Une sorte de barrière » « Parcours de trajectoires à l'intérieur d'une sphère » « but a bit like "objects" too intriguing. an obvious resemblance to the "pebble box" (I suppose this is deliberate). »
PB2 — avec visu	« déplacer une boule » « champ de forces – importance de la visualisation sur sensation » « The visual feedback pushes my perception towards "objects", although there is also a force-field in the "free" space above the objects. I could happily have played with this one for hours. I discovered - by accident - that certain action-sequences make one or more of the big purple balls fly into the air. »

1.3 Retours de visite pour l'expérience 3 (PebbleBox ACROE)

Expérience	Retour subjectif : que suggère l'expérience ?
PB3 — cas 1	« It seems that there is a rugged sphere in the middle of the workspace » « Playing a bell, or a set of bells » « représentation d'une cloche (type cloche de vache) » « moving around and hitting on little sounding bells or sounding spheres » « carillon de porte, harpe » « A box of metallic marbles. » « There is something in center, maybe... » « retour de force lorsque l'on change de direction » « aucune idée » « faire bouger des boules métalliques dans un espace restreint de sorte qu'elles sont parfois bloquées – déplacer des objets sur un milieu type tôle ondulée » « in a plain there is sth in the center with high rugosity » « Billes au fond d'une boîte. » « le stylet parcourt la surface de boules à peu près fixes » « espace dans lequel on ressent une résistance localisée à certains endroits. après un certain temps, les résistances ne sont pas localisées aux mêmes endroits »

Suite page suivante (PB3)

Expérience	Retour subjectif : que suggère l'expérience ?
	« exploration d'un espace avec 3 objets » « In context - a stick in a pebble-box ! With the sounds : a small number (6 ? ?) big objects on the bottom (deep sounds), and many small objects jumping around above them (tinkling sounds) »
PB3 — cas 2	« playing some ringing spheres, which have an attractive force one to each other or an elastic force keeping them in the rest position » « jeux avec des boules élastiques » « jouer d'un instrument de musique » « maybe chimes being moved by wind » « A bunch of spheres » « on sent que l'on peut se déplacer dans toutes les directions quand on ne rencontre rien. Dès que l'on touche une boule, il y a un retour de force » « un matériau granulaire ; on ressent très nettement que la résistance qu'oppose les billes dépend de la vitesse de déplacement » « boules qui tournent dans un espace » « exploration d'un espace avec des boules » « je suis plus abile avec ma main ; j'ai conscience que dans un simple mouvement de main il y a plusieurs centaines de possibilités ; je sens la médiation. Jeu de quilles, les boules ; pas de ping pong ; jeux de billes » « Well - great fun ! ! With the visual feedback, two clearly different situations : (i) one ball in the middle, 7 around it ; (ii) all 8 balls in a tight outside circle, nothing in the middle. »
PB3 — cas 3	« petits objets, ça frotte » « Irregular jelly. » « on sent un très léger retour » « assez peu de retour sur cette expérience, j'ai l'impression de me déplacer sur un plan avec une légère rugosité au centre » « le stylet se meut dans un espace vide » « exploration d'un milieu » « roulement a billes » « very nice ! I had to change register completely, "tone down" the dynamics and explore very slight, subtle effects. Even trying to take this into account, I also had the IMPRESSION (by now, I've learned to guess that my impression is wrong) that there were no discrete objects, rather a gently distributed force-field - that could nevertheless gain "consistency" by certain sorts of movements (but I discovered them by accident and cannot identify these movements clearly) »

2 Captures d'écran du scénario



The experiment will set with three successive steps using three different demonstration stations.





At any time, you can restart from the beginning by clicking on "Home", at the bottom of the screen.
To continue, click on "Next".

Home Next



Contribution

You can actively participate to the experiment :

☐ Anonymous

Last name :

First name :

Profession :

Hobbies :

Home Next



First Pebble Box

This experiment uses the wooden box on your left as interface, and the headphones.



When you click on "Next", please wear the headphones and palm the pebbles.

Home Next



Situation 1

A scene is being simulated.
Plunge your hands in the chest to explore it.



Home Next



Situation 1

What are the situations suggested by this experiment?
(you can give several answers, even if they are very different)

Do you think that the association between the sound heard and your manipulation is:

Why?

Home Next



Situation 2

A scene is being simulated.
Plunge your hands in the chest to explore it.



Home Next



Situation 2

What are the situations suggested by this experiment?
(you can give several answers, even if they are very different)

Do you think that the association between the sound heard and your manipulation is:

Why?

Home Next



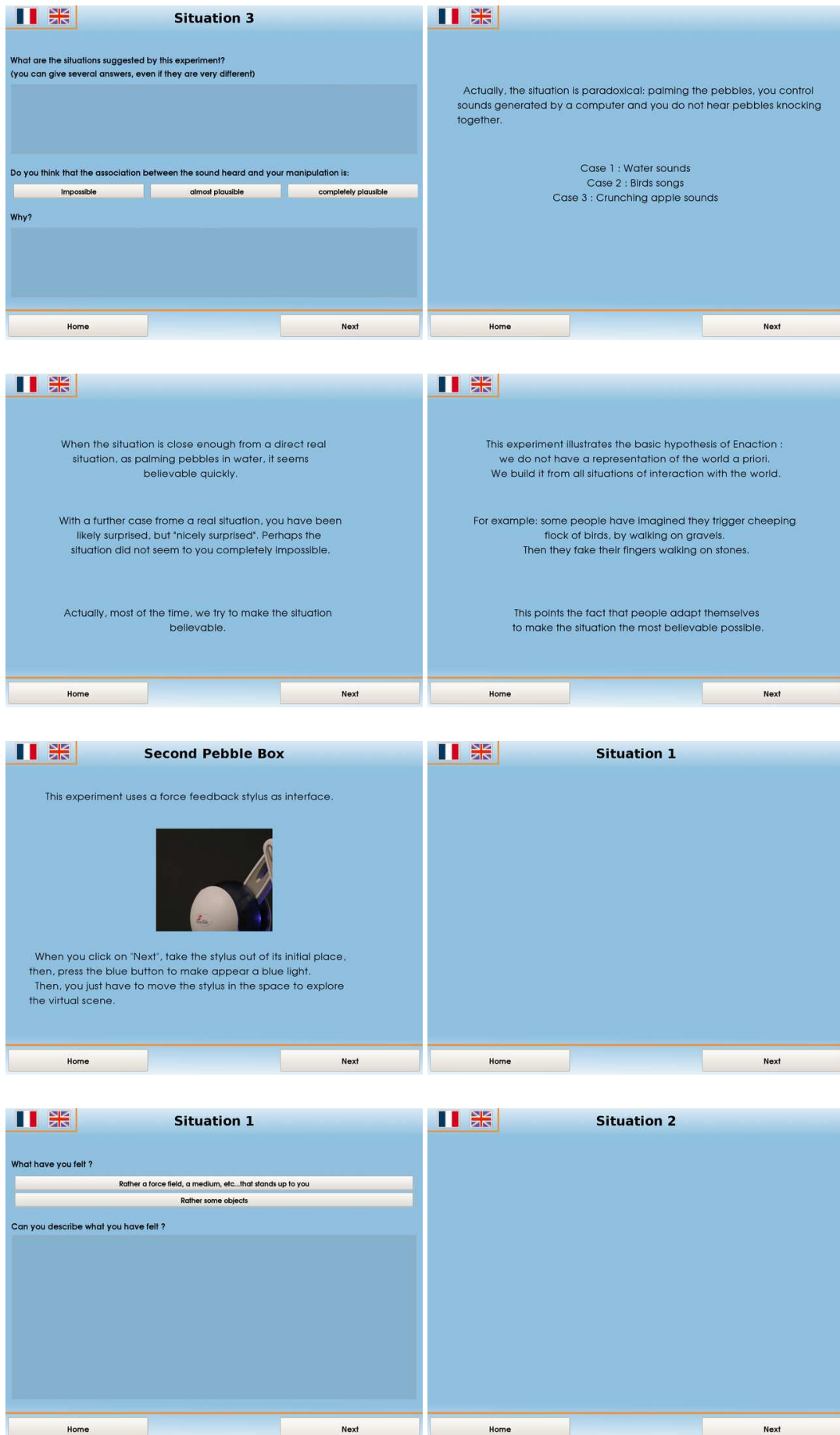
Situation 3

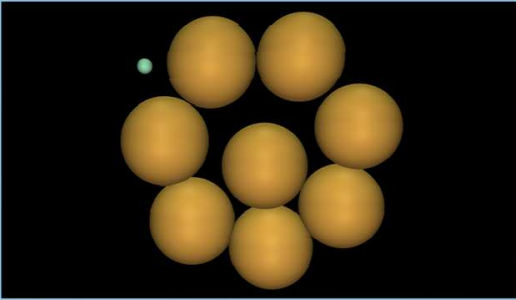
A scene is being simulated.
Plunge your hands in the chest to explore it.

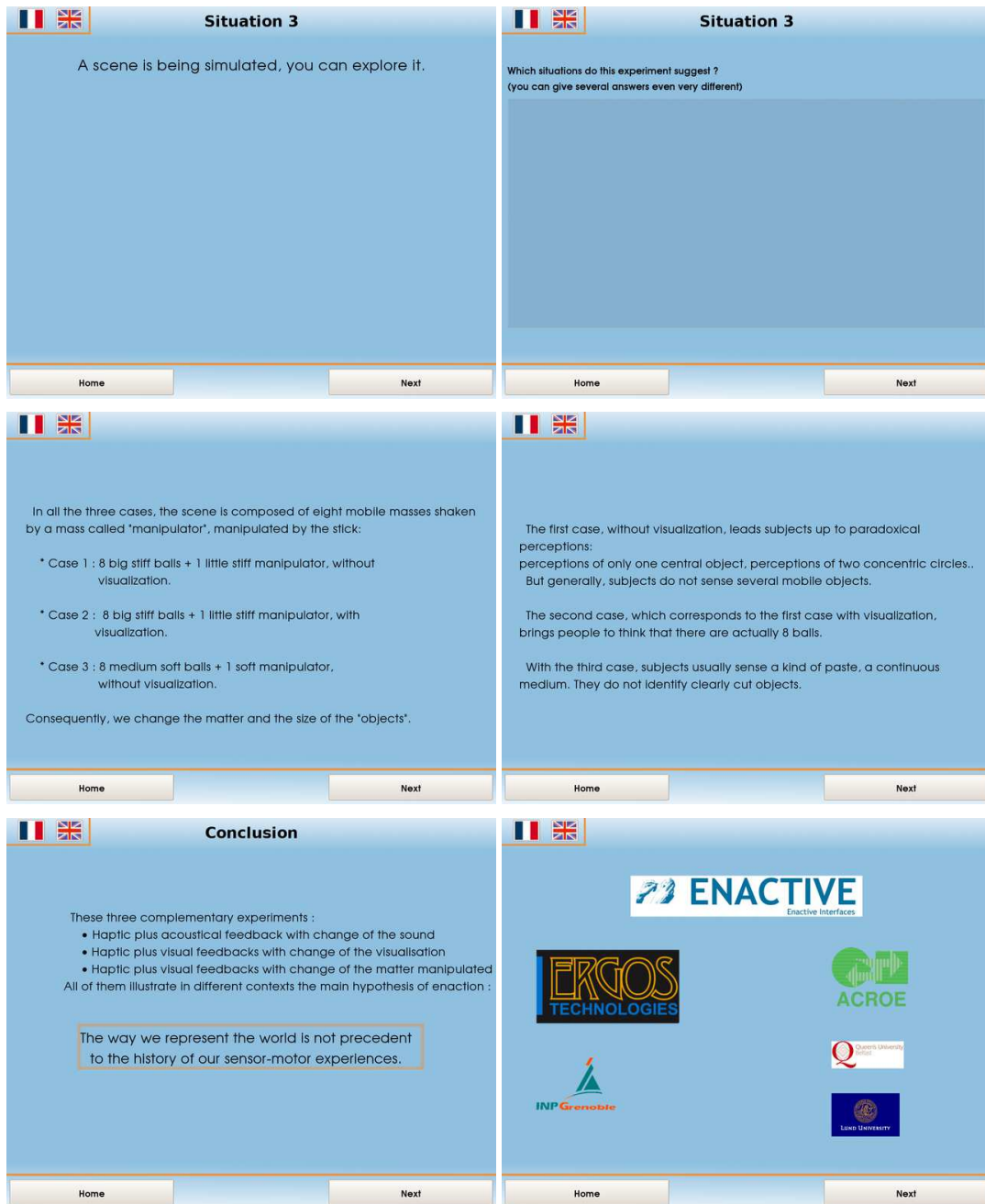


Home Next

2. CAPTURES D'ÉCRAN DU SCÉNARIO



Situation 2 What have you felt ? Rather a force field, a medium, etc... that stands up to you Rather some objects Can you describe what you have felt ? Home Next	Situation 2 Actually, with the force feedback stylus, you shake ten quite rigid balls. The green ball represents the stylus position, thus your hand position. The difference between both cases is: Case 1 : you can't see any ball Case 2 : you see all the ten balls In both cases, the force feedback is the same. Home Next
Situation 2 Thereby, again, the imagined scene depends on our perceptions. In the first case, subjects imagine rarely that there are clearly cut objects (force field, etc...). In the second one, haptic feelings match with the visual and people feel clearly cut objects. This experiment illustrates the fact that the imagined scene is inferred from the correlation of our perceptions, here between the haptic and the visual. Home Next	Third Pebble Box This experiment uses the force feedback stick from ACROE as interface.  When you click on "Next", you will just have to use the stick to explore the scene. The stick can move in all directions, only on a horizontal plane. Home Next
Situation 1 A scene is being simulated, you can explore it. Home Next	Situation 1 Which situations do this experiment suggest ? (you can give several answers even very different) Home Next
Situation 2 A scene is being simulated, you can explore it.  Home Next	Situation 2 Which situations do this experiment suggest ? (you can give several answers even very different) Home Next



Annexe C

Données pour le Scénario « Ergotic Sounds »

Cette annexe rapporte les réponses des visiteurs du scénario « Ergotic Sounds », l'analyse des résultats et quelques statistiques sur les visites (nombre et durée). Nous présentons également en fin d'annexe les captures d'écran du scénario (page C-15).

1 Statistiques sur le nombre de visites et les temps de visites

Tableau C.1. Nombre de visites

Expérience	1	2	3
Jour 1	44	40	38
Jour 2	23	17	10
Jour 3	53	38	33
Jour 4	36	31	24
Jour 5	43	38	30
Total sur la semaine	199	164	135

Tableau C.2. Durée totale des visites (h :min :sec)

Expérience	1	2	3	Total
Jour 1	0 :39 :01	1 :01 :39	1 :27 :08	3 :07 :48
Jour 2	0 :41 :24	0 :12 :04	0 :16 :42	1 :10 :10
Jour 3	0 :51 :15	0 :51 :01	1 :31 :05	3 :13 :21
Jour 4	0 :55 :18	1 :27 :21	1 :07 :40	3 :30 :19
Jour 5	0 :36 :07	1 :20 :45	0 :54 :12	2 :51 :04
Total sur la semaine	3 :43 :05	4 :52 :50	5 :16 :47	13 :52 :42

Tableau C.3. Nombre de visites de durée supérieure à 30 s par expérience

Expérience	1	2	3	Total
Jour 1	33	27	33	93
Jour 2	16	8	8	32
Jour 3	31	27	31	89
Jour 4	26	28	22	76
Jour 5	22	28	27	77
Total sur la semaine	128	118	121	367

Tableau C.4. Durée moyenne des visites de durée supérieure à 30 s par expérience

Expérience	1	2	3	Total
Jour 1	0 :01 :06	0 :02 :07	0 :02 :36	0 :05 :49
Jour 2	0 :02 :30	0 :01 :23	0 :02 :01	0 :05 :54
Jour 3	0 :01 :26	0 :01 :46	0 :02 :55	0 :06 :07
Jour 4	0 :02 :04	0 :03 :05	0 :03 :03	0 :08 :12
Jour 5	0 :01 :25	0 :02 :47	0 :01 :59	0 :06 :11
Durée moyenne	0 :01 :42	0 :02 :14	0 :02 :31	0 :06 :27

2 Analyse des résultats

Tableau C.5. Dans quel cas avez-vous le mieux réussi la tâche demandée ? (Cas 1 : friction faible ; Cas 2 : friction normale ; Cas 3 : friction exagérée)

	Cas 1	Cas 2	Cas 3	Pas du tout	Total
Jour 1	2	5	1	1	9
Jour 2	0	0	0	0	0
Jour 3	3	11	0	1	15
Jour 4	5	7	1	0	13
Jour 5	8	8	2	3	21
Total sur la semaine	18	31	4	5	58
	31 %	53 %	7 %	9 %	

Tableau C.6. Quel était, selon vous, le cas le plus agréable à jouer ? (Cas 1 : friction faible ; Cas 2 : friction normale ; Cas 3 : friction exagérée)

	Cas 1	Cas 1	Cas3	Total
Jour 1	0	6	1	7
Jour 2	0	0	0	0
Jour 3	4	10	1	15
Jour 4	6	7	1	14
Jour 5	7	10	3	20
Total sur la semaine	17	33	6	56
	30 %	59 %	11 %	

3 Retours des visites

Les pages suivantes contiennent les réponses laissées par les personnes ayant accepté de participer activement à l'expérience.

Exp.1 – sensation	Comment qualifieriez-vous ce que vous avez ressenti ? (Plusieurs réponses sont possibles)
Exp.1 – objet	Quel est, selon vous, quel est l'objet que vous avez manipulé ? (Plusieurs réponses sont possibles)
Exp.2 – différence	Le ressenti du toucher est-il différent de l'expérience précédente ?
Exp.2 – sensation	Comment qualifieriez-vous ce que vous avez ressenti ? (Plusieurs réponses sont possibles)
Exp.2 – objet	Quel est, selon vous, quel est l'objet que vous avez manipulé ? (Plusieurs réponses sont possibles)
Exp.3 – Meilleure réussite	Pensez-vous avoir mieux réussi la tâche dans le cas 1, le cas 2, le cas 3 ou pas du tout ?
Exp.3 – Cas le plus agréable	Quel est le cas le plus agréable à jouer ?
Exp.3 – explication	Pourquoi ?

Exp.1 – sensation	Segment of a circle as virtual wall
Exp.1 – objet	Segment of a circle
Exp.2 – différence	yes
Exp.2 – sensation	Playing a violine
Exp.2 – objet	A violine
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	Like rubbing across a friction surface
Exp.1 – objet	violin string and bow
Exp.2 – différence	yes
Exp.2 – sensation	Like the cello string, because the requirement of continuous motion to sustain the sound altered my movements
Exp.2 – objet	Cello
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	rubber resistance
Exp.1 – objet	rubber mallet
Exp.2 – différence	no
Exp.2 – sensation	bowing
Exp.2 – objet	a bow : audio feedback cue suggested
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	Frottement
Exp.1 – objet	violon
Exp.2 – différence	non
Exp.2 – sensation	vibrations, frottements
Exp.2 – objet	contrebasse
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	blocage non voulu
Exp.1 – objet	une balle
Exp.2 – différence	
Exp.2 – sensation	archet de violoncelle
Exp.2 – objet	
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	a rubber surface
Exp.1 – objet	a rubber stick
Exp.2 – différence	non
Exp.2 – sensation	rugueux
Exp.2 – objet	
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	
Exp.1 – objet	
Exp.2 – différence	non, c'est pareil
Exp.2 – sensation	Selon la pression que j'exerçais, il y avait un son plus fort ou moins fort ; Le retour d'effort se sent comme une corde
Exp.2 – objet	Une contrebasse (surtout pour le son)
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	alternance des forces et frottements
Exp.1 – objet	pédalo
Exp.2 – différence	oui
Exp.2 – sensation	frottement diffus d'une surface contre une autre
Exp.2 – objet	un archet de violoncelle
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 3
Exp.3 – explication	

Exp.1 – sensation	rugosité aléatoire
Exp.1 – objet	glace
Exp.2 – différence	oui
Exp.2 – sensation	le tactile diminue au profit du son
Exp.2 – objet	un violoncelle
Exp.3 – Cas le plus agréable	Aucun
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	

Exp.1 – sensation	rond
Exp.1 – objet	cylindre
Exp.2 – différence	Oui
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	Cas 1
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	Espace de frottement regulier, l’arche d’un violoncelle
Exp.1 – objet	
Exp.2 – différence	non
Exp.2 – sensation	être un peu musicien
Exp.2 – objet	manipule l’arche sur un violoncelle
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	désagréable
Exp.1 – objet	
Exp.2 – différence	oui
Exp.2 – sensation	plus doux plus agréable
Exp.2 – objet	violon
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 3
Exp.3 – explication	cas n°1 : ressemble plus au violon

Exp.1 – sensation	quand on me dit que c’est un violon je dis que c’est vrai mais je préférerais manipuler un stylet en ligne droite
Exp.1 – objet	
Exp.2 – différence	
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	on arrive a avoir des sons différents sans que cela soit trop dur a manipuler

Exp.1 – sensation	Glissement sur une surface. Frottement ; peu de résistance de l'objet frotté ; Il a un comportement visqueux de surface, mais relativement faible. Pas ou peu de frottement sec : en appuyant fort sur l'axe vertical, l'objet ne résiste que peu au déplacement horizontal. J'aurais aimé qu'il résiste plus - qu'il m'empêche de bouger horizontalement.
Exp.1 – objet	
Exp.2 – différence	
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	Aucun
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	Parce qu'il devient possible de "chercher" "tourner autour" d'un plus grand nombre de modes de jeu ; mettre la corde en oscillation "proprement" est plus difficile, mais on peut obtenir plus de comportements.

Exp.1 – sensation	objet métallique
Exp.1 – objet	boule en caoutchouc
Exp.2 – différence	oui
Exp.2 – sensation	on est moins sensible au toucher
Exp.2 – objet	
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	a hard sphere
Exp.1 – objet	I would imagine at least half a sphere ;but why do i imagine a sphere and not just a line ?
Exp.2 – différence	yes, there was vibration, resistance, and sound
Exp.2 – sensation	warmth, vibration, stickiness relative to earlier sensation
Exp.2 – objet	of couse I imagine a bow on a cello, but i have never bowed a cello
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	a round rubber ball
Exp.1 – objet	
Exp.2 – différence	not really
Exp.2 – sensation	sound changes according to movement
Exp.2 – objet	cello
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	vibrations
Exp.1 – objet	tight string
Exp.2 – différence	probably is the same but felt different
Exp.2 – sensation	more tension
Exp.2 – objet	cello
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	case 1 sounded better but case 2 was easier to manipulate

Exp.1 – sensation	
Exp.1 – objet	
Exp.2 – différence	
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	Aucun
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	parce que ça gratte. on sent un grain, une résistance au geste qui offre d'avantage d'expressivité "musicale"...

Exp.1 – sensation	Surface régulière
Exp.1 – objet	boule
Exp.2 – différence	oui
Exp.2 – sensation	surface vibratoire
Exp.2 – objet	violon
Exp.3 – Cas le plus agréable	Cas 1
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	Comportement "doux" de la corde

3. RETOURS DES VISITES

Exp.1 – sensation	Ça vibre et sa bloque un peu
Exp.1 – objet	
Exp.2 – différence	oui
Exp.2 – sensation	C'est plus fort et dans le casque on entend comme un violoncelle
Exp.2 – objet	L'archet d'un violoncelle
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	Il y avait moins de frottement et moins d'hésitation

Exp.1 – sensation	Une granularité et un collement
Exp.1 – objet	un truc plastique sur du verre
Exp.2 – différence	oui
Exp.2 – sensation	moins collant
Exp.2 – objet	un archet de violon
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	sensation de toucher une matière rugueuse et vibrante
Exp.1 – objet	une corde de violoncelle
Exp.2 – différence	il est le même sans qu'on associe la vibration au son
Exp.2 – sensation	une sensation de professionnelle du violoncelle !
Exp.2 – objet	un violoncelle ? ou une contrebasse ?
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	parce que c'est le mouvement est le plus fluide

Exp.1 – sensation	Sort of friction, can't really see much difference between the device switched on and off.
Exp.1 – objet	Little stones in a bucket.
Exp.2 – différence	yes
Exp.2 – sensation	
Exp.2 – objet	String
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 3
Exp.3 – explication	The sound was the nicest and the feeling (response from the device) as well.

Exp.1 – sensation	une vibration
Exp.1 – objet	un objet vibrant
Exp.2 – différence	pas forcément, le retour audio indique qu’il y a des différences d’intensité que je n’ai pas ressenti au niveau du toucher
Exp.2 – sensation	un contact rugueux
Exp.2 – objet	un instrument à cordes frottées
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	
Exp.3 – explication	Le cas 3 est très rugueux (trop), et le cas 1 moins rugueux. Le cas 2 fournit un retour plus équilibré

Exp.1 – sensation	des vibrations. retour léger, impression de couvrir une onde.
Exp.1 – objet	une demi sphère
Exp.2 – différence	
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	accrochement
Exp.1 – objet	corde
Exp.2 – différence	peut-être
Exp.2 – sensation	frottement
Exp.2 – objet	violoncelle
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	Tous on passe du frottement au pizzicato

Exp.1 – sensation	léger
Exp.1 – objet	cube de liège
Exp.2 – différence	non
Exp.2 – sensation	vibration et tension
Exp.2 – objet	une corde tendue
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	vibrations / force s’opposant au mouvement
Exp.1 – objet	
Exp.2 – différence	non
Exp.2 – sensation	
Exp.2 – objet	
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	plus de facilité de mouvement

Exp.1 – sensation	surface rugueuse
Exp.1 – objet	balle de caoutchouc
Exp.2 – différence	non
Exp.2 – sensation	cordes
Exp.2 – objet	violon
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	vibration
Exp.1 – objet	
Exp.2 – différence	oui
Exp.2 – sensation	je joue du violoncelle
Exp.2 – objet	archet
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	j’ai ressenti des vibrations le long de mon bras ressemblant étrangement à ce qu’on peut ressentir en se déplaçant sur une corde
Exp.1 – objet	guitare, violon, harpe
Exp.2 – différence	le même mais plus fort
Exp.2 – sensation	pareil que précédemment comme si je tenais un archet
Exp.2 – objet	violoncelle
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	texture
Exp.1 – objet	surface of a ball
Exp.2 – différence	not sure ; sound might have made a difference
Exp.2 – sensation	string of a cello
Exp.2 – objet	bow of a cello
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	
Exp.3 – explication	it is nice and strong ; but not too rough

Exp.1 – sensation	quelque chose d'élastique et rugueux
Exp.1 – objet	élastique
Exp.2 – différence	oui
Exp.2 – sensation	frottement
Exp.2 – objet	corde frottée de contrebas
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	pas de frottement dans le cas 1 du coup meilleure "réussite" à manipuler l'archet. MAIS le cas 2 est beaucoup plus réaliste.

Exp.1 – sensation	moving and feeling the only the handle
Exp.1 – objet	something like a pencil, a brush
Exp.2 – différence	definitely
Exp.2 – sensation	first playing a ball and after playing some musical instrument
Exp.2 – objet	I guess a musical instrument
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	doux, sphérique
Exp.1 – objet	une boule de poils
Exp.2 – différence	non
Exp.2 – sensation	
Exp.2 – objet	balle qui rebondit et archet
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	vibrations à relativement hautes fréquences
Exp.1 – objet	corde de guitare
Exp.2 – différence	oui
Exp.2 – sensation	vibrations légères, son grave
Exp.2 – objet	archet de violon
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	Le mouvement est plus régulier que dans les autres cas

Exp.1 – sensation	Le mouvement est plus régulier que dans les autres cas
Exp.1 – objet	a metal plate
Exp.2 – différence	yes
Exp.2 – sensation	like a cello ; it also felt like a round object
Exp.2 – objet	the string of a cello
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 2
Exp.3 – explication	I believe I can control my hand better when there is a little friction

Exp.1 – sensation	facile a manipuler
Exp.1 – objet	je ne sais pas
Exp.2 – différence	oui c'est plus représentatif
Exp.2 – sensation	d'étrange et perturbant, vertigineux
Exp.2 – objet	une contrebasse
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 3
Exp.3 – explication	on sent les vibrations mais sans qu'elles soient trop fortes.

Exp.1 – sensation	curved plane, going up towards the front
Exp.1 – objet	curved plane
Exp.2 – différence	similar area, but maybe with added vibration
Exp.2 – sensation	bowing a string
Exp.2 – objet	a kind of flat string
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	there is less interuption to the force

Exp.1 – sensation	Une sorte de frottement
Exp.1 – objet	levier de vitesse
Exp.2 – différence	oui
Exp.2 – sensation	Une vibration
Exp.2 – objet	Une corde
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	impression élastique genre trampoline matelas
Exp.1 – objet	
Exp.2 – différence	oui
Exp.2 – sensation	la sensation change
Exp.2 – objet	contrebasse
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	on l'impression que l'archet quitte le contact dans le cas 1. les vibrations sont plus fortes dans les autres cas.

Exp.1 – sensation	I like the mechanical design, it works well. Vibrations did not help discovering the shape.
Exp.1 – objet	a sphere
Exp.2 – différence	yes
Exp.2 – sensation	
Exp.2 – objet	a cello
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	first time nothing
Exp.1 – objet	half of sphere
Exp.2 – différence	yes
Exp.2 – sensation	something is there
Exp.2 – objet	it is a hil
Exp.3 – Cas le plus agréable	Cas 2
Exp.3 – Meilleure réussite	Cas 3
Exp.3 – explication	you have feedback ; but for using a long time it is better in case 1

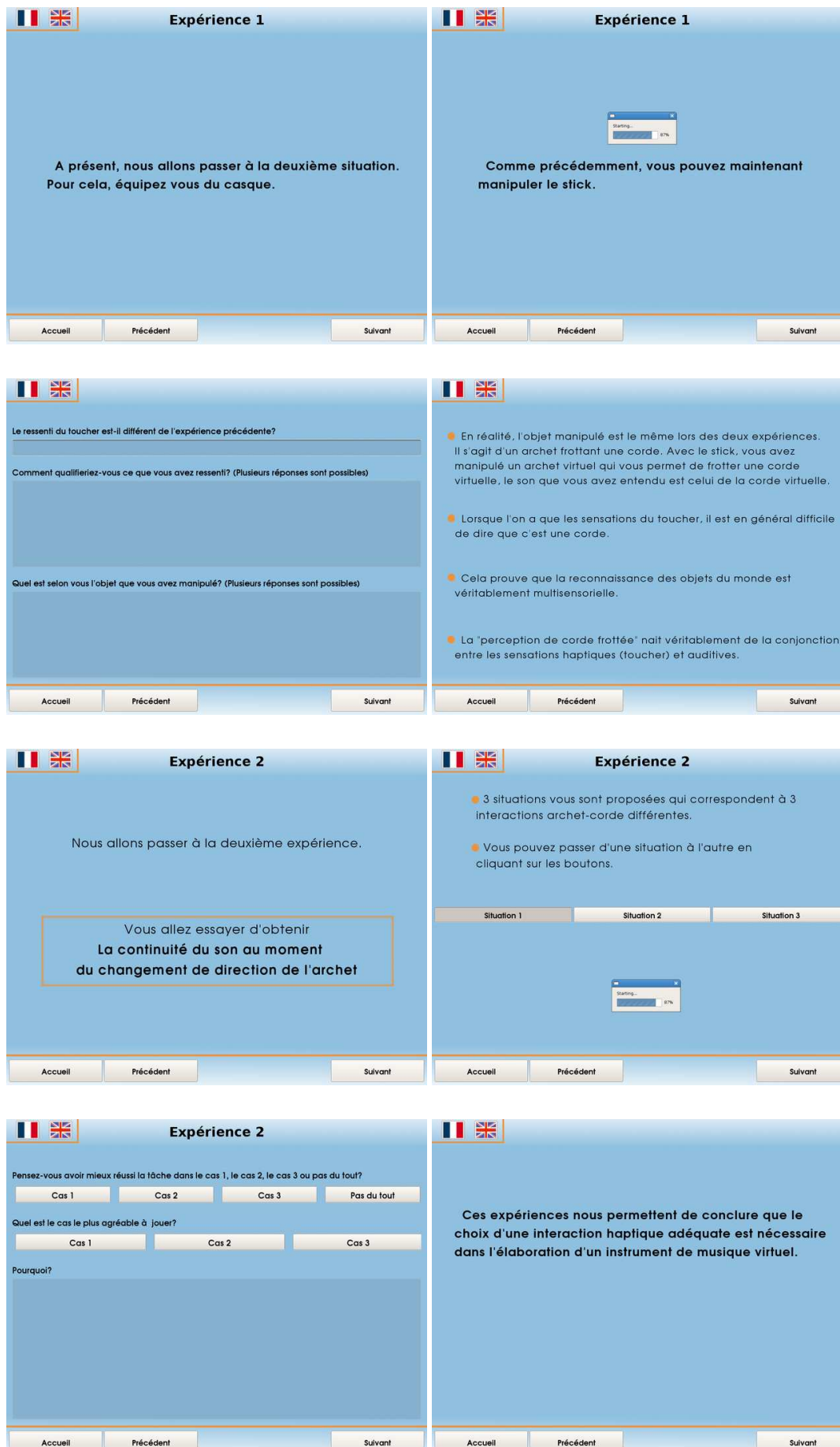
Exp.1 – sensation	étrange et bizarre je n'arrive pas à deviner
Exp.1 – objet	je n'en sais rien
Exp.2 – différence	Oui
Exp.2 – sensation	Particulier mais réaliste
Exp.2 – objet	violoncelle ou contrebasse
Exp.3 – Cas le plus agréable	
Exp.3 – Meilleure réussite	
Exp.3 – explication	

Exp.1 – sensation	surface arrondie et rugueuse
Exp.1 – objet	aucune idée
Exp.2 – différence	oui (approche générale qui change)
Exp.2 – sensation	le son permet de déterminer la nature de l'objet
Exp.2 – objet	violoncelle ou contrebasse
Exp.3 – Cas le plus agréable	Cas 3
Exp.3 – Meilleure réussite	Cas 1
Exp.3 – explication	le cas 1 est plus « facile » à jouer mais il offre moins de subtilités de jeu.

4 Captures d'écran du scénario

The image displays four screenshots of a software interface for a scenario, arranged in a 2x2 grid. Each screenshot shows a different stage of the user experience.

- Top Left (Contribution):** A screen titled 'Contribution' with a header showing French and UK flags. It asks the user to participate actively and provides a form with fields for 'Nom', 'Prénom', 'Profession', and 'Hobbies'. Navigation buttons 'Accueil', 'Précédent', and 'Suivant' are at the bottom.
- Top Right (Présentation):** A screen titled 'Présentation' with the same header. It contains instructions and a video of a hand manipulating a stick. Text includes: 'Sur votre droite se trouve le stick à retour d'effort', 'Toutes les expériences requièrent son utilisation, il convient de le manipuler comme suit:', 'L'exploration s'effectue dans un plan perpendiculaire au sol. Libre à vous de saisir le stick comme bon vous semble. A priori le contrôle est facilité en le saisissant par la boucle. Veillez à le manipuler avec précautions.', and 'Vous allez mener deux expériences. Dans la première, deux cas vont vous être présentés successivement. Cliquez sur le bouton Suivant pour expérimenter la première situation.' Navigation buttons are at the bottom.
- Bottom Left (Présentation):** A screen titled 'Présentation' with the same header. It shows a 'Starting...' progress bar at 30% and the text: 'Vous pouvez maintenant manipuler le stick, comme expliqué précédemment.' Navigation buttons are at the bottom.
- Bottom Right (Expérience 1):** A screen titled 'Expérience 1' with the same header. It asks: 'Comment qualifieriez-vous ce que vous avez ressenti? (Plusieurs réponses sont possibles)' and 'Quel est selon vous l'objet que vous avez manipulé? (Plusieurs réponses sont possibles)'. There are two large text input areas for answers. Navigation buttons are at the bottom.





Annexe D

Le Formalisme CORDIS-ANIMA

*« CORDIS-ANIMA est le nom d'un système pour la modélisation et la simulation numérique en temps-réel, d'objets physiques manipulables, audibles et visibles. »
C. Cadoz, A. Luciani et J.L. Florens [Cadoz et al., 1990]*

1 Introduction

CORDIS-ANIMA est à la fois un formalisme et un langage de modélisation physique particulière basé sur la mécanique newtonienne, et discrétisé à la fois dans le temps et dans l'espace. Les objets CORDIS sont construits par assemblage de blocs fonctionnels, appelés modules, qui constituent les briques physiques du modèle.

La représentation d'objets physiques par l'ordinateur implique que le formalisme de modélisation choisi soit assujéti à une discrétisation temporelle, en raison de la nature du support algorithmique contenu dans l'ordinateur.

Pour que cette représentation, pour l'instant enclose dans l'ordinateur, puisse être rendue accessible à l'utilisateur, nous devons utiliser des transducteurs. Les transducteurs permettent de connecter le monde numérique de l'ordinateur au monde physique et sensori-moteur de l'utilisateur : « dans sa définition basique, un transducteur est une entité matérielle qui peut transformer un phénomène physique donné en un autre phénomène physique de nature différente » [Cadoz, 2004]. Entre autres, les transducteurs sont donc les outils qui nous permettent de traduire les informations traitées par l'ordinateur sous la forme de signaux électriques en phénomènes lumineux (écrans), mécaniques (haut-parleurs), etc. Cependant la mécanique newtonienne traduit les interactions entre les objets par des relations duales entre variables intensives (la force, la pression, la température...) et des variables extensives (la position, la vitesse, la charge électrique...). Il n'est possible de traduire cette relation duale qu'avec l'aide de deux types de transducteurs : un capteur et un actionneur. Cela signifie que d'un point de vue extérieur, le modèle simulé par l'ordinateur peut se résumer à un point de communication associant une variable intensive et une variable extensive.

Le formalisme CORDIS est construit sur cette dualité entre le transducteur-capteur, donnant un signal d'entrée au simulateur, et le transducteur-effecteur qui utilise les signaux de sortie du simu-

lateur : toutes les entités définies dans CORDIS possèdent au moins un point de communication entrant, et un point de communication sortant. En outre, la mécanique newtonienne associant toujours de manière duale des variables extensives et des variables intensives (l'association de une ou plusieurs variables extensives définissant la valeur d'une variable intensive, et inversement), deux types de points de communication complémentaires associant des variables intensives et extensives sont définis dans le langage CORDIS. Ces deux types de points de communication constituent la base du formalisme de modélisation CORDIS (figure D.1) :

- Les points de communication dits « M », dont l'entrée est une variable intensive (une force) et la sortie une variable extensive (une position)
- Les points de communication dits « L », complémentaires des points M, dont l'entrée est une variable extensive (une position), et la sortie une variable intensive (une force).

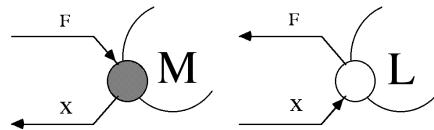


Figure D.1. Représentation schématique des points de communication M et L

2 Topologie

Les règles de connexion entre les points de connexion de type M et les points de connexion de type L sont construites sur la base de l'observation des relations entre variables intensives et extensives, qui leur servent de support de définition :

- La force est une variable intensive : il est possible d'additionner plusieurs forces à l'entrée d'un point M.
- À l'inverse, un point L ne reçoit qu'une position — variable intensive —, et ne permet pas la sommation.

En conséquence, le langage CORDIS définit les règles de connexion suivantes (Figure 1.3) :

1. La connexion entre deux points de communication consiste à utiliser la sortie de l'un comme entrée de l'autre et réciproquement. En conséquence, ne sont connectables que des points de types opposés : un point de communication M avec un point de communication L.
2. On peut connecter plusieurs points de communication L à un même point de communication M. La force entrante dans le point M est la somme des forces sortantes des divers points L connectés à ce point M. La position entrante dans les divers points L est la position sortante du point M commun auquel ils sont connectés.
3. Il est interdit de connecter plusieurs points de communication M à un même point de communication L.

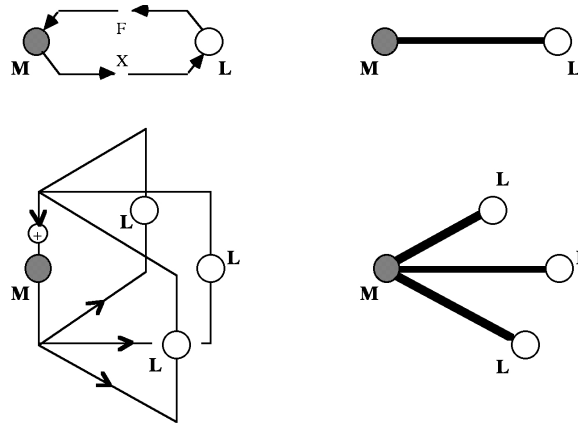


Figure D.2. Règles de connexion des points de communication CORDIS

3 Module

Le module CORDIS est une entité algorithmique indépendante. Chaque module est relié au reste du modèle par ses points de communication. Ce sont les relations entre points de communications qui définiront les relations entre les différents modules d'un même modèle CORDIS. De l'observation des relations entre points de communications M et L, il apparaît rapidement qu'une base minimale permettant la création d'un système générique est constituée de deux modules élémentaires : un premier constitué d'un seul point M, et d'un second constitué de deux points L (figure D.3).

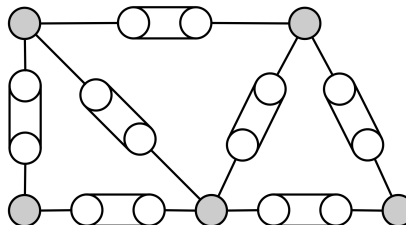


Figure D.3. Un exemple de réseau CORDIS-ANIMA construit à partir des deux types de modules élémentaires.

Dans tout système newtonien, nous pouvons distinguer des comportements propres, comme l'inertie, et des comportements d'interaction, comme la gravité ou l'élasticité. Dans CORDIS, les comportements propres et comportements d'interaction sont dédiés à des modules spécifiques : les éléments de comportement propres seront appelés <MAT>, représentant des « points matériels », reliés entre eux par des éléments de comportement d'interaction appelés <LIA>, représentant les comportements d'interaction (figure D.4).

- Le <MAT> est cohérent avec le point de communication M. Le plus petit module de type <MAT> comporte un point M.
- Le <LIA> est cohérent avec le point de communication L. Le plus petit module de type <LIA> comporte deux points L.

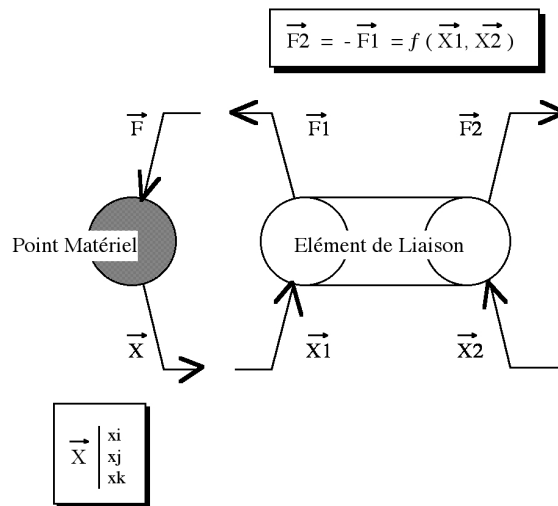


Figure D.4. Modules <MAT> et <LIA>

4 T-Simulation

Le principe de la T-Simulation est le schéma temporel général adopté pour représenter le principe de fonctionnement d'un algorithme de simulation [Cadoz *et al.*, 1990]. Ce principe est né des observations suivantes :

- Les variables utilisées pour l'interprétation des algorithmes dans un ordinateur ayant nécessairement une forme numérique, les signaux d'entrée et de sortie du processus de simulation auront aussi un format numérique : ce seront des signaux échantillonnés.
- Le temps d'exécution d'un algorithme exécuté par ordinateur ne peut être nul.

En conséquence, le schéma temporel adopté pour le principe de T-Simulation est le suivant :

- Les communications entre l'extérieur et l'intérieur de la machine ne se font qu'aux instants d'échantillonnage déterminés par un signal d'horloge isochrone. Toutes les variables doivent donc être rendues accessibles à l'interface de communication de la machine pour des instants correspondant au début ou à la fin d'un pas d'échantillonnage.
- On traite en entrée les signaux échantillonnés aux conditions de Shannon à la période T (constante).
- On produit en sortie des signaux échantillonnés selon les mêmes conditions. Ces signaux sont le résultat de l'application de l'algorithme de simulation sur les signaux d'entrée.
- Une relation spécifique est établie entre les entrées et les sorties : les instants d'entrée et de sortie sont synchrones, et entre les entrées et les sorties qui en résultent, il s'écoule une seule période T .

Dans ces conditions, les objets physiques dont on peut donner une simulation valide sont dits « T-Simulables ». Cette notion a un certain nombre d'incidences, en particulier :

- Pour un objet donné, il existe une valeur T_e au-delà de laquelle un objet n'est plus T-Simulable pour tout T choisi. Cela signifie que pour pouvoir T-Simuler cet objet, nous devons choisir une fréquence de simulation F_s telle que $F_s \geq 1/(2T_e)$. Si on s'avise de simuler tout de même cet objet avec T ne respectant pas cette condition, il s'ensuit une dérive dont la gravité dépendra

du rapport entre T_e et T . Cette dérive est comparable à la conséquence du sous-échantillonnage dans le traitement des signaux numériques.

- Certains objets ne sont pas T-Simulables, aussi petit que soit le T choisi. Le passage du modèle continu au modèle discret introduit des déformations dans la représentation de l'objet simulé [Florens et Urma, 2006].

On peut alors donner la définition de la T-Simulabilité : un objet sera dit T-Simulable si on peut exprimer ses variables de sortie S à l'indice n en fonction des entrées E et des variables d'état X aux indices $n, n-1, n-2 \dots$. Un objet sera T-Simulable s'il peut être exprimé par une suite numérique explicite de la forme :

$$S(n) = \sum \alpha_i E(n-i) + \sum \beta_i X(n-i) + \gamma, 0 \leq i \leq n \quad (D.1)$$

Le principe de T-Simulation impose les contraintes suivantes quant à son application dans le langage CORDIS, et son implémentation matérielle dans un simulateur :

- Les schémas algorithmiques choisis doivent être explicites, c'est-à-dire que leur exécution ne prend qu'un nombre fini d'opérations élémentaires, et que le nombre des opérations élémentaires nécessaires peut être connu à l'avance (à l'inverse, les formulations algorithmiques implicites ne vérifient pas cette propriété). Seule la formulation explicite des algorithmes permet d'assurer qu'un algorithme de simulation peut se terminer dans la période T fixée.
- La plateforme de simulation doit permettre de choisir une période T suffisamment petite pour la représentation des objets physiques choisis. En particulier, la simulation de phénomènes vibratoires dans le domaine acoustique impose une fréquence de simulation entre 10 et 50 kHz selon l'objet simulé.

5 Interface simulateur/extérieur

Considérons maintenant les relations entre un modèle CORDIS et l'extérieur du simulateur. Nous avons vu que ce sont les transducteurs qui assurent la fonction d'interface entre le monde sensorimoteur de l'utilisateur, et le monde des signaux numériques du simulateur.

5.1 Transducteur gestuel rétroactif

Dans le cas de l'interfaçage du modèle simulé au monde physique par un transducteur gestuel rétroactif (TGR), deux solutions s'offrent à nous suivant la technologie utilisée par le TGR :

5.1.1 Le TGR fonctionne en mode impédance

Le TGR produit en entrée du simulateur des signaux de type position et reçoit de la sortie du simulateur des signaux de type force. Selon le formalisme CORDIS, le TGR est consistant avec un point de communication de type M. Il se connecte donc à un ou plusieurs points de type L du modèle (figure D.5).

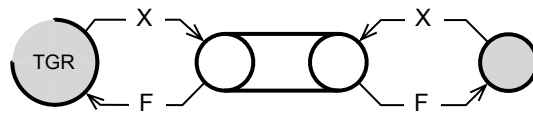


Figure D.5. Connexion d'un TGR en mode impédance au modèle CORDIS

5.1.2 Le TGR fonctionne en mode admittance

À chaque instant d'échantillonnage, le TGR produit en entrée du simulateur des signaux de type force et reçoit de la sortie du simulateur des signaux de type position. Selon le formalisme CORDIS, le TGR est consistant à un point de communication de type L. Il se connecte donc à un point de type M (figure D.6).

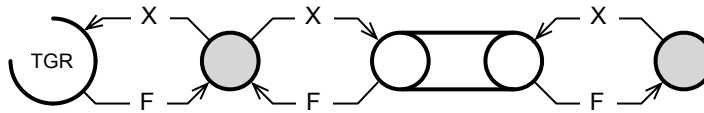


Figure D.6. Connexion d'un TGR en mode admittance au modèle CORDIS

5.2 Sortie sonore

Dans notre monde physique, le phénomène sonore se traduit par la mise en vibration de la matière. Du point de vue du modèle CORDIS, cela est équivalent à observer le mouvement des points matériels, donc des modules de type <MAT>. Pour produire du son avec un modèle CORDIS, nous exportons la sortie en position d'un point de communication de type M à l'extérieur du modèle (figure D.7).

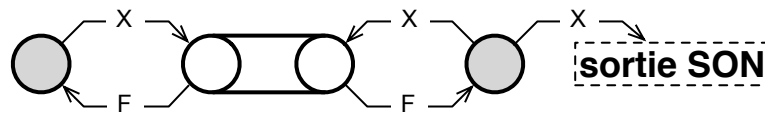


Figure D.7. Connexion d'une sortie sonore à un modèle CORDIS

5.3 Sortie visuelle

Le processus de la génération de la visualisation est beaucoup plus complexe que celui du son, et constitue une problématique de recherche à part entière pour le groupe de recherche ACROE & ICA. À la différence du phénomène sonore, le phénomène visuel en effet ne met pas seulement en jeu des phénomènes mécaniques, mais aussi l'interaction de la lumière avec la matière de l'objet. Outre le phénomène mécanique, la représentation visuelle d'un objet CORDIS nécessite donc un processus supplémentaire, chargé de produire la forme visuelle de l'objet à partir de son comportement mécanique. Suivant le processus choisi pour la génération de la forme visuelle, le

concepteur du modèle peut être amené à utiliser des combinaisons variables des signaux de type position et des signaux de type force.

6 Séquence d'un pas de simulation

Les modules <MAT> et <LIA> correspondent à deux types algorithmiques distincts. Le module de type <MAT> est consistant à un point M, il prend en entrée des forces et retourne des positions :

$$\Phi_{\langle \text{MAT} \rangle}(F) \rightarrow X \quad (\text{D.2})$$

Le module de type <LIA> est consistant à un point L, il prend en entrée des positions et retourne des forces :

$$\Phi_{\langle \text{LIA} \rangle}(X) \rightarrow F \quad (\text{D.3})$$

Ainsi, tous les modules d'un même type peuvent être calculés indépendamment les uns des autres sans relation de causalité. Un algorithme CORDIS se déroule donc en deux phases distinctes, où tous les modules du même type sont calculés. Les deux solutions consistant à calculer d'abord le comportement de tous les modules de type <MAT> puis ensuite de tous les modules de type <LIA>, ou inversement les modules <LIA> puis <MAT> sont équivalentes du point de vue algorithmique. C'est la connexion du système de simulation avec les signaux d'entrée et de sortie du simulateur qui va déterminer l'ordre de ces deux phases :

- En mode impédance, le transducteur gestuel est vu depuis le modèle CORDIS comme un point de communication M. Au coup d'horloge CLK, il fournit en entrée du modèle des positions et prend des forces. L'utilisation d'un transducteur gestuel fonctionnant en impédance oblige donc à calculer dans un premier temps la phase <MAT>, puis la phase <LIA> (figure D.8).
- En mode admittance, le transducteur gestuel est vu depuis le modèle CORDIS comme un point de communication L. Au coup d'horloge CLK, il fournit en entrée du modèle des forces et prend des positions. L'utilisation d'un transducteur gestuel fonctionnant en mode admittance induit le schéma inverse (figure D.9).

Pour réaliser notre simulateur, nous utiliserons les TGR de la panoplie ERGOS© [Florens *et al.*, 2004, erg, 2008]. Les systèmes ERGOS fonctionnent en mode impédance ; en conséquence, l'exécution d'un pas de simulation d'un modèle CORDIS se fera suivant le schéma suivant (figure D.10) :

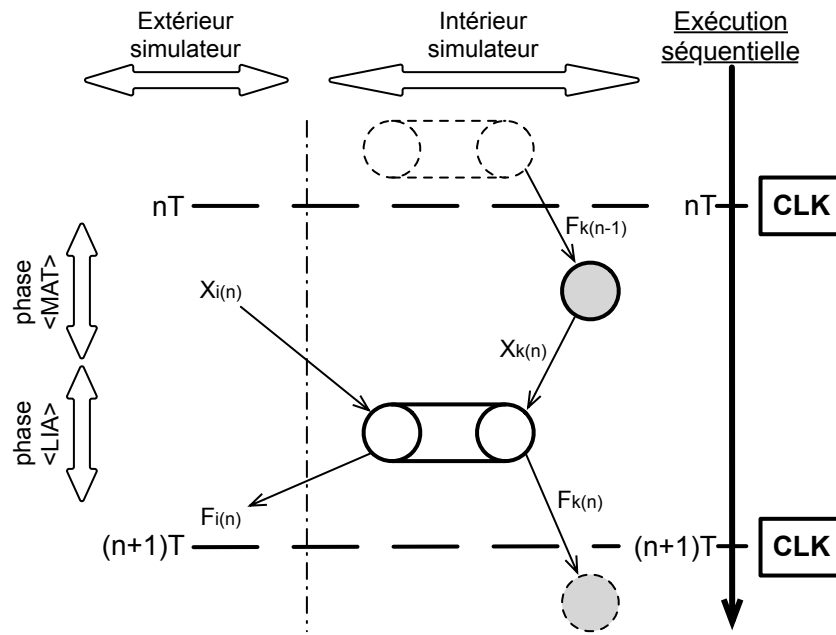


Figure D.8. Séquence des phases <LIA> et <MAT> avec un TGR en mode impédance

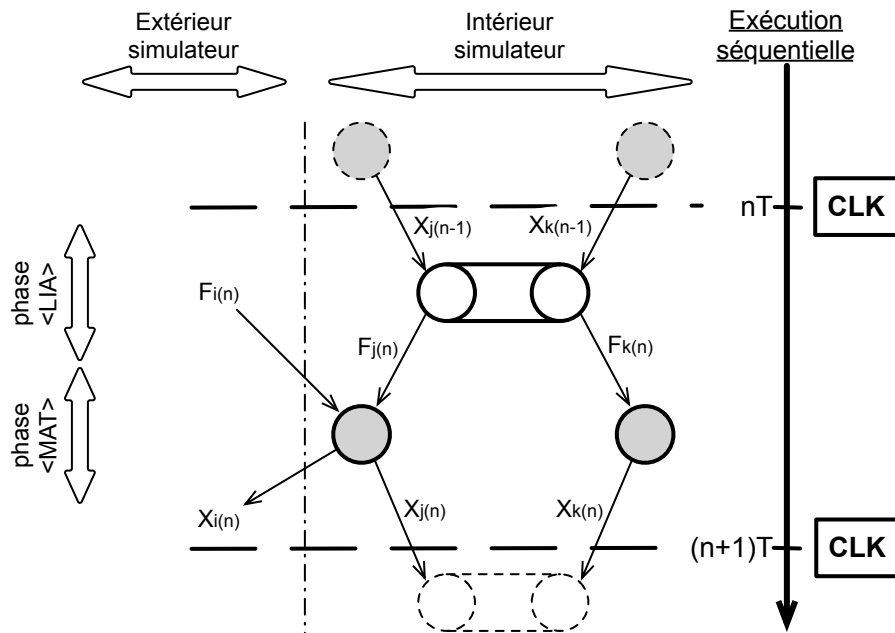


Figure D.10. Séquence d'un pas de simulation CORDIS

La valeur minimum nécessaire pour le pas d'échantillonnage T sera alors définie par la somme :

- Du transfert des signaux d'entrée, et du transfert simultané des signaux de sortie calculés à la phase $(n-1)T$
- Du temps minimal nécessaire pour l'exécution de la phase <MAT>
- Du temps minimal nécessaire pour l'exécution de la phase <LIA>

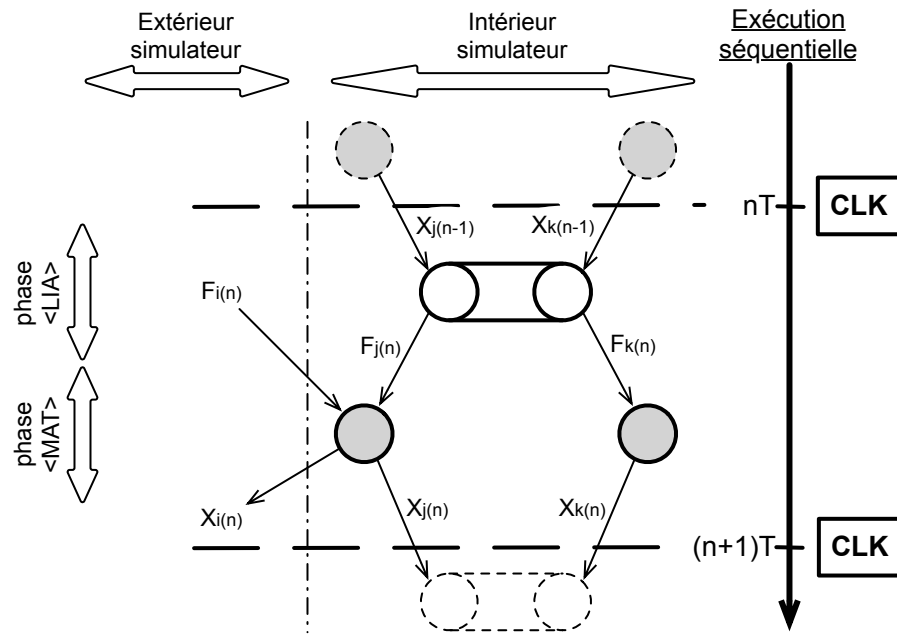


Figure D.9. Séquence des phases <MAT> et <LIA> avec un TGR en mode admittance

Cette valeur définit des contraintes temporelles pour l'implémentation du formalisme CORDIS dans une plateforme de simulation en temps réel.

7 Les modules

Le langage CORDIS nous permet de représenter un très grand nombre d'objets physiques par la mise en réseau d'éléments de type <MAT>/<LIA>. Cette distinction entre éléments matériels et éléments de liaison au sein du modèle n'est pourtant encore qu'une typologie. En exprimant le formalisme newtonien sous cette forme, on peut extraire plusieurs fonctionnements types : inertie, élasticité, viscosité, etc.

7.1 Modules élémentaires

7.1.1 MAS

Représente l'élément matériel élémentaire : il représente une masse ponctuelle. Le module MAS a pour paramètre M : la masse .

7.1.2 SOL

Ce module représente un point fixe, c'est-à-dire un élément matériel d'inertie infinie. Le SOL n'a pas de paramètres.

7.1.3 RES

Ce module permet de modéliser les comportements élastiques entre deux éléments matériels. Le module RES a pour paramètres la raideur K et la longueur à vide L_0 .

7.1.4 FRO

Modélise les comportements visqueux. Le module FRO a pour paramètres la viscosité Z et la longueur à vide L_0 .

7.1.5 BUT

Le BUT est un module d'interaction non linéaire. Il permet de modéliser toutes les interactions de butée mécanique. Les paramètres K et Z correspondent respectivement à la raideur et la viscosité lorsqu'il y a interaction visco-élastique. La distance à partir de laquelle l'interaction mécanique est mise en jeu est paramétrée par le seuil S : en d'autres termes, il y a interaction viscoélastique seulement si la distance entre les deux éléments <MAT> que le BUT connecte est inférieure au seuil.

7.1.6 LNL

C'est une généralisation des interactions non linéaires. Ce module permet de définir des profils d'élasticité et de viscosité quelconques. Cela revient à exprimer les relations $\Phi_K(d)$ et $\Psi_Z(v)$ exprimant la force respectivement en fonction de la distance (comportement de type élasticité) et de la vitesse (comportement de type viscosité). Remarquons que, dans le module LNL, les paramètres équivalents K et Z sont intrinsèques à la définition de Φ_K et Ψ_Z .

7.2 Modules intégrés

Enfin, les modules CEL et REF sont des modules intégrés : il s'agit de la combinaison de plusieurs modules élémentaires, mais ces modules sont respectivement consistants à des types <MAT> et <LIA> du point de vue de leurs points de communication avec le reste du modèle :

7.2.1 REF

Ce module est la combinaison des modules RES et FRO : il représente les comportements viscoélastiques. Ses paramètres sont K , Z et la longueur à vide L_0 .

7.2.2 CEL

La CEL représente la combinaison d'un module SOL et d'un module MAS, connectés par un REF. C'est en fait l'oscillateur élémentaire universel (une masse attachée à un point fixe par un ressort-frottement). Ses paramètres sont M , K , Z et L_0 .

8 Conditions initiales

Lors de la simulation d'un module CORDIS, pour le calcul du pas de simulation n , il est nécessaire de connaître l'état de certaines variables au pas de simulation $n - 1$. Étant donné que nous avons choisi de commencer le calcul de la simulation par la phase <MAT>, nous devons connaître la valeur des forces au pas $n - 1$ (figure D.8), ainsi que des positions retardées aux pas $n - 1$ et $n - 2$.

Pour le premier pas de simulation à $n = 0$, nous choisissons de mettre toutes les forces F_n égales à zéro. Les positions retardées sont déterminées par l'utilisateur lors du paramétrage du modèle. En conséquence, dans un modèle CORDIS, seuls les modules de type <MAT> se voient attribuer des conditions initiales à leurs variables d'état (de la forme $[X]$).

TITRE

« HAPTIC PROCESSOR UNIT » : VERS UNE PLATE-FORME TRANSPORTABLE POUR LA SIMULATION TEMPS-RÉEL SYNCHRONE MULTISENSORIELLE

RÉSUMÉ

Ce travail se situe dans le domaine de l'Interaction Personne-Système, et plus particulièrement dans celui de la simulation instrumentale multisensorielle telle que conceptualisée par le groupe de recherche ACROE & ICA, qui nécessite un couplage fort homme-instrument.

La première partie de cette thèse présente les différents degrés d'intégration du geste dans l'ordinateur, puis propose une approche fonctionnelle des technologies pour le retour d'effort. Nous dégagons de cette analyse les grandes approches qui se partagent actuellement le champ de la recherche « haptique ».

Nous présentons ensuite une étude sur les différents composants matériels et logiciels nécessaires à la chaîne de simulation haptique, ainsi que les différentes approches utilisées pour connecter un système à retour d'effort à un processus de simulation en temps réel. L'analyse du rôle des composants de la chaîne de simulation et de leurs relations permet de formaliser le concept de « *Haptic Processor Unit* ». Ce composant permet en particulier de garantir les conditions de réactivité propres à la simulation multisensorielle. La nouvelle architecture de simulation multisensorielle que nous avons réalisée, ERGON_X, met en œuvre le concept de HPU. ERGON_X est un simulateur compact et transportable, et permet d'utiliser des fréquences de simulation jusqu'à 44 100 Hz.

La dernière partie présente la validation de la plate-forme de simulation ERGON_X. Elle est essentiellement orientée vers l'implantation de nouveaux modèles, utilisés dans le cadre d'un travail de recherche sur la situation instrumentale médiatisée. Le « E » est un modèle de démonstration des performances de la technologie ERGOS que la nouvelle architecture de simulation permet d'exploiter pleinement. Les modèles de tapping (percussion) et de pâtes déformables ont permis d'avancer des résultats sur l'interaction homme-objet, et valident le simulateur comme un outil pour l'expérimentation psychophysique. Les *Enactive Emblematic Scenarii* « *Ergotic Sounds* » (frottement d'archet) et « *Pebble Box* » (la boîte à cailloux) sont des illustrations du concept de l'Enaction. Elles valident l'utilisation de l'architecture de simulation comme une plate-forme pour l'expérimentation et ouvrent de nouvelles perspectives de recherche sur l'enaction et la notion de présence en simulation interactive.

MOT-CLEFS

interface haptique, transducteur gestuel rétroactif, retour d'effort, interaction multi-modale, simulation instrumentale multisensorielle, modélisation physique, temps-réel synchrone, système embarqué

TITLE

“HAPTIC PROCESSOR UNIT” : TOWARDS A TRANSPORTABLE PLATFORM FOR SYNCHRONOUS MULTISENSORY REAL-TIME SIMULATION

ABSTRACT

This work is related to the field of Human-Computer Interaction, and particularly to the field of multisensory instrumental simulation, as conceptualized by the research group ACROE & ICA, and which needs a strong coupling between the human and the instrument.

The first part of this thesis presents various degrees of the integration of gesture in computer uses, then develops a functional approach of force feedback technologies. This analysis elicits the mainstreams that are currently sharing the field of haptics research.

We then present a study of the hardware and software components that are used in haptic simulation, and the various approaches used to connect a force feedback device to a real time modelling system. The analysis of the role of each of the components in the simulation chain and their relationships allowed us to conceptualize the “Haptic Processor Unit”. This component guarantees in particular the conditions of reactivity that are required for multisensory simulation. The new simulation architecture that we designed in this work, named ERGON_X, implements the concept of HPU. ERGON_X is a compact and transportable simulator, and handles simulation frequencies up to 44 100 Hz.

The third part presents the validation of the simulation platform ERGON_X. It mainly focuses on the design of new models, which were used in the framework of the research carried on by ACROE & ICA about instrumental interaction. The “E” is a model demonstrating the capabilities of the ERGOS technology, which is now fully exploitable thanks to this new simulation architecture. The models of tapping and of deformable paste allowed us to bring new results on human-object interaction, and validate the simulator as a tool for psychophysical experimentation. The *Enactive Emblematic Scenarii* “*Ergotic Sounds*” and “*Pebble Box*” illustrate the conception of Enaction. They validate the use of our simulation architecture as an experimental platform and lead us to a paradigm shift from “instrumental interaction” to “enactive interaction”.

KEYWORDS

haptic device, gesture retroactive transducer, force feedback, multi-modal interaction, instrumental multisensory simulation, physical modelling, synchronous real time, embedded system
