



HAL
open science

Collaborative hands-on training on haptic simulators

Angel Ricardo Licona Rodriguez

► **To cite this version:**

Angel Ricardo Licona Rodriguez. Collaborative hands-on training on haptic simulators. Automatic. Université de Lyon, 2020. English. NNT : 2020LYSEI018 . tel-03078482

HAL Id: tel-03078482

<https://theses.hal.science/tel-03078482>

Submitted on 16 Dec 2020

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.



N° d'ordre NNT : 2020LYSEI018

THÈSE DE DOCTORAT DE L'UNIVERSITÉ DE LYON

opérée au sein de
l'Institut National des Sciences Appliquées de Lyon

École Doctorale N° 160
Electronique, Electrotechnique, Automatique (EEA)

Spécialité de doctorat : Automatique

Soutenue publiquement le 12/03/2020, par :
Angel Ricardo Licona Rodriguez

Collaborative Hands-on Training on Haptic Simulators

Devant le jury composé de :

Pierre VIEYRES	Professeur des Universités Université d'Orléans	Rapporteur
Emmanuel PROMAYON	Professeur des Universités Université Grenoble alpes	Rapporteur
Véronique PERDEREAU	Professeur des Universités Université Pierre et Marie CURIE	Examineur
Minh Tu PHAM	Maître de Conférences HdR INSA Lyon	Directeur de thèse
Arnaud LELEVÉ	Maître de Conférences HdR INSA Lyon	Co-directeur de thèse
Damien ÉBÉRARD	Maître de Conférences INSA Lyon	Co-encadrant

Département FEDORA – INSA Lyon - Ecoles Doctorales – Quinquennal 2016-2020

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	<u>CHIMIE DE LYON</u> http://www.edchimie-lyon.fr Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage secretariat@edchimie-lyon.fr INSA : R. GOURDON	M. Stéphane DANIELE Institut de recherches sur la catalyse et l'environnement de Lyon IRCEL YON-UMR 5256 Équipe CDFA 2 Avenue Albert EINSTEIN 69 626 Villeurbanne CEDEX directeur@edchimie-lyon.fr
E.E.A.	<u>ÉLECTRONIQUE,</u> <u>ÉLECTROTECHNIQUE,</u> <u>AUTOMATIQUE</u> http://edeaa.ec-lyon.fr Sec. : M.C. HAVGOUDOUKIAN ecole-doctorale.eea@ec-lyon.fr	M. Gérard SCORLETTI École Centrale de Lyon 36 Avenue Guy DE COLLONGUE 69 134 Écully Tél : 04.72.18.60.97 Fax 04.78.43.37.17 gerard.scorletti@ec-lyon.fr
E2M2	<u>ÉVOLUTION, ÉCOSYSTÈME,</u> <u>MICROBIOLOGIE, MODÉLISATION</u> http://e2m2.universite-lyon.fr Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 INSA : H. CHARLES secretariat.e2m2@univ-lyon1.fr	M. Philippe NORMAND UMR 5557 Lab. d'Ecologie Microbienne Université Claude Bernard Lyon 1 Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69 622 Villeurbanne CEDEX philippe.normand@univ-lyon1.fr
EDISS	<u>INTERDISCIPLINAIRE</u> <u>SCIENCES-SANTÉ</u> http://www.ediss-lyon.fr Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 INSA : M. LAGARDE secretariat.ediss@univ-lyon1.fr	Mme Sylvie RICARD-BLUM Institut de Chimie et Biochimie Moléculaires et Supramoléculaires (ICBMS) - UMR 5246 CNRS - Université Lyon 1 Bâtiment Curien - 3ème étage Nord 43 Boulevard du 11 novembre 1918 69622 Villeurbanne Cedex Tel : +33(0)4 72 44 82 32 sylvie.ricard-blum@univ-lyon1.fr
INFOMATHS	<u>INFORMATIQUE ET</u> <u>MATHÉMATIQUES</u> http://edinfomaths.universite-lyon.fr Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 infomaths@univ-lyon1.fr	M. Hamamache KHEDDOUCI Bât. Nautibus 43, Boulevard du 11 novembre 1918 69 622 Villeurbanne Cedex France Tel : 04.72.44.83.69 hamamache.kheddouci@univ-lyon1.fr
Matériaux	<u>MATÉRIAUX DE LYON</u> http://ed34.universite-lyon.fr Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bât. Direction ed.materiaux@insa-lyon.fr	M. Jean-Yves BUFFIÈRE INSA de Lyon MATEIS - Bât. Saint-Exupéry 7 Avenue Jean CAPELLE 69 621 Villeurbanne CEDEX Tél : 04.72.43.71.70 Fax : 04.72.43.85.28 jean-yves.buffiere@insa-lyon.fr
MEGA	<u>MÉCANIQUE, ÉNERGÉTIQUE,</u> <u>GÉNIE CIVIL, ACOUSTIQUE</u> http://edmega.universite-lyon.fr Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bât. Direction mega@insa-lyon.fr	M. Jocelyn BONJOUR INSA de Lyon Laboratoire CETHIL Bâtiment Sadi-Carnot 9, rue de la Physique 69 621 Villeurbanne CEDEX jocelyn.bonjour@insa-lyon.fr
ScSo	<u>ScSo*</u> http://ed483.univ-lyon2.fr Sec. : Véronique GUICHARD INSA : J.Y. TOUSSAINT Tél : 04.78.69.72.76 veronique.cervantes@univ-lyon2.fr	M. Christian MONTES Université Lyon 2 86 Rue Pasteur 69 365 Lyon CEDEX 07 christian.montes@univ-lyon2.fr

Acknowledgments

Español

Desde que era un niño siempre soñé con la oportunidad de volverme un científico. Gracias a mis padres aquella oportunidad se hizo realidad. Agradezco su apoyo incondicional, su amor y sus consejos a través estos años, sin ellos esto no habría sido posible.

También agradezco a mi director de Tesis, Dr. Arnaud Leleve, en primera estancia por haberme aceptado, y también por sus consejos y ayuda para que este proyecto saliera adelante. Agradezco también a mis otros dos directores, Damian Eberard y Minh Tu Pham por su ayuda durante mi estancia, especialmente durante los últimos meses para asegurar la entrega de un gran trabajo.

Agradezco también a la gente del laboratorio, en especial a Paolo Massioni, Sylvie Sesmat, Federico Bribiesca Argomedo y Vicent Lechappe, por su amistad durante estos tres años.

Agradezco a CONACyT y al pueblo mexicano por haberme otorgado mi beca y permitirme convertirme en un doctor. Espero que el fruto de mis estudios pueda ayudar a la superación de mi país y ayude a la humanidad.

Finalmente quisiera agradecer a mi esposa, Ma de los Angeles Alamilla Daniel, quien estuvo conmigo en los momentos más claros y oscuros de estos últimos tres años, quien limpiaba mis lágrimas y celebraba mis triunfos. Sin sus palabras de apoyo y consuelo, quizás nunca hubiera sido capaz de ser lo que hoy soy. Cumplimos un sueño más, pero aún nos faltan mucho por recorrer, aún estamos empezando.

English

Ever since I was a child, I always dreamed of the opportunity to become a scientist. Thanks to my parents that opportunity came true. I appreciate their unconditional support, their love, and their advice through these years, without them this would not have been possible.

I also thank my thesis director, Dr. Arnaud Leleve, for have accepted me, and also for his advice and help to get this project done. I also thank my other two directors, Damian Eberard and Minh Tu Pham for their help during my stay, especially during the last months to ensure the quality of this thesis.

I want to thank the people at the laboratory Ampère, especially to Paolo Massioni, Sylvie Sesmat, Federico Bribiesca Argomedo, and Vicent Lechappe, for their friendship during these three years.

I want to thank CONACyT and my compatriots for granted me a scholarship which allowed me to become a doctor. I hope that the fruit of my studies can help to develop my country and help humanity.

Finally, I would like to thank my wife, Ma de los Angeles Alamilla Daniel, who stayed with me in the clearest and darkest moments of these last three years. Who wiped my tears and celebrated my triumphs. Without your words of support and comfort, perhaps I would never have been able to be what I am now. We fulfilled one more dream, but we still have a long way to go, we are still starting.

Contents

I	Résumé en Français	1
1	Introduction	3
1.1	Définition du problème	3
1.1.1	Le problème de la formation pratique non supervisée	6
1.1.2	Cas d'utilisation principal et exigences	7
1.2	Objectives of the thesis	10
1.3	Contributions	11
1.4	Document Outline	12
2	État de l'art	15
2.1	Introduction	15
2.2	Les interfaces haptiques	16
2.2.1	Classification des interfaces haptiques	16
2.2.2	Avantages et inconvénients	19
2.3	Simulateurs pour l'apprentissage pratique dans le domaine médical	19
2.4	Les architectures de contrôle des simulateurs	22
2.4.1	Configurations multilatérales pour les architectures de contrôle	22
2.4.2	Architecture Dual user	24
2.4.3	Partage d'autorité dans un contexte de formation pratique	24
2.4.4	Autorité adaptative	26
2.4.5	Étude bibliographique des architectures de contrôle dans les simulateurs haptiques	27
2.5	Conclusion de cette étude	33

3	Résumé des contributions	35
II	Background	39
1	Introduction	41
1.1	Problem definition	41
1.1.1	The problem of non-guided training	44
1.1.2	Main use case and main requirements	45
1.2	Objectives of the thesis	47
1.3	Contributions	49
1.4	Document Outline	49
2	State of art	51
2.1	Introduction	51
2.2	Haptic devices	52
2.2.1	Haptics device classification	53
2.2.2	Advantages and disadvantages	54
2.3	Medical haptic training simulators	55
2.4	Simulator control architectures	57
2.4.1	Multilateral configurations for control architectures	57
2.4.2	Dual user scheme	60
2.4.3	Dominance sharing in a training context	60
2.4.4	Adaptive dominance	61
2.4.5	Bibliographical study of control architectures for haptic simulators	62
2.5	Conclusion	68
III	Contributions	69
3	Expanding to n-DoF (eESC architecture)	71
3.1	1-DoF ESC architecture	71
3.2	The Intrinsically Passive Control	73
3.2.1	Impedance-Admittance IPC	75

3.2.2	Transparency and stability of the IPC	76
3.3	From 1-DoF to n-DoF	84
3.3.1	Problem	84
3.3.2	Assumptions and proposal	84
3.3.3	Simulation	86
3.4	Test bench for experimental results	91
3.4.1	Core Setup	91
3.4.2	User interface	93
3.5	Experimental validation for Articular approach	95
3.5.1	Determining control parameters	95
3.5.2	Types of experiments	97
3.5.3	Articular expansion	98
3.5.4	Cartesian expansion	103
3.6	Discussion	110
4	Enhanced AAA	113
4.1	AAA for ESC	113
4.1.1	Adaptive Virtual Boundary	113
4.1.2	Virtual Boundary Torque	114
4.1.3	Adaptive Dominance Factor	114
4.1.4	Issue with current AAA	115
4.2	Proposed eAAA	115
4.2.1	Line profile	116
4.2.2	Concave curve profile	117
4.2.3	Convex curve profile	117
4.2.4	Bézier curve profile	118
4.3	Experimental validation	119
4.3.1	Correctional force validation	119
4.3.2	Free motion	120
4.4	Adaptive dominance validation	123
4.4.1	Exercise selection	123
4.4.2	Validation	126

4.5	Results and discussion	130
5	Expanding eESC to $m - 1$ trainees	131
5.1	Problem	131
5.2	Expansion	131
5.3	Experimental results	135
5.3.1	New setup	135
5.3.2	Free motion	135
5.3.3	Force tracking	137
5.4	Conclusion	141
	Acknowledgments	144
IV	Appendix	147
A	Mathematical models	149
A.1	Geomagic Touch 3D	149
A.1.1	Forward kinematics	149
A.1.2	Inverse kinematics	150
A.1.3	Jacobian matrix	150
A.1.4	Lagrangian formulation	152
A.2	Root Mean Square	153
B	Hybrid matrix of Impedance-Admittance IPC	155
C	Time Domain Passivity Control (TDPC)	159
C.1	Passivity	159
C.1.1	Passivity for an M-port network	160
C.1.2	Passivity observer	161
D	Survey AAA validation	163

List of Figures

1-1	Simulateurs type mannequins	5
1-2	Simulateur haptique pour la laparoscopie : <i>Lapsim</i> (SurgicalScience)	6
1-3	Simulateur haptique en réalité augmentée : Immersive touch (University of Illinois, USA)	7
1-4	Simulateurs mono et multi apprenants	8
1-5	Cas d'utilisation principal du simulateur idéal de formation pratique au geste	9
2-1	Simulateur pour apprendre les techniques d'arthroplastie	16
2-2	Processus d'interactions entre l'utilisateur et un monde virtuel dans un simulateur	17
2-3	Manette haptique pour jeux vidéo	17
2-4	Structures sérielles et parallèles pour des interfaces haptiques	18
2-5	Exemples d'interfaces haptiques pour chaque catégorie de rendu haptique	18
2-6	Suturehap: stitching simulator	20
2-7	Simulateur de palpation	20
2-8	hapTEL: simulateur de procédure dentaire	21
2-9	NeuroTouch: simulateur pour l'endoscopie	21
2-10	<i>Lapsim</i> ®: simulateur en laparoscopie	22
2-11	Bilateral teleoperation scheme	23
2-12	Configuration SM/SS	23
2-13	Configuration SM/MS	23
2-14	Configuration MM/SS	24
2-15	Configuration MM/MS	24
2-16	Architecture Dual-user	25
2-17	Les situations de partage d'autorité dans un contexte de formation au geste	25

1-1	Mannequin based simulators	42
1-2	Commercial simulators	44
1-3	Trainee teaching concepts	46
1-4	Guided training simulator: Case Diagram	47
2-1	FundamentalVR immersive simulator for Knee Arthroplasty	52
2-2	Haptic environment diagram	52
2-3	Videogame controller with force feedback motors	53
2-4	Serial and parallel mechanism haptic devices	54
2-5	Haptic Devices Classification	54
2-6	Suturehap: stitching simulator	55
2-7	Palpation simulator	56
2-8	hapTEL: Dental procedure simulator	56
2-9	NeuroTouch simulator: Endoscopy simulator	57
2-10	<i>Lapsim</i> ®: Laparoscopy simulator	57
2-11	Haptic simulation architecture	58
2-12	SM/SS configuration	58
2-13	SM/MS configuration	59
2-14	MM/SS configuration	59
2-15	MM/MS configuration	59
2-16	Dual-user system	60
2-17	Dominance sharing training cases	61
3-1	ESC architecture dual-user haptic system for one dof	71
3-2	Impedance IPC controller for the master	74
3-3	Admittance IPC controller for the slave	74
3-4	Impedance-Admittance IPC for ESC	75
3-5	Teleoperator scheme	77
3-6	ESC's IPC transparency diagram	79
3-7	Bode diagram for the ESC hybrid matrix with different dampers	81
3-8	Comparison between differents architectures	83
3-9	Multiple DoF Articular eESC architecture	85

3-10 Multiple DoF Cartesian eESC architecture	85
3-11 Simulation diagram: Cartesian vs Articular approaches	87
3-12 Expansion n-DoF: simulated Articular approach trajectory	88
3-13 Expansion n-DoF: simulated Cartesian approach trajectory	88
3-14 Expansion to n-DoF (Cartesian approach): simulated Articular position	89
3-15 Expansion to n-DoF (Cartesian approach): simulated position error	90
3-16 Expansion to n-DoF (Articular approach): simulated Cartesian position	90
3-17 Geomagic 3D Touch TM	91
3-18 3D printed environment	92
3-19 Setup organization	92
3-20 Framework connection	93
3-21 Framework layer hierarchy with the new user interface	94
3-22 Chai3D user interface	95
3-23 Determining R_f in X axis with a Geomagic 3D Touch	96
3-24 Expansion n-DoF: trajectory in free motion	99
3-25 Expansion to n-DoF: Articular positions in free motion	100
3-26 Expansion to n-DoF: torques and α in free motion	100
3-27 Expansion to n-DoF: Articular error in free motion	101
3-28 Expansion to n-DoF: Energy E_p in free motion	101
3-29 Expansion to n-DoF: Trajectory in wall contact	102
3-30 Expansion to n-DoF: Position and force along Cartesian x in wall contact	102
3-31 Expansion to n-DoF: Energy E_p in wall contact	103
3-32 Haption Virtuose 6-Desktop	104
3-33 Framework connection for Haption Virtuose slave	104
3-34 Expansion to n-DoF (Cartesian approach): Trajectory in free motion	105
3-35 Expansion to n-DoF (Cartesian approach): Positions	106
3-36 Expansion to n-DoF (Cartesian approach): Forces	106
3-37 Expansion to n-DoF (Cartesian approach): Positions error	107
3-38 Expansion to n-DoF (Cartesian approach): Positions in wall contact	108
3-39 Expansion to n-DoF (Cartesian approach): forces in wall contact	108
3-40 Expansion to n-DoF (Cartesian approach): position errors in wall contact	109

3-41	Expansion to n-DoF (Cartesian approach): force errors in wall contact	110
4-1	eAAA default profiles	116
4-2	Validation trajectory	120
4-3	Forces and dominance change for various profiles	121
4-4	AAA position tracking test: positions	122
4-5	AAA position tracking test: forces	122
4-6	AAA validation setup	127
4-7	Environment interaction object	127
5-1	ESC expanded to m devices for only one dof (Cartesian example)	132
5-2	Dominance hierarchy	134
5-3	Expansion m-trainee: Trajectory in free motion	136
5-4	Expansion m-trainee: Position with $\alpha = 0$ in free motion	137
5-5	Expansion m-trainee: Position errors with $\alpha = 0$ in free motion	138
5-6	Expansion m-trainee: Position and force in axis x in force tracking	139
5-7	Expansion m-trainee: Position and force in axis y in force tracking	139
5-8	Expansion m-trainee: Position and force in axis z in force tracking	140
A-1	Geomagic Touch 3D kinematic analysis	149
C-1	Input and Output from a H system	160
C-2	M-port network	161

List of Tables

2.1	MM/SS architectures comparison table	30
2.1	MM/SS architectures comparison table	65
3.1	Analysis of the hybrid matrix in Low and High Frequency	80
3.2	IPCs' parameters for simulation validation: Cartesian vs Articular	87
3.3	Geomagic 3D Touch specifications	91
3.4	IPCs' parameters for Articular approach	93
3.5	Haptic Device's specifications	104
3.6	IPCs' parameters for Cartesian approach	105
4.1	General Constraint and General Feasibility criteria for proposed exercises	124
4.2	Motion and Interaction criteria for proposed exercises	125
4.3	Final score for proposed exercises	125
4.4	General Constraint and General Feasibility criteria for proposed exercises: Second wave	125
4.5	Motion and Interaction criteria for proposed exercises: Second Wave	126
4.6	Final score for proposed exercises: Second Wave	126
4.7	Push and Pull results	129
5.1	RMS error in free motion	137
D.1	Participants' answers to the survey	165

Part I

Résumé en Français

Chapter 1

Introduction

1.1 Définition du problème

La formation des étudiants en médecine est un long processus qui nécessite de nombreuses heures de pratique et d'étude afin d'assurer l'acquisition d'un savoir et d'une dextérité indispensables. Pour maîtriser la palpation, l'injection de produits, la chirurgie et tout autre processus qui implique une interaction directe avec le patient, les étudiants ont besoin de pratiquer de nombreuses fois. Cependant, il n'est pas recommandé de le faire sur des patients pour des raisons éthiques car ceux-ci pourraient être blessés ou simplement en souffrir (Beal et al. [2017]).

Pour cette raison, il est encore aujourd'hui crucial que ces étudiants améliorent leur compétences manuelles sans toucher un patient. Les étudiants en médecine utilisent traditionnellement des simulateurs d'entraînement qui les aident à acquérir de l'expérience en leur fournissant des sensations pseudo-réalistes simulant le comportement et les réactions mécaniques du corps humain. Certains simulateurs, utilisés depuis des siècles, sont réalisés en bois, cuir, métal et même à base d'organes voire même de cadavres humains. Par exemple, en obstétrique, le "Phantom" a été développé au 18^e siècle en France par les obstétriciens Grégoire et fils. Ce premier simulateur était constitué d'un vrai pelvis humain et d'un cadavre de bébé afin de reproduire le processus d'accouchement (Owen [2012]). De nos jours, les étudiants en médecine ont la possibilité de s'entraîner sur plusieurs types de chirurgie sur des corps morts ou presque. Cependant, l'usage de cadavres entraîne de nombreuses difficultés (Berger et al. [2002]):

- les cadavres n'ont plus de réaction biologique, ce qui empêche notamment de déterminer s'ils sont blessés par des gestes inappropriés ;

- il est nécessaire d'obtenir le consentement d'une personne ou de sa famille pour pouvoir utiliser son corps post-mortem ;
- un équipement spécifique et onéreux est indispensable pour empêcher sa décomposition naturelle
- les cadavres sont à usage limité car les tissus sont altérés définitivement et ne peuvent pas être restaurés.

Depuis les années 60, les simulateurs « mannequins » sont progressivement apparus. Par exemple, Resusci-Anne est une mannequin de formation pratique pour la réanimation cardiopulmonaire. Mais celui-ci ne s'est pas diffusé du fait de son coût élevé (Fritz et al. [2008]). Des recherches ont été réalisées sur la restitution des signes vitaux humains pendant les sessions de formation. L'Université de Miami a créé le célèbre mannequin Harvey, utilisé pour former à l'examen en cardiologie. Il fut le premier être couramment utilisé dans l'ère des simulateurs d'apprentissage de type mannequin (Kunkler [2008]).

L'usage de mannequins (Figure 1-1) repose sur une représentation du comportement des patients plus réaliste que celui observable avec les cadavres. En effet, il est alors possible de générer arbitrairement des événements critiques pendant une opération. Un exemple est le mannequin CASE (Cooper and Taqueti [2008]) développé en 1987. Il a été utilisé pour l'entraînement à l'anesthésie et simulait des signes vitaux du patient grâce à des instruments virtuels sur ordinateur.

Le mannequin était manipulé par les formateurs¹ pour créer des situations dans lesquelles la vie du patient était en danger afin que les étudiants apprennent à prendre les bonnes décisions.

Plus récemment, les étudiants en médecine ont eu accès à des simulateurs mannequins plus avancés et plus réalistes leur permettant de s'entraîner sur des techniques plus complexes comme en endoscopie (Deutsch [2008]). Ces simulateurs utilisent désormais une combinaison de systèmes mécaniques, électroniques et logiciels afin de synthétiser une réponse réaliste du corps d'un patient pendant une opération. Il est désormais possible de créer des scénarios rares (très difficile avec des cadavres) et plus besoin de s'approvisionner en organes ou cadavres (Fritz et al. [2008]). Cependant, leur morphologie figée limite la variété des cas d'étude pour la formation. Par exemple, ils ne peuvent pas servir pour s'entraîner à la chirurgie car les apprenants ne peuvent pas modifier/couper/suturer leurs organes.

¹Afin d'alléger l'écriture pour mieux me centrer sur le fond, je n'utiliserai pas d'écriture inclusive dans le texte français de ce document. Il va de soi que chaque acteur/trice dans le texte peut aussi bien être du genre féminin comme masculin.



Figure 1-1 – Simulateurs type mannequins

Grâce aux avancées technologiques, un nouveau type de simulateur d'apprentissage résout certains des problèmes mentionnés. Les premiers simulateurs informatiques proposaient des environnements virtuels dans lesquels les apprenants pouvaient interagir et modifier les organes des patients. Pour cela, des modèles biomécaniques du corps humain étaient utilisés pour simuler une réaction réaliste. Cependant, ces simulateurs n'offrent pas de retour d'effort, ce qui limite le réalisme de l'opération (Neumann et al. [1998] et Tsai et al. [2001]). Des simulateurs plus récents implémentent des interfaces haptiques (voir détails en section 2.2)) qui permettent aux utilisateurs de manipuler un outil virtuel pour opérer un patient tout aussi virtuel. Simultanément, ils peuvent sentir les efforts d'interaction entre leur outil et les organes du patient, ce qui produit un réalisme proche d'une opération réelle.

Un exemple de simulateur haptique en formation médicale est le *Lapsim*®(voir Figure 1-2) développé par SurgicalScience. Ce simulateur est utilisé pour former les étudiants en médecine à la chirurgie laparoscopique. Les utilisateurs interagissent avec un monde virtuel affiché à l'écran en manipulant deux interfaces haptiques reproduisant un retour d'effort

Un autre exemple est le Immersive Touch (Figure 1-3), un simulateur développé par l'Université de l'Illinois, qui utilise un miroir transparent pour projeter des figures en 3D dans l'espace et fournir un retour haptique. L'utilisateur doit porter des lunettes spécifiques afin de transmettre au simulateur la position et l'orientation de sa tête afin d'ajuster la perspective stéréoscopique de l'image. Cette méthode de rendu permet à l'utilisateur de bouger ses mains dans un environnement virtuel, ce qui est indispensable pour situer des repères importants (Cristian et al. [2005]).

Les simulateurs haptiques ont des avantages sur les simulateurs informatiques classiques, comme



Figure 1-2 – Simulateur haptique pour la laparoscopie :*Lapsim*(SurgicalScience)

l'usage d'environnements virtuels dans lesquels les étudiants peuvent s'entraîner sur différentes morphologies de patients et sur des cas variés. D'autre part, ils peuvent répéter plusieurs fois cet apprentissage sans toucher un patient et sans risque de pénurie de pièce d'usure (Sutherland et al. [2013]). Pour cette raison, l'usage des simulateurs haptiques est devenu le nouveau paradigme dans le processus de formation médicale (Jones et al. [2015]). Cependant, un inconvénient significatif est le coût d'usage de ce type de simulateur qui peut monter jusqu'à \$1 million de dollars (Zevin et al. [2014]).

1.1.1 Le problème de la formation pratique non supervisée

Bien que les simulateurs permettent aux étudiants de s'autoformer, ils ne sont pas la panacée. En effet, il est encore aujourd'hui nécessaire d'être guidé par un expert dont les trajectoires d'outils et les commentaires sont bien plus formateurs que de longues répétitions sans guide intelligent (Schout et al. [2010]). Cette façon de former, dans laquelle le formateur tient les mains de l'apprenant qui opère peut s'appeler de la **formation pratique guidée ou supervisée**. Hors simulateur c'est possible et même traditionnel. Sur un simulateur, c'est aussi envisageable bien que plus rare. L'inconvénient majeur de cette forme d'apprentissage est que l'apprenant ne peut pas doser ses gestes car le retour d'effort de l'outil est partagé entre les deux protagonistes. Pourtant, cet apprentissage des efforts à fournir est indispensable dans la formation des médecins et ne se fait que « sur le tas » aujourd'hui.

Il est donc indispensable de proposer des simulateurs haptiques offrant la possibilité de transmettre du formateur à l'apprenant ces gestes dans toute leurs dimensions : cinématique et dynamique. Une étude bibliographique récente a montré que la vaste majorité des simulateurs disponibles sur étagère ne

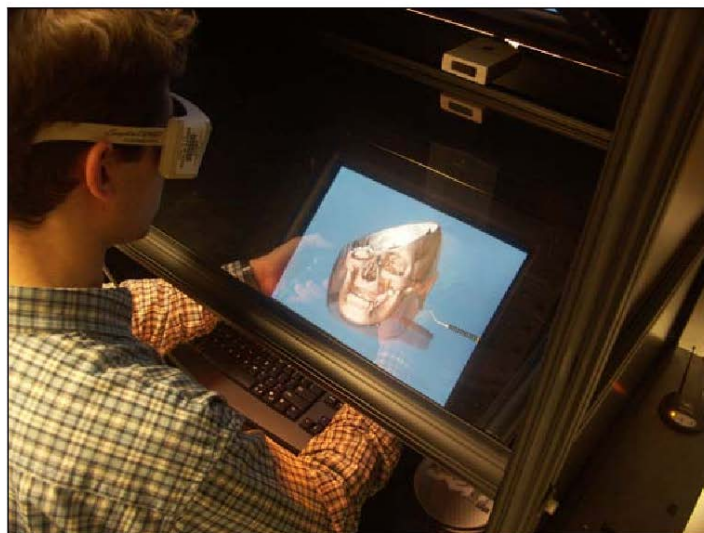


Figure 1-3 – Simulateur haptique en réalité augmentée : Immersive touch (University of Illinois, USA)

proposent pas de deuxième interface pour le formateur afin de l'intégrer dans la simulation. Certains modèles sont équipés de deux interfaces haptiques mais pour un usage à deux mains par le même utilisateur (Escobar-Castillejos et al. [2016]).

Certaines recherches ont cherché à combler ce manque de support pour un apprentissage supervisé sur simulateur haptique. Les résultats de ces recherches sont décrits en section 2.4.5 mais cependant ils ne répondent pas à l'ensemble des exigences qu'un tel simulateur doit remplir ; exigences précisées dans la section suivante.

1.1.2 Cas d'utilisation principal et exigences

Le cas d'utilisation suivant aide à déterminer les principales exigences du système étudié. Supposons, en premier lieu, que le formateur (un chirurgien expérimenté) cherche à montrer les trajectoires à

réaliser avec son outil chirurgical afin de réaliser une tâche impliquant successivement des mouvements libres et des contacts entre son outil et son patient. Cela implique un retour haptique réaliste pour qu'il puisse doser ses efforts, comme dans un contexte de téléopération bilatérale. Le formateur prend la main sur le simulateur (il devient *leader* et l'apprenant *follower*) afin de réaliser une **démonstration**. Il a alors un retour haptique complet provenant de l'esclave, comme s'il manipulait directement ses outils. Pendant cette démonstration, l'interface haptique de l'apprenant suit la trajectoire de celle du *leader*. S'il vient à en dévier, il ressent immédiatement un effort compliant qui le ramène sur la trajectoire imposée. Pendant les phases d'interaction entre l'outil et son environnement, une interface graphique affichée sur un moniteur guide l'apprenant pour qu'il place son interface dans la même position que celle du leader afin de ressentir les mêmes sensations haptiques. Ensuite, le fonctionnement peut être inversé afin que l'apprenant guide l'esclave et que le formateur l'évalue: il s'agit alors du mode **évaluation**.

D'autre part, le simulateur ne doit pas se limiter à un usage à deux utilisateurs. Un système étendu à m apprenants l'utilisant simultanément présente un intérêt certain. Ainsi le formateur peut réaliser une seule fois la même démonstration à m apprenants (Figure 1-4b), plutôt que de la répéter m fois (Figure 1-4a). Aussi, en mode évaluation, les étudiants qui ne sont pas évalués peuvent aussi suivre les performances de l'impétrant(e), ce qui est aussi intéressant du point de vue pédagogique. Ce concept offre un temps d'enseignement plus efficace et optimise l'usage de tels simulateurs.

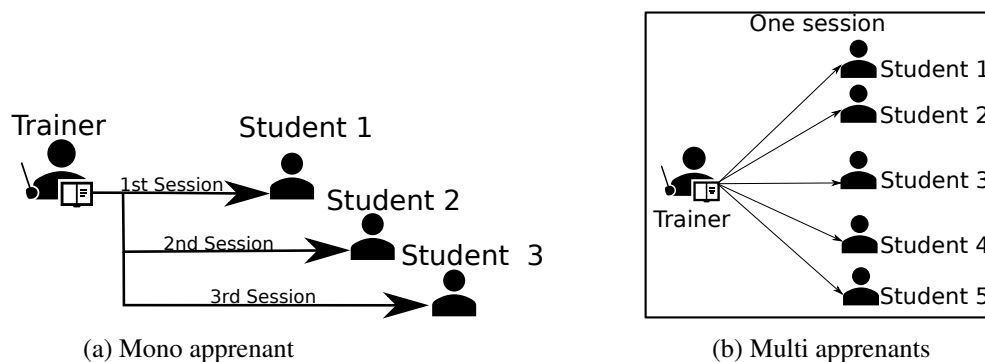


Figure 1-4 – Simulateurs mono et multi apprenants

La Figure 1-5 synthétise ce scénario principal à l'aide d'un diagramme de cas d'utilisation suivant le standard SysML. Ce scénario implique les exigences suivantes pour un tel simulateur offrant la possibilité de former les apprenants sur la reproduction de trajectoires mais également d'interactions inhérentes entre l'outil et son environnement.

Exigences :

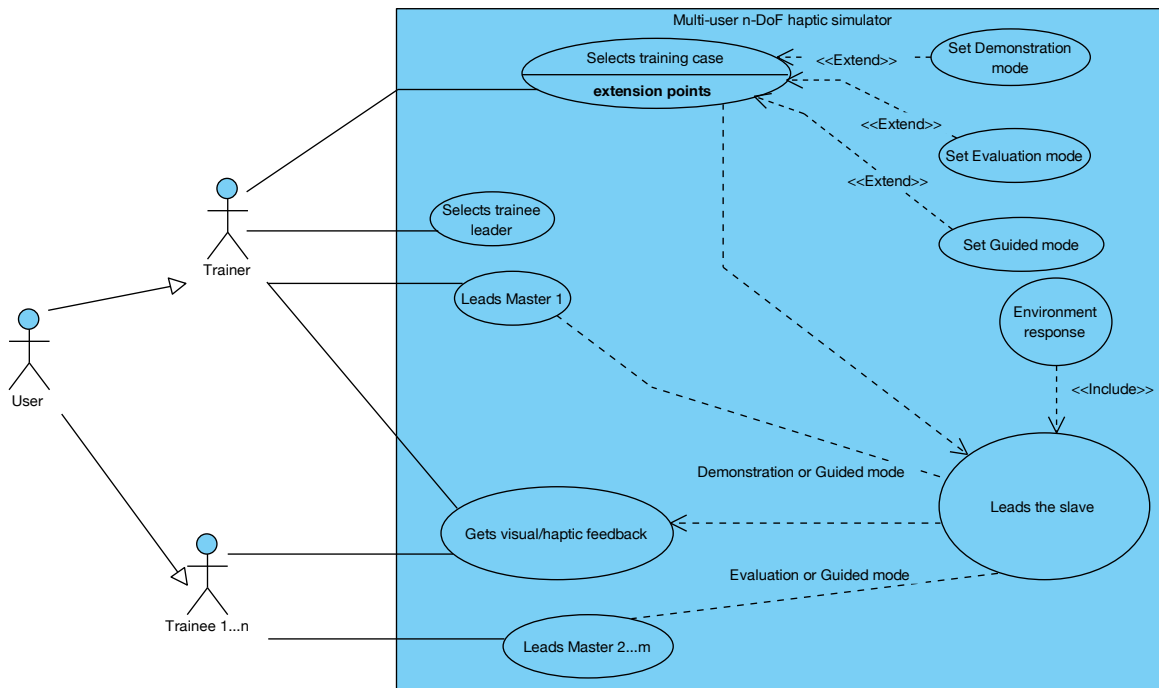


Figure 1-5 – Cas d'utilisation principal du simulateur idéal de formation pratique au geste

Requirements

- (R1) Multi-apprenant :** le simulateur doit être capable d'accueillir plusieurs utilisateurs simultanément incluant un formateur et un à plusieurs apprenants.
- (R2) Sélection du mode de formation :** le formateur peut changer de mode démonstration/évaluation.
- (R3) Dominance adaptative :** le simulateur doit pouvoir adapter le niveau de contrôle (dominance) de l'apprenant sur la simulation, selon les performances de celui-ci.
- (R4) Sélection de l'apprenant leader :** le formateur peut sélectionner quel apprenant est leader.
- (R5) Suivi en position :** le simulateur contrôle les interfaces haptiques des utilisateurs et l'esclave (réel ou virtuel) afin que les positions des followers et de l'esclave suivent celle du leader.
- (R6) Suivi en effort :** quand l'esclave est en interaction avec son environnement, le leader doit ressentir ces efforts d'interaction, ainsi que les followers à partir du moment où ils

ont positionné leur interface comme celle du leader.

(R7) Compatibilité entre interfaces maîtres et robot esclave : le simulateur doit autoriser l'usage d'interfaces et d'un robot ayant des cinématiques différentes.

(R8) Stabilité : le simulateur doit être stable quelques que soient les mouvements et interactions outil-environnement.

De ces exigences, nous déduisons d'emblée qu'il faut une architecture de contrôle connectant plusieurs interfaces haptiques avec un robot esclave (réel ou virtuel).

1.2 Objectives of the thesis

Les objectifs du Groupe de Travail Robotique du laboratoire Ampère sont de développer un simulateur de formation pratique à deux utilisateurs (dual-user) capable de former les apprenants sur des trajectoires et des interactions outil-environnement. Dans ce projet, un premier travail a été réalisé par M. Fei Liu entre 2013 et 2016 dans le cadre de son doctorat, qui a mené à la conception de l'architecture Energy Shared Control (ESC).

ESC utilise des contrôleurs de type Intrinsically Passive Controllers (IPC) (détaillés en section 3.2) pour le contrôle compliant des interfaces et du robot. Elle est composée d'interconnexions passives afin d'assurer la passivité de l'ensemble et donc sa stabilité (des détails sur cet aspect sont disponibles en annexe C). Des expérimentations ont montré une bonne transparence en terme de suivi en position et en efforts lors des interactions, sur une architecture à un seul degré de liberté.

Les principaux avantages d'ESC et les exigences auxquelles il répond sont :

Avantages

- ESC intègre le réglage dynamique de la dominance entre les utilisateurs permettant de travailler dans les modes démonstration/évaluation ainsi que de la faire évoluer de manière adaptative (R2) ;
- ESC propose une fonction de dominance adaptative (Adaptive Authority Adjusting AAA) ajustant la dominance entre l'apprenant et le formateur en fonction des performances de l'apprenant (R3) ;
- Il transmet les efforts mesurés entre l'outil et son environnement aux utilisateurs (leader

ainsi que follower(s) pour peu qu'il(s) positionne(nt) adéquatement son/leur interface haptique) (R5) ;

- L'effort reproduit par les interfaces haptiques des utilisateurs est simplement et précisément celui d'interaction entre l'outil et son environnement (R6) ;
- Intégrant un contrôleur de passivité (Time Domain Passivity Control (TDPC), dont les détails sont fournis en annexe C), sa stabilité est garantie (R8).

Les principaux inconvénients d'ESC sont:

Disadvantages

- ESC n'a été validé expérimentalement que pour un seul degré de liberté ;
- La fonction AAA dépend de quatre variables difficiles à paramétrer par le formateur en fonction du niveau d'expertise de l'apprenant et de la précision désirée de la tâche.

En tenant compte de ces inconvénients, les objectifs de mon doctorat sont :

Objectives

- (O1) Étendre ESC à n degrés de liberté (ddl) (R7) ;
- (O2) Étendre le système à plusieurs apprenants sans remettre en cause l'architecture actuelle (R1), et fournir un mécanisme de sélection par le formateur de l'apprenant leader (R4) ;
- (O3) Améliorer la fonction AAA afin qu'elle soit plus facile d'utilisation (R3) ;
- (O4) S'assurer que la version étendue d'ESC conserve ses performances en terme de suivi de trajectoire et d'efforts (R5 et R6) ;
- (O5) Permettre l'usage d'interfaces et robots ayant des cinématiques différentes (R7) ;
- (O6) Garantir que la version étendue d'ESC préserve ses propriétés de stabilité (R8).

1.3 Contributions

Pour atteindre ces objectifs, nous avons réalisé un état de l'art (présenté en section 2.4.5), qui nous a permis de déterminer les avantages et inconvénients des solutions présentes dans la littérature sci-

entifique. Nous avons pu en conclure que les objectifs d'extension et amélioration d'ESC étaient cohérents vis à vis de l'existant.

Nous avons ensuite commencé par étendre ESC à n degrés de liberté, car avec un seul ddl, il n'était pas possible d'en faire un simulateur utile pour l'apprentissage du geste. Cette extension est détaillée au chapitre 3 et a mené à publication dans la revue *Robotica* en 2019 (Liu et al. [2019]). Nous avons montré que cette architecture peut fonctionner également avec des interfaces maîtres et robot esclave à cinématiques différentes. Nous avons validé expérimentalement cette proposition avec 3 degrés de liberté (position dans l'espace). This work has been submitted for presentation at ICRA 2020 conference.

La fonction Adaptive Authority Adjusting (AAA) proposée initialement par Fei Liu dans (Liu et al. [2015]) a pour objectif de modifier dynamiquement le niveau de dominance de chaque utilisateur en fonction des performances de l'apprenant. Nous avons proposé une nouvelle fonction pour cela, qui ne fait appel qu'à deux paramètres numériques et le choix d'un profil d'utilisation dépendant de la tâche à réaliser et de l'expertise de l'apprenant. Nous avons réalisé une validation expérimentale avec une dizaine d'utilisateurs afin de déterminer si cette fonction aide à l'apprentissage du geste en terme de trajectoires et d'efforts d'interaction.

Nous avons ensuite étendu l'architecture ainsi obtenue à m apprenants simultanés en maintenant les propriétés de stabilité et transparence de l'architecture ESC initiale. Ce travail a été présenté lors de la conférence IROS 2019 (Licona Rodriguez et al. [2019]).

1.4 Document Outline

Ce document est organisé comme suit :

Au chapitre 2, nous introduisons les interfaces haptiques, comment elles fonctionnent, et nous en proposons une classification. Ensuite, nous réalisons un état de l'art sur les simulateurs informatiques pour la formation pratique en médecine, faisant appel à des interfaces haptiques. Nous étudions leurs architectures et précisons pourquoi celles-ci ne remplissent pas toutes les exigences listées précédemment.

Au chapitre 3, nous rappelons l'architecture ESC développée par Fei LIU afin d'en proposer deux méthodes d'extension en terme de nombre de degrés de liberté (approche articulaire et cartésienne).

Au chapitre 4, nous rappelons le principe de fonctionnement de la fonction AAA pour adapter dynamiquement la dominance entre formateur et apprenant(s) et nous détaillons les améliorations

apportées afin qu'elle soit plus ergonomique et efficace. Nous détaillons également les retours d'usage réalisés sur cette fonction et discutons des résultats obtenus et des perspectives envisageables.

Au chapitre 5, nous détaillons l'approche utilisée pour étendre le simulateur en terme de nombre d'apprenants simultanés. Nous présentons ensuite les résultats expérimentaux validant cette approche.

Enfin, nous terminons ce mémoire avec une conclusion couvrant l'ensemble de ces travaux et nous proposons des perspectives pour la poursuite de ce projet de simulateur haptique pour la formation pratique supervisée.

Chapter 2

État de l'art

2.1 Introduction

Les humains mémorisent assez précisément la position de leurs membres : c'est ce qui s'appelle la kinesthésie (Feygin et al. [2002]). Ce mot dérive du grec *kinēin*, « mettre en mouvement » et *aisthēsis*, « perception », ce qui peut s'interpréter comme une perception du mouvement. La kinesthésie est l'agglomération de trois sous-sens : la position et l'orientation, le mouvement et l'effort (Rosker and Sarabon [2010]).

Grâce à ce sens, les humains obtiennent une connaissance cognitive du point de vue psychomoteur et apprennent à réaliser des gestes et former des postures afin de réaliser des tâches données (Xu and Ke [2014]). Un exemple de cette approche est l'apprentissage d'un instrument de musique. En effet, les musiciens apprennent à lire les partitions et comment l'assemblage de plusieurs sons réalise une mélodie et une harmonie agréables à l'oreille. Ils acquièrent la compétence nécessaire pour reproduire cela principalement par répétition du geste (Galvao and Kemp [1999]), en plus de l'acquisition des fondamentaux théoriques. Grâce à la kinesthésie, il est possible de former des apprenants, en complément des canaux visuels et auditifs plus classiques. Il a été démontré que l'apprentissage du geste est même plus efficace quand il s'agit de gestes précis et complexes (Panait et al. [2009]).

Ainsi, les simulateurs d'apprentissage qui utilisent les techniques de réalité virtuelle (ou augmentée) s'enrichissent d'interfaces haptiques afin d'aider l'apprenant à interagir avec le simulateur via un ou plusieurs joysticks, des gants haptiques ou tout autre équipement susceptible de générer un retour haptique. Ces simulateurs se retrouvent de plus en plus dans des domaines d'application comme en aéronautique, en médecine, en ingénierie, ou tout autre domaine où l'apprentissage du geste est

indispensable et potentiellement dangereux tant qu'il n'est pas maîtrisé.

2.2 Les interfaces haptiques

Une interface haptique (ou dispositif haptique) est un système visant à reproduire la sensation de toucher d'un objet virtuel ou distant. Par exemple, ces dispositifs sont utilisés pour l'immersion dans des mondes virtuels impliquant la vision (généralement stéréoscopique), l'ouïe et le toucher (Figure 2-1).



Figure 2-1 – Simulateur immersif pour apprendre les techniques d'arthroplastie du genou ¹

L'interaction entre l'interface haptique et l'utilisateur forme une boucle fermée impliquant un mouvement conjoint à un effort. Selon la causalité choisie lors de la modélisation, il est possible de considérer que la position de l'interface est imposée par l'utilisateur tandis que celle-ci impose un effort résistant en contre-partie (cas le plus fréquent). Dans ce cas, la position de l'interface est transmise au monde virtuel et selon l'interaction générée dans ce monde, un effort résistant est transmis à l'interface pour représenter une collision, le glissement sur une surface, la déformation d'un objet mou, une vibration, ... Le monde virtuel est affiché sur un (ou deux) écran(s) selon le mode de vision mono ou stéréoscopique afin que l'utilisateur obtienne un retour visuel cohérent avec le retour haptique transmis par l'interface adéquate. Ce mécanisme est illustré en figure 2-2.

2.2.1 Classification des interfaces haptiques

Les interfaces haptiques sont utilisées pour des tâches industrielles, en recherche, et même dans la vie courante. La multiplicité des domaines applicatifs rend difficile leur classification. Ainsi, il est

¹The Verge. Consulté April 21, 2020 URL: <https://www.theverge.com/2018/8/14/17670304/virtual-reality-surgery-training-haptic-feedback-fundamentalvr>

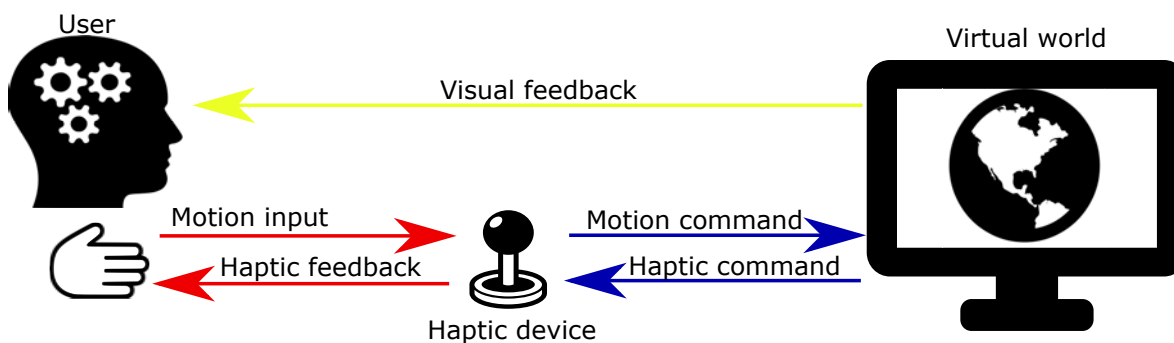


Figure 2-2 – Processus d’interactions entre l’utilisateur et un monde virtuel dans un simulateur

possible de les classer selon leur usage, par exemple: pour de la consommation courante ou un usage industriel (Sreelakshmi and Subash [2017]).

Cette première catégorie regroupe les interfaces haptiques low-cost intégrées dans des équipements de la vie courante comme des smartphones, joysticks, et autres volants de jeux de course automobile. Ils fonctionnent grâce à des moteurs électriques à forte inertie comme en figure 2-3. Celui-ci est utilisé pour alerter le joueur ou lui donner une vague sensation de collision ou d’explosion en complément de l’image à l’écran.



Figure 2-3 – Manette haptique pour jeux vidéo

La seconde catégorie est dédiée à un usage industriel et couvre un large spectre de dispositifs disponibles sur étagère. Ceux-ci sont utilisés pour réaliser des tâches complexes dans des applications scientifiques, de la sculpture virtuelle ou dans des simulateurs de formation pratique, entre autres exemples. La plupart de ces dispositifs présentent une chaîne cinématique sérielle (voir figure 2-4a) voire redondante (voir figure 2-4b). Les exemples cités ici fournissent de 3 à 6 degrés de liberté (d’actionnement)

Les interface industrielles peuvent également être classées en fonction de leur usage et de la sen-



(a) 3D Systems Touch™ 3D Stylus



(b) Force Dimension™ Omega.3

Figure 2-4 – Structures sérielles et parallèles pour des interfaces haptiques

sation recherchée (Culbertson et al. [2018]).

Les systèmes pouvant être pris en main : ceux-ci fournissent à l'utilisateur la sensation qu'il pousse/tire un objet et peuvent être utilisés pour reproduire la déformation au touché d'organes et des gestes de dissection, par exemple. Ils sont attachés au sol et proposent un stylet ou un outil que l'utilisateur prend en main afin de guider l'appareil et de ressentir des efforts d'interaction issus du monde virtuel (Figure 2-5a).

Les systèmes portables (wearable) : ils fournissent la sensation d'attraper des objets. Ils sont normalement attachés à l'utilisateur et fonctionnent en limitant ses mouvements grâce à des actionneurs (figure 2-5b).

les systèmes « touchables » : ces dispositifs donnent la sensation d'explorer une surface en agissant directement sur les nerfs cutanés situés sous la peau (Figure 2-5c).



(a) 3D Systems Touch X®

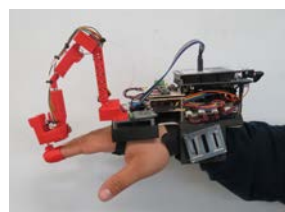
(b) Hexotrac¹(c) Touchable haptic device²

Figure 2-5 – Exemples d'interfaces haptiques pour chaque catégorie de rendu haptique

¹Sarakoglou et al. [2016]

²Perez et al. [2015]

2.2.2 Avantages et inconvénients

L'intérêt de chaque configuration dépend de la tâche envisagée. Cependant des limites communes sont à prendre en compte :

- ce type de matériel est coûteux ;
- l'effort généré est limité en amplitude et hétérogène en direction;
- limited bandwidth for the data transmission;
- la bande passante pour la génération et la transmission de signaux d'efforts est généralement trop limitée. En effet, pour un signal haptique, la bande passante doit atteindre au moins 1 Khz (Tavakoli et al. [2007]), ce qui limite des applications nécessitant des environnements virtuels complexes et donc coûteux en temps de calcul.

2.3 Simulateurs pour l'apprentissage pratique dans le domaine médical

La médecine est une science très appliquée. De nombreuses procédures font appel à la sensibilité au toucher du médecin. Les simulateurs haptiques sont développés pour aider les étudiants en médecine à s'entraîner sur des procédures en leur permettant de répéter indéfiniment leurs gestes sans risque d'usure contrairement aux moyens de simulation plus classiques tels que les cadavres et mannequins. Il existe une grande variété de simulateurs couvrant de nombreux domaines médicaux (Escobar-Castillejos et al. [2016]). Citons, à titre d'exemple :

Les simulateurs de suture : utilisés pour apprendre aux étudiants comment recoudre la peau pour bien cicatrifier les plaies. Ils utilisent typiquement deux interfaces haptiques pour guider une aiguille et une paire de ciseaux-pinces virtuels. Les interfaces permettent à l'utilisateur de positionner et orienter les outils dans l'espace. Un exemple est le Suturehap (Figure 2-6) présenté dans (Escobar-Castillejos et al. [2016]).

Les simulateurs de palpation : utilisés pour aider les étudiants en médecine à parfaire leur sens du toucher lorsqu'ils localisent des imperfections dans le corps d'un patient, comme des tumeurs ou des hernies. Comme pour les simulateurs de suture, ils ont généralement deux interfaces haptiques, une pour chaque main. À l'écran, l'utilisateur guide ses mains virtuelles et il perçoit un retour d'effort issu de l'interaction entre ses mains virtuelles et le corps tout aussi virtuel du patient. Cela leur permet d'apprendre les niveaux de pression à exercer pour effectuer un diagnostic précis sans pour autant faire

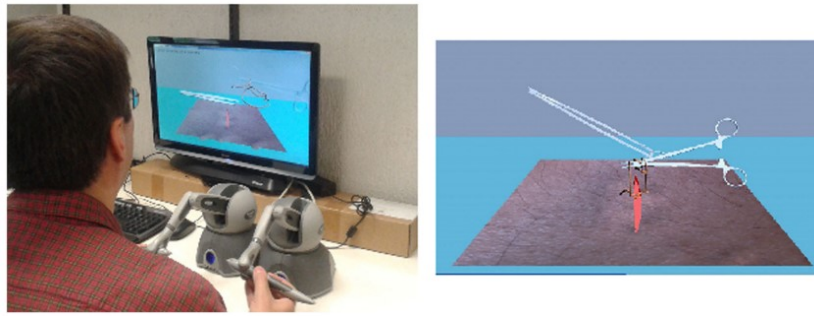


Figure 2-6 – Suturahap: stitching simulator

mal au patient. Un exemple d'un tel simulateur est celui présenté dans (Ullrich and Kuhlen [2012]) visible en figure 2-7.

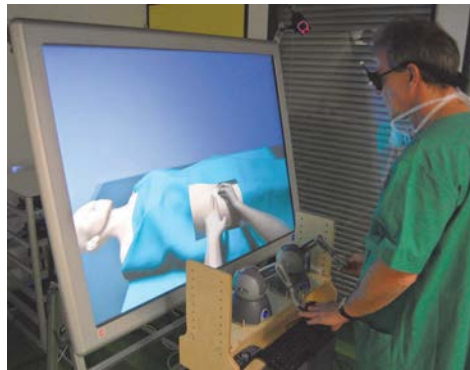


Figure 2-7 – Simulateur de palpation

Les simulateurs de procédures dentaires : ces simulateurs utilisent un modèle 3D de la dentition d'un(e) patient(e) et doivent permettre de déformer, sectionner, sentir un état de surface rugueux, ... Le dentiste peut changer d'outils et perçoit les forces d'interaction entre les dents et ceux-ci. Un exemple est celui proposé dans (Tse et al. [2010]) qui permet de s'entraîner à soigner des caries, percer et restaurer des dents (voir figure 2-8).

Les simulateurs pour l'endoscopie : ils sont utilisés pour former les étudiant(e)s à la chirurgie mini invasive, comme en neurochirurgie. Ces simulateurs apprennent aux étudiant(e)s à manipuler les instruments en sécurité et quels efforts appliquer avec ces outils sur les organes des patient(e)s. Un exemple est le NeuroTouch (R) (figure 2-9) présenté dans (Delorme et al. [2012]).

Les simulateurs pour la laparoscopie : ces simulateurs doivent reproduire des efforts de découpe et des déformations d'organes. Un exemple est le simulateur commercial LapSim (R) (figure 2-10)



Figure 2-8 – hapTEL: simulateur de procédure dentaire

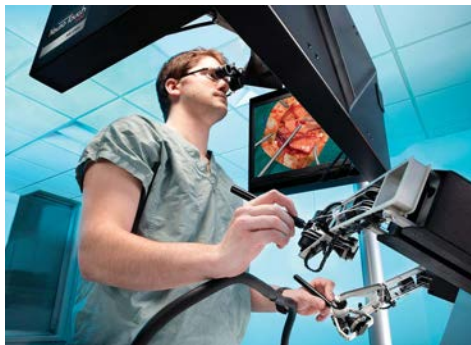


Figure 2-9 – NeuroTouch: simulateur pour l'endoscopie

décrit dans (Woodrum et al. [2006]). Il fait appel à des interfaces haptiques spécifiques reproduisant les vrais outils de laparoscopie.



Figure 2-10 – *Lapsim*®: simulateur en laparoscopie

2.4 Les architectures de contrôle des simulateurs

Bien que certains simulateurs proposent la manipulation simultanée de deux interfaces haptiques, les outils virtuels sont détachés, ce qui signifie que les mouvements de chaque interface sont indépendants tant qu'une collision virtuelle n'a pas lieu. Cette configuration ne permet pas de réaliser une formation supervisée. Pour ce faire, les deux interfaces haptiques doivent être interconnectées via un canal de communication et des lois de commande assurant le suivi en position et en effort.

2.4.1 Configurations multilatérales pour les architectures de contrôle

Pour plus de clarté concernant ces architectures de contrôle pour simulateurs haptiques, voici les principes fondamentaux de la téléopération qui les sous-tend.

La téléopération consiste à contrôler à distance depuis une interface « maître » un système (l'esclave) en échangeant des données via un médium de communication (Cui et al. [2003]). Le schéma de téléopération unilatérale, le plus simple qui existe, consiste à transmettre les ordres de l'utilisateur vers le système à contrôler, sans retour d'information. Ce manque de retour serait problématique, c'est pourquoi les architectures de téléopération sont plutôt bilatérales, ce qui autorise un suivi en position et en effort (Niemeyer and Slotine [1991]).

Une architecture de simulation haptique comporte cinq éléments : l'utilisateur (humain), le dispositif maître, le contrôleur, l'esclave et l'environnement. L'esclave peut être un vrai robot ou un logiciel simulant l'outil que l'utilisateur est sensé manipuler ; dans ce cas l'environnement est également virtuel (Secchi et al. [2007]). La figure 2-11 illustre ce schéma où \dot{x}_m et \dot{x}_s sont respectivement les vitesses du maître et de l'esclave. F_h et F_e sont les forces exercées par l'utilisateur sur le maître et l'environnement sur l'esclave. De plus, F_{c1} et F_{c2} sont les signaux de contrôle envoyés par le contrôleur

au maître et à l'esclave.



Figure 2-11 – Bilateral teleoperation scheme

Dans ce schéma, l'utilisateur impose le mouvement et reçoit un retour en effort de la part de son interface haptique maître représentant l'effort exercé par l'environnement sur l'esclave. Une architecture initialement bilatérale, présentant plus d'un maître ou esclave devient alors une architecture multilatérale. Selon le nombre de maîtres et d'esclaves, il est possible de discerner selon (Shahbazi et al. [2013]) les architectures :

Single Master with Single Slave (SM/SS): un seul utilisateur manipule un seul esclave.

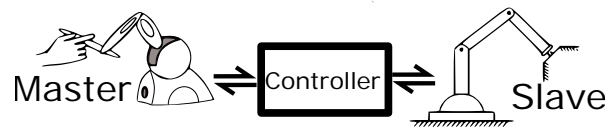


Figure 2-12 – Configuration SM/SS

Single Master with Multi Slave (SM/MS): un seul utilisateur manipule plusieurs esclaves.

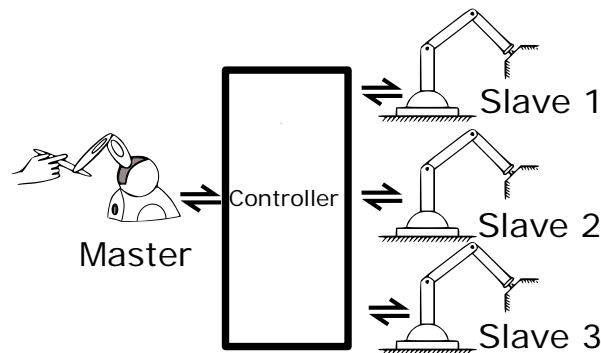


Figure 2-13 – Configuration SM/MS

Multi Master with Single Slave (MM/SS): plusieurs utilisateurs simultanés manipulent un seul esclave. Cette configuration peut être utilisée pour de la formation au geste ou de la comanipulation.

Multi Master with Multi Slave (MM/MS): plusieurs utilisateurs simultanés manipulent plusieurs esclaves. Cette configuration peut être utile pour collaborer dans le contrôle simultané d'une flottille

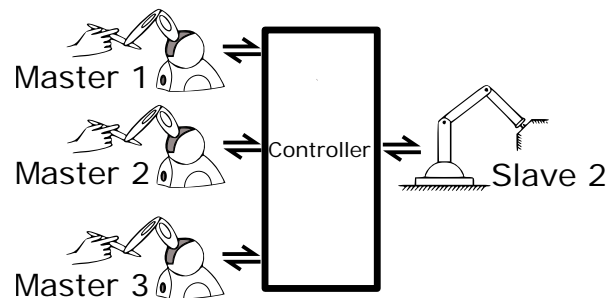


Figure 2-14 – Configuration MM/SS

de robots par exemple ou, dans le cadre de la formation au geste, d'utiliser une interface haptique et un outil esclave par main, éventuellement pour l'apprenant et le formateur

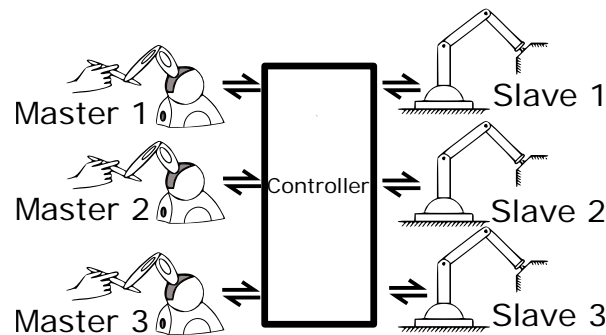


Figure 2-15 – Configuration MM/MS

2.4.2 Architecture Dual user

Une architecture typique utilisée pour implémenter un simulateur de formation supervisée au geste est la Dual-User, qui est un cas particulier de la configuration Multi-Master/Single Slave (MM/SS).

Dans ce cas, deux utilisateurs manipulent chacun une interface haptique (Master 1 et Master 2). Un esclave, connecté aux maîtres, interagit directement avec son environnement, les deux étant soit réels soit virtuels (voir figure Figure 2-16).

2.4.3 Partage d'autorité dans un contexte de formation pratique

A partir du moment où plusieurs utilisateurs manipulent un même esclave, un problème d'autorité surgit afin de trancher entre les consignes de position différentes entre les deux maîtres. Un mécanisme est donc mis en place pour partager cette autorité.

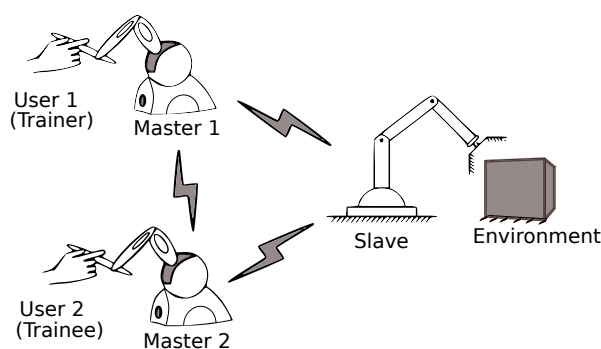


Figure 2-16 – Architecture Dual-user

(Khademian and Hashtudi-Zaad [2007]) a proposé l'usage d'une variable nommée α qui établit l'autorité de chaque dispositif maître sur l'esclave. Dans un contexte de formation au geste, trois situations sont envisageables (cf. figure 2-17).

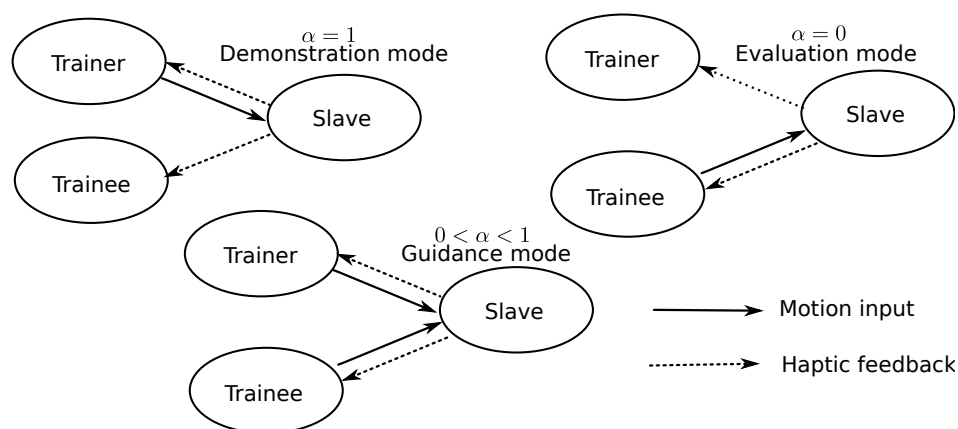


Figure 2-17 – Les situations de partage d'autorité dans un contexte de formation au geste

Mode Démonstration

Dans ce mode, $\alpha = 1$ et le formateur a toute autorité sur l'esclave, comme s'il était le seul utilisateur : il impose une trajectoire et ressent les éventuels efforts d'interaction issus de la simulation. L'apprenant est passif, il est guidé par son interface haptique et peut éventuellement ressentir les mêmes efforts d'interaction que le formateur si le contrôleur le permet et pour peu qu'il se place dans les bonnes conditions (voir section 3.1). Dans tous les cas, si ce dernier s'éloigne de la trajectoire imposée par le formateur, son interface haptique lui impose un effort lui indiquant dans quelle direction revenir sur la bonne trajectoire. Ainsi, le formateur peut enseigner un geste particulier en terme de trajectoires et d'efforts d'interaction dès lors que son outil entre en contact avec l'environnement de simulation. Il

est possible d'avoir un formateur virtuel reproduisant une trajectoire et une interaction pré-enregistrée.

Mode de guidage partagé

Dans ce mode, $0 < \alpha < 1$, l'apprenant et le formateur partagent l'autorité sur le système proportionnellement à la valeur de α . Plus α tend vers 0, plus l'apprenant prend de l'autorité sur l'esclave aux dépens du formateur. Par exemple, pour $\alpha = 0.3$, l'esclave se déplacera à une position correspondant à la moyenne pondérée des positions des maître avec une pondération de 70 % pour l'apprenant et 30 % pour le formateur.

Mode évaluation

Dans ce mode, symétrique du mode démonstration, $\alpha = 0$ et l'apprenant pilote entièrement l'esclave pendant que le formateur observe passivement les gestes de l'apprenant. Ce mode est utile pour que le formateur puisse évaluer les performances de l'apprenant.

2.4.4 Autorité adaptative

Le formateur peut à tout moment changer la valeur de α pendant la session de simulation. Cependant ceci peut être assez contraignant lorsque, en mode évaluation, il doit se concentrer sur les gestes de l'apprenant.

En effet, si ce dernier réalise un geste inapproprié, il est indispensable que le formateur puisse prendre la main instantanément et corriger le geste, s'il le désire, sans perdre du temps à modifier α . Afin de rendre ce type d'évolution de α plus facile, un mécanisme d'autorité adaptative peut être mis en place. La valeur de α définissant l'autorité de chaque utilisateur peut alors dépendre des performances de l'apprenant (autrement dit de l'écart de position entre l'apprenant et le formateur) pour redonner le contrôle au formateur le cas échéant.

Une fois l'apprenant revenu sur la trajectoire corrigée du formateur, celui-ci reprend automatiquement la main.

Une première fonction d'autorité adaptative a été proposée dans (Shahbazi et al. [2013]). D'autres auteurs ont proposé d'autres approches (Huang and Lu [2015]) et (Liu et al. [2015]). Toutes font appel à l'écart de position/orientation entre les interfaces haptiques de l'apprenant et du formateur.

2.4.5 Étude bibliographique des architectures de contrôle dans les simulateurs haptiques

Plusieurs architectures de contrôle ont été étudiées concernant les simulateurs d'apprentissage supervisé du geste. Nous présentons ici les travaux les plus pertinents vis-à-vis de notre cahier des charges et précisons quelles exigences, chacun omet de satisfaire.

1. Katsura et al. [2007] ont proposé une loi de commande multilatérale à retour d'effort pour des tâches de coopération. Leur solution convient aux dispositifs ayant des cinématiques différentes (par exemple, une translation pour le maître 1, deux translations pour le maître 2 et 3 translations pour l'esclave). Les auteurs ont fait appel à un contrôle en accélération, des observateurs de perturbation et des transformations spatiales pour faciliter la coopération entre les deux utilisateurs. Cette solution, bien qu'intéressante sur le fond, ne répond pas à notre besoin car le simulateur doit reproduire les gestes de chacun et non les compléter. **(R5 non satisfait)**.
2. Moghimi et al. [2008] ont également développé une architecture pour la coopération dans des environnements virtuels. Pour cela, ils ont intégré un contrôleur adaptatif et fourni une preuve de stabilité via une fonction de Lyapunov. Une validation expérimentale avec des dispositifs à 2 degrés de liberté est mentionnée. Les auteurs affirment que cette architecture devrait être testée pour un usage pédagogique **(R1 et R7 non satisfaits)**.
3. Shahbazi et al. [2011] ont développé une architecture dual-user utilisable avec des dispositifs à cinématiques différentes. Ils ont mis au point un contrôleur adaptatif partagé afin de compenser les incertitudes de la dynamique des dispositifs et les retards de transmission incertains. Cette solution fonctionne pour un usage coopératif mais est limitée pour la formation car quand l'utilisateur 1 est leader dans une situation d'interaction outil/environnement, le contrôleur ne garantit pas que le follower reçoive le même retour d'effort. **(R1, R5 et R6 non satisfaits)**.
4. Li et al. [2013] a conçu une architecture fondée sur un contrôle adaptatif à logique floue afin d'assurer la stabilité en présence de retards de transmission, des perturbations et des dynamiques incertaines des dispositifs. Le suivi est réalisé soit en position soit en effort pour chaque dispositif. Par exemple, le suivi en position de l'esclave est réalisé avec le maître 1 tandis que le suivi en effort l'est avec le maître 2. Cette solution n'est donc pas utilisable pour la formation du geste où les deux composantes position/force sont liées et doivent être transmises simultanément.

- ment entre les utilisateurs pour les besoins de démonstration ou d'évaluation. **(R1 et R6 non satisfaits)**.
5. Khademian and Hashtrudi-Zaad [2013] ont proposé une solution dont ils ont étudié expérimentalement la transparence sur des dispositifs planaires à trois degrés de liberté. Leurs contrôleurs garantissent la stabilité du système malgré les incertitudes liées à l'environnement et aux dynamiques des utilisateurs, grâce à l'usage du critère de stabilité inconditionnelle de Llewellyn. Les expérimentations ont été réalisées dans un objectif pédagogique. Cependant, nous pouvons nous interroger sur la stabilité de leur contrôleur pour des dispositifs agissant dans la troisième dimension du fait des effets de gravité. D'autre part, cette solution n'intègre pas de gestion de l'autorité entre utilisateurs, ce qui empêche de l'utiliser clairement en mode démonstration/évaluation **(R1 et R2 non satisfaits)**.
 6. Shahbazi et al. [2013] ont proposé d'utiliser des virtual fixtures pour former les étudiants en médecine à la chirurgie mini-invasive. Les auteurs utilisent un contrôleur d'impédance et ont intégré un mécanisme d'autorité adaptative. Une évaluation en temps réel des performances de l'apprenant détermine le niveau d'autorité. Cependant, si l'usage des virtual fixtures est valable pour apprendre des trajectoires, il risque de parasiter les retours d'effort issus de l'esclave lors d'une interaction outil/environnement **(R1 et R6 non satisfaits)**.
 7. Shahbazi et al. [2014] ont conçu une architecture fondée sur une logique floue pour des usages pédagogiques. Celle-ci fonctionne avec quatre niveaux d'expertise des utilisateurs: débutant, intermédiaire, avancé et expert. Le simulateur modifie les retours vers les apprenants en fonction de leur niveau d'expertise. Plus l'apprenant est expert, plus il reçoit de retour en effort à la place du formateur. Nous nous interrogeons sur l'intérêt d'augmenter progressivement ce retour d'effort ; l'apprentissage du geste avec des niveaux intermédiaires de retour d'effort ne risquent-ils pas d'être perturbants pour l'apprenant et moins efficaces pédagogiquement. Ce système ne permet donc pas de mode démonstration/évaluation pures où le follower peut ressentir les mêmes efforts que le leader et pas non plus de moyen pour le formateur de corriger les trajectoires erronées de l'apprenant **(R1 et R2 non satisfaits)**.
 8. Shahbazi et al. [2015] ont proposé une architecture MM/SS où un dispositif esclave est téléopéré pour réaliser une tâche coopérative. Les auteurs ont étudié sa passivité en tenant compte des retards de transmission (avec l'hypothèse que les retards sont identiques entre chaque dispositif).

Une loi de commande propose un retour d'effort à chaque utilisateur. Toutefois, leurs résultats expérimentaux montrent que lorsqu'un utilisateur (le formateur par ex.) a le contrôle complet de l'esclave, les autres utilisateurs ne sont pas guidés. Ce défaut n'est pas compatible avec notre cahier des charges (**R1, R5 et R6 non satisfaits**).

9. Shamaei et al. [2015] ont développé une architecture trilatérale pour un usage pédagogique. Ils ont proposé d'utiliser deux facteurs d'autorité (α et β , afin de passer en mode démonstration et évaluation) et un contrôleur à logique floue pour améliorer les performances de l'architecture de Shahbazi et al. [2014]. Cependant ce contrôleur ne propose pas de mode intermédiaire ni d'autorité adaptative (**R1 and R2 non satisfaits**).
10. Huang and Lu [2015] ont proposé une architecture qui sépare le facteur d'autorité en trois sous-facteurs afin d'obtenir un contrôle partagé de l'esclave dans lequel un utilisateur protège les gestes du second. Ils ont introduit un mécanisme adaptatif pour aider les utilisateurs dans le réglage des trois variables. Ils ont réalisé des expérimentations faisant appel à des maîtres différents (un Phantom Omni (3 ddl actifs) et un Omega (2 ddl actifs)). Cependant, cette architecture présente la même faiblesse que celle de Shahbazi et al. : l'absence de possibilité de transmettre entre utilisateurs l'information de l'effort à appliquer lors d'une interaction outil/environnement (**R1 et R2 non satisfaits**).
11. Zakerimanesh et al. [2017] proposent une architecture dual-user avec un contrôleur non linéaire proportionnel+amortissement (nP+D) initialement proposé pour les configurations de téléopération (SM/SS). Cette stratégie de contrôle permet de tenir compte des saturations des moteurs et dépend de deux facteurs d'autorité : α qui détermine la suprémacie du formateur sur l'apprenant, et β qui détermine la suprémacie de chaque utilisateur sur l'esclave. Ces deux facteurs sont en pratique des pondérations utilisées par le contrôleur afin de calculer les efforts que chaque dispositif doit générer, en fonction notamment des erreurs de positions relatives et de la performance de l'apprenant. Ils ont démontré la stabilité et simulé l'ensemble sur un robot planaire à 3 ddl (**R1 et R7 non satisfaits**).
12. Panzirsch et al. [2017] ont développé une architecture à des fins coopératives avec laquelle les utilisateurs doivent réaliser des tâches complexes sur un esclave distant. Cette solution a la propriété de fournir à chaque utilisateur une sensation répliquant les mouvements de chaque partenaire. Par exemple, les auteurs réalisent une tâche de levée d'une boîte. Dans la vie

courante, deux humains peuvent sentir les mouvements de leur partenaire soulevant la même boîte. Leur sens kinesthésique leur permet ainsi de synchroniser leurs efforts et mouvements. Cette architecture tente d'imiter ce retour kinesthésique via des interfaces haptiques. Cependant la tâche de coopération n'est pas toujours adaptée à un usage pédagogique et surtout au cas d'utilisation proposé dans ce document : aucun utilisateur n'est leader, ce qui rend difficile la réalisation de démonstrations et d'évaluations (**R1, R2, R5 et R6 non satisfaits**).

Le tableau 2.1 synthétise les caractéristiques des architectures mentionnées ci-dessus. Il est ordonné chronologiquement et les noms des auteurs permettent de les identifier clairement car ces solutions n'ont pas toujours un nom. La colonne « DoF » indique le nombre de degrés de liberté testés (ou proposés par les auteurs quand précisés). Nous rappelons dans ce tableau les avantages et inconvénients principaux de chaque architecture et la colonne « Training purpose » détermine si chaque solution peut être utilisée à des fins pédagogiques compatibles avec notre cas d'utilisation.

Table 2.1 – MM/SS architectures comparison table

No.	Authors	DoF	Avantages	Inconvénients	Training purpose
1	Katsura et al. [2007]	n	Fonctionne avec des dispositifs à cinématiques différentes	Les apprenants ne peuvent pas reproduire les mêmes gestes que le formateur	
2	Moghimi et al. [2008]	2	Fonctionne avec des dispositifs de type admittance et impédance	Validé que sur 2 ddl hors contexte pédagogique.	✓

Table 2.1 continued from previous page

No.	Authors	DoF	Advantages	Disavantages	Training purpose
3	Shahbazi et al. [2011]	1	Fonctionne en présence de retards de transmission et d'incertitudes sur les modèles (contrôle adaptatif)	Pas de certitude que les apprenants puisse recevoir le même retour en effort que le formateur	
4	Li et al. [2013]	n	Fonctionne en présence de retards de transmission et d'incertitudes sur les modèles.	L'effort et la position sont gérées par deux dispositifs haptiques séparés pas conjointement.	
5	Khademian et al. [2013]	3	Fonctionne malgré les incertitudes sur l'environnement et les utilisateurs.	Testé uniquement avec des dispositifs planaires. Pas de gestion d'autorité	✓
6	Shahbazi et al. [2013]	n	Évaluation en temps réel de la performance des apprenants	Ne traite que de la trajectoire, pas des interactions outil/environnement	

Table 2.1 continued from previous page

No.	Authors	DoF	Advantages	Disavantages	Training purpose
7	Shahbazi et al. [2014]	1	Modifie le retour de l'apprenant selon son expérience	Que le mode démonstration	✓
8	Shahbazi et al. [2015]	1	Fonctionne en présence de retards de transmission	Pas de synchronisation entre effort et position pour les apprenants	
9	Shamaei et al. [2015]	1	2 paramètres indépendants d'autorité	Pas de mode intermédiaire mode	✓
10	Huang and Lu [2015]	2	Fonctionne avec des dispositifs à cinématiques différentes	Pas de transmission des efforts de du formateur à l'apprenant	
11	Zakerimanesh et al. [2017]	n	Tient compte des saturations des moteurs et propose 2 facteurs d'autorité	Testé uniquement avec des dispositifs planaires.	

Table 2.1 continued from previous page

No.	Authors	DoF	Advantages	Disavantages	Training purpose
12	Panzirsch et al. [2017]	n	Étudié pour des tâches où les utilisateurs sont en interaction (port de colis par ex.)	Ne fonctionne qu'en coopération, pas en démonstration /évaluation	

La plupart des architectures mentionnées ci-dessus, est conçue selon un schéma trilatéral ou dual-user qui ne se prête pas facilement à une extension à d'autres dispositifs (maîtres ou esclaves), ce qui ne répond pas à l'exigence **R1**. En effet, l'ajout de nouveaux utilisateurs ferait monter la complexité de manière exponentielle, ce qui limiterait leur viabilité et donc utilisabilité. D'autre part, aucune ne démontre la possibilité de former l'apprenant à des gestes impliquant des interactions outil/environnement et permettant donc à l'apprenant de suivre une trajectoire et de sentir une telle interaction.

2.5 Conclusion de cette étude

Les interfaces haptiques commencent à être utilisées par les simulateurs informatiques en vue de la formation pratique des médecins. Ils permettent aux apprenants d'apprendre des gestes médicaux avec un rendu kinesthésique de plus en plus réaliste, sans toucher un patient.

Toutefois, les simulateurs du commerce ne proposent d'interface que pour l'apprenant ; le formateur reste en dehors de la simulation avec la seule possibilité de guider les mains de l'apprenant guidant les interfaces haptiques. Ce mode de formation aide à transmettre une trajectoire mais ne permet en aucun cas de montrer directement les efforts à exercer. Seule l'intégration du formateur dans la simulation, via ses propres interfaces haptiques peut permettre cette transmission via le canal kinesthésique, pour peu que le contrôleur du simulateur soit conçu pour, ce qui n'a pas été constaté lors de cette recherche bibliographique.

Ainsi, le simulateur « idéal » remplissant l'ensemble des exigences précisées dans ce document

n'existe pas encore. Nous avons donc tenté de nous en approcher via les contributions détaillées dans la suite de ce rapport.

Chapter 3

Résumé des contributions

La première contribution de ces travaux a consisté à étendre l'architecture ESC (Energy Shared Control) proposée initialement par M. Fei LIU (cf figure 3-1, page 71). Celle-ci est constituée d'une interconnexion de composants passifs réalisant ainsi un système globalement passif. La compliance nécessaire à tout système haptique est apportée par l'usage de contrôleurs de type IPC (Intrinsically Passive Controls). Ces contrôleurs ont aussi pour fonction de passiver l'environnement et l'utilisateur follower du point de vue du reste du système. Ces trois contrôleurs sont interconnectés par un réseau de connexions dynamiques permettant de modifier de manière continue et à tout moment l'autorité des deux utilisateurs sur l'esclave sans remettre en cause la passivité du système. Les détails de conception sont fournis dans (Liu et al. [2015]).

Une étude de la transparence d'ESC, détaillée au chapitre 3, a été menée afin d'optimiser les paramètres des contrôleurs IPC et obtenir le meilleur compromis entre stabilité et transparence (qualité de suivi en position et en effort des différents dispositifs entre eux). En utilisant un modèle linéaire de la chaîne de téléopération liant l'utilisateur leader et l'environnement (à travers l'interface haptique maître, les contrôleurs IPC, les interconnexions et le robot esclave) présenté sous forme d'une matrice hybride, nous avons approximé le comportement du système à basse fréquence sachant que les mouvements des chirurgiens sont rarement saccadés et déterminé les plages de valeur acceptables pour maintenir la stabilité et obtenir une transparence maximale en position (en régime libre) et en effort (lors des phases de contact).

Une première extension à n degrés de liberté a été réalisée en dupliquant cette architecture sur chaque articulation des dispositifs maîtres et esclave, sachant que cette approche ne convient que pour des dispositifs identiques en terme de géométrie. D'autre part, il a été supposé que les couplages

éventuels entre les différentes articulations sont absorbés par les contrôleurs IPC. Étant donné que le dispositif esclave risque fortement de ne pas avoir la même cinématique que les dispositifs haptiques maîtres, la deuxième approche a consisté à dupliquer ESC sur chaque degré de liberté considéré dans l'espace cartésien (3 translations et 3 rotations). Les premières expérimentations ont mis en relief que le paramétrage optimal en terme de compromis stabilité/transparence des contrôleurs IPC diffère entre approche articulaire et cartésienne. Nous avons donc déterminé méthodiquement tous les paramètres des 3 contrôleurs IPC en tenant compte de l'amortissement désiré des mouvements afin d'éviter des trajectoires oscillantes ou trop amorties. Nous avons simulé les deux architectures d'après une trajectoire circulaire de référence dans l'espace cartésien afin de confirmer nos choix. Ceux-ci mènent à un suivi en position tout à fait acceptable avec l'approche articulaire et présentant des transitoires légèrement oscillants au début de la simulation puis un régime permanent stable et tout aussi acceptable pour l'approche cartésienne. Nous avons ensuite mis en place un banc d'essai utilisant 3 interfaces haptiques Geomagic 3D Touch interconnectées via un PC faisant tourner Matlab Simulink en temps réel. La bibliothèque de fonctions OpenHaptics a été enrichie afin de pouvoir communiquer simultanément avec les 3 interfaces. Un environnement d'interaction spécifique a été fabriqué en impression 3D afin de reproduire des trajectoires circulaires en relief permettant de mettre en valeur les propriétés de suivi en position et en effort (cf. figure 3-17, page 91). La figure 3-18, page 92, illustre l'organisation spatiale de ce banc et l'environnement proposé à chaque utilisateur. L'architecture informatique est synthétisée en figure 3-19, page 92. Une interface utilisateur 3D a été programmée grâce aux bibliothèques de fonction Chai3D afin d'aider l'utilisateur follower à se placer à la même position que le leader pour ressentir les bons efforts d'interaction (figure 3-21, page 94). Une méthodologie de réglage optimal des paramètres expérimentaux des contrôleurs IPC est proposée et appliquée.

Les validations expérimentales ont consisté à réaliser des sessions incluant :

- des passages entre modes de démonstration et évaluation afin de démontrer qu'un changement abrupt d'autorité n'induit pas d'instabilité ;
- des trajectoires en régime libre afin de tester la qualité du suivi en position des différents dispositifs et le retour haptique qui ramène l'utilisateur follower sur la trajectoire du leader ;
- des phases d'interaction outil/environnement afin d'évaluer la qualité du retour en effort du leader par rapport à l'effort outil/environnement et du follower quand il positionne adéquatement son interface haptique ;

- des essais avec l'architecture articulaire et cartésienne (dans ce cas avec un dispositif esclave de type Virtuouse 6-Desktop d'Haption).

L'ensemble de ces tests expérimentaux montrent des performances acceptables étant donné que les interfaces haptiques utilisées sont plutôt de bas de gamme : erreur moyenne en position de 2,5mm et erreur en effort sous le seuil de précision (moins d'1 N). Il est à noter que la mesure des efforts est réalisée d'après les données transmises par la bibliothèque de fonctions OpenHaptics, d'après une estimation fondée sur la consommation des moteurs de chaque articulation. Des essais avec de réels capteurs d'effort est à envisager pour une validation encore plus probante.

Le chapitre 4 détaille les améliorations apportées à la fonction d'adaptation automatique de l'autorité entre utilisateurs. Nous rappelons le principe exposé par M. Fei LIU et nous avons proposé une nouvelle fonction pour déterminer α en fonction de la distance entre les interfaces des deux protagonistes. Quatre profils ont été proposés, sélectionnables par le formateur, afin d'offrir à l'apprenant plus ou moins de liberté d'action en mode évaluation. Cette évolution a été testée expérimentalement en régime libre et en interaction avec l'approche articulaire. Afin de valider l'intérêt de cette fonction du point de vue pédagogique, un retour d'usage a été mis en place. Des exercices concrets amenant à faire apprendre aux apprenants des trajectoires et des efforts ont été proposés et testés par 18 utilisateurs novices (seulement neuf d'entre eux avaient accès à cette fonction). Les conclusions de cette première étude n'ont malheureusement pas permis de déterminer de plus-value à cette fonction. Une discussion approfondie précise les facteurs ayant pu polluer les résultats expérimentaux afin de proposer un nouveau protocole expérimental pour une prochaine étude d'usage.

La contribution suivante, détaillée au chapitre 5, a consisté à étendre le nombre d'apprenants pouvant utiliser simultanément le simulateur. Cette extension ouvre la voie à des démonstrations réalisées pour un public plus large avec les mêmes propriétés de formation aux trajectoires et aux efforts qu'en mono-apprenant. L'approche théorique est détaillée et implique d'ajouter un nouveau paramètre δ afin de sélectionner l'apprenant leader en mode évaluation. Les autres apprenants sont alors followers et peuvent suivre simultanément (et éventuellement évaluer) leur collègue toujours sur les mêmes critères de performance trajectoire/efforts. Le formateur a toujours les mêmes outils qu'en mono-apprenant : la possibilité d'évaluer la performance de l'apprenant sélectionné et de pouvoir corriger sa trajectoire (AAA). Des expérimentations ont validé cette approche. Faute d'un nombre suffisant d'interfaces haptiques, nous avons dû simuler en logiciel l'une d'entre elles. Cependant, cela nous a permis de montrer qu'une réaction idéale d'un utilisateur mène à de très bonnes performances

du système (en terme de positionnement précis par rapport au leader et donc de retour d'effort précis).

Ce document s'achève par une conclusion synthétisant l'ensemble de ces contributions, leurs performances et leurs limites. Nous rappelons quels objectifs initiaux ont été remplis et nous proposons deux pistes d'améliorations pour la suite de ce projet.

Part II

Background

Chapter 1

Introduction

This work is introduced in a medical context as this one is particularly illustrative. Yet, it can apply to any other hands-on training context where the training of gestures including trajectories and interaction forces is necessary.

1.1 Problem definition

The training of medical students is a long process that requires several hours of practice and study to ensure the acquisition of knowledge and dexterity. To master the palpation, injection, surgeries, or any other processes that involve direct interaction with the patient, students need to practice many times. However, medical students can't practice with patients, due to the ethical problem that it entails, such as injuring or permanently damaging the patient (Beal et al. [2017]). For this reason, there is still a crucial need for medical students to improve their manual skills without touching a real patient.

Medical students have been using training simulators, which help them acquire the necessary experience by providing pseudo-realistic sensations of the behavior and reactions of the human body. Some simulators are based on wood, leather, metal, or corpses and have been used for centuries. An example is the midwifery simulator "Phantom" developed in the 18th century in France by the obstetricians Gregoire and son. This early obstetrician simulator used a real human pelvis with the corpse of an infant to replicate the delivering child process (Owen [2012]).

Nowadays, medical students can practice several surgeries using corpses; however, the use of cadavers involves several problems (Berger et al. [2002]):

- Corpses have no biological reaction, so medical students cannot determine if they are injuring

them.

- It is necessary to have the consent of the patient or the family to harvest and operate the cadaveric organs, which complicates their acquisition for training purposes.
- It requires equipment to preserve them from rotting.
- Corpses are disposable, as the tissues are permanently altered and cannot be restored.

Besides corpses have been used for training simulations, over time, mannequins-based simulators appeared in the early 1960s. An example is *Resusci-Anne*, a training mannequin for practicing cardiopulmonary resuscitation that was not widespread due to its high cost (Fritz et al. [2008]). After that, researches began to focus on the rendering of the human body's vital signs for training sessions. The University of Miami created the famous Harvey mannequin, which was used to train cardiology examination and electrocardiography, beginning the era of mannequin-based training simulators (Kunkler [2008]).

The use of mannequins (Figure 1-1) leans a representation of the patients' behavior more realistic than with corpses, as it is now possible to the recreate critical events during the operation. One example is the mannequin named CASE (Cooper and Taqueti [2008]), developed in early 1987, was used for anesthesia training and allowed to simulate the patient vitals using virtual instruments in a computer system. The mannequin is manipulated by the trainers who create situations where the life of the patient is in danger, so students can make decisions and learn from them.



Figure 1-1 – Mannequin based simulators

In more recent years, medical students have had access to more advanced and realistic simulators based on mannequins which recreate several vitals of the patient and even allow them to practice sophisticated methods like endoscopy (Deutsch [2008]). These simulators now use a combination of mechanisms, electronic systems, and software algorithms to recreate the response of the patient during

the operation. The use of mannequins provides several advantages over the corpses like uncommon scenarios that can be created, and there is no shortage of supplies (Fritz et al. [2008]). However, as they are based on a fixed mechanism, the mannequins cannot be modified to offer more variety of training. For example, they cannot provide surgery training, where the organs of the patients must be modified.

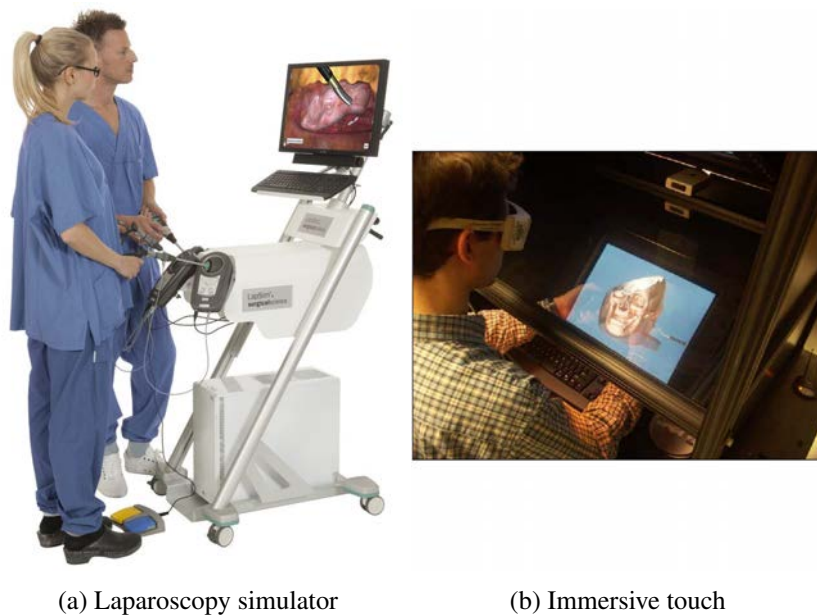
Thanks to the technological advances, a new generation of training simulators solves some of the aforementioned problems. The first computer-based simulators proposed virtual environments where students could interact with the patient's organs. For this, the simulator uses biomechanical mathematical models of the human body to simulate its behavior. However these simulators do not offer force reaction feedback, so the students cannot get a realistic immersion in the operation (Neumann et al. [1998] and Tsai et al. [2001]).

These simulators provide advantages over the classical ones like the use of virtual environments where students can practice with several kinds of bodies that vary in shape and weight; students can repeat the procedure several times in the same virtual environment without getting short of supplies (Sutherland et al. [2013]). For this reason, the use of computed simulators has become the new paradigm in the process of teaching medical students (Jones et al. [2015]). However, one significant drawback is the cost of the use of these simulators which can rise to \$1,000,000 (Zevin et al. [2014]).

More recent computed simulators implement haptic devices (detailed in section 2.2) that allow them to manipulate a virtual tool to operate the virtual model. At the same time, they receive force feedback from the operation. The models used in these simulators tend to behave like the real-life counterparts in matters of deformation and force feedback, which provides the student an experience close to a real human operation.

An example of a haptic simulator used in medical training is the *Lapsim*® (Figure 1-2a) developed by SurgicalScience. This simulator is used to train medical students for laparoscopic surgeries. The users interact with a virtual world displayed on the screen and manipulate two haptic devices to input their hand motions into the virtual environment while they receive force feedback at the same time.

Another example is Immersive Touch (Figure 1-2b), a training system developed by the University of Illinois, which utilizes a transparent mirror to project 3D models in space and haptic devices to provide force feedback. The user must wear head-tracking glasses to register her/his position and orientation to adjust the stereoscopic perspective of the projected model. This method of rendering the scene allows the student to move her/his head around the virtual environment which provided



(a) Laparoscopy simulator

(b) Immersive touch

Figure 1-2 – Commercial simulators

different viewing perspectives, which is essential to find important landmarks (Cristian et al. [2005]).

1.1.1 The problem of non-guided training

Although medical simulators should allow students to self-teach, they could develop bad techniques. Because of this, it is still required the guidance of an expert whose motion feedback and commentaries lead them since their first sessions to improve their learning (Schout et al. [2010]). This approach, where a trainer can manipulate and correct the student movements, is called **guided training**. In a guided training session, the trainees' hands are conveyed by the trainer during the instruction. In real-life applications, this is made by putting the hands of the trainer over the ones of the trainee. However, this is very impractical for gesture training with computer-based haptic simulators. As the trainer holds the trainee's hands, it is not possible to determine which user is manipulating the slave, and how much force each user perceives from the environment. This last property is a very crucial factor in real-life operations where the patient integrity can be affected.

Because of this, it is necessary that each user has access to a haptic device when they perform the training. Based on a bibliographical review, most commercial simulators do not include a second haptic console that can be used by the trainer during the teaching process. Some commercial simulators focus on the instruction of double-handled operations, but they do not provide an option of guided training mode, as the virtual tools in the environment are completely independent and the only

interaction between them is when a collision occurs (Escobar-Castillejos et al. [2016]).

Some researchers have tried to cope with this lack of dual-user manipulation functions for guided training systems. Their results are shown in section 2.4.5 where I explain why they still not fulfill guided training and expectations when dealing with force levels.

1.1.2 Main use case and main requirements

The following typical use case helps determine the main requirements of the system. Suppose, at first, that the trainer (an experienced surgeon) aims at demonstrating the right trajectories of her/his surgical tool, which involves performing tasks featuring free motions and some tool-environment contacts. This implies that she/he requires realistic force feedback to dose her/his force, as in a bilateral teleoperation context. The trainer manually sets the compliance of the system to become **leader** (the trainee becomes **follower**): it is a **demonstration mode**. She/he then gets full-force feedback from the slave to perform the task as if she/he were handling the real instruments. Meanwhile, the trainee's device follows the trajectory of the leader one. If the trainee deviates from this reference trajectory when the system is in free motion, the compliance of the device brings her/him back to the right position. In case of interaction between the tool and its environment, the trainee can also feel in her/his hands the right level of effort to provide to the tool, using a display which guides her/his to set her/his device at the right position with the right feedback force. Afterward, the functioning can be inverted by reversing the compliance so that the trainee manipulates and the trainer follows and evaluates trainee's motions and applied forces: it is an **evaluation mode**.

Also, the simulator is not limited to only two users. An extended system with m -trainee in parallel teaching is achievable. A trainer can then teach several trainees at the same time (Figure 1-3b), instead of one by one (Figure 1-3a). Furthermore, a student can present her/his skills to several evaluators and get quantitative feedback from the system, and qualitative feedback from the evaluators. This concept offers a more efficient and less consuming teaching time, as students spend more time practicing with the teacher than waiting for her/his turn on the machine.

Fig 1.5 synthetizes this main use scenario with a Use Case Diagram following the SysML standard. This scenario leads to the following requirements for a training simulator enabling efficient motion and force haptic training.

Based on the Case Diagram the requirements to fulfill are:

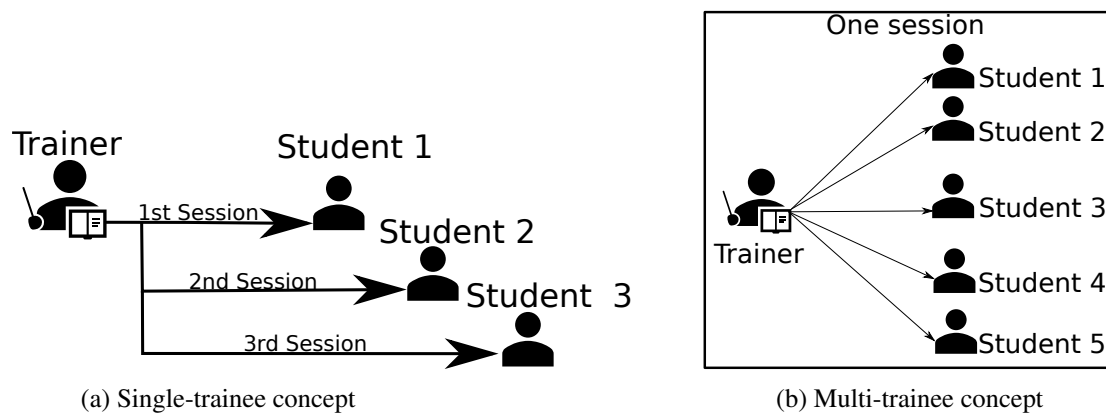


Figure 1-3 – Trainee teaching concepts

Requirements

- (R1) **Multi-trainee:** The simulator must be able to be operated by several users. From those users, one is the trainer, and the remaining are the trainees.
- (R2) **Training mode selection:** The trainer can set the training mode, which assigns the leadership on the slave to the trainer, the trainee, or the cooperation of both.
- (R3) **Adaptive leadership:** The simulator must be able to adapt the user leadership between the trainer and the selected trainee, based on the performance of the last one.
- (R4) **Trainee leader selection:** The trainer can also select the leading trainee, who is the student that will control the slave.
- (R5) **Position tracking:** The system applies a correctional force when one of the following users deviates from the leader trajectory in order to keep her/him on it.
- (R6) **Force tracking:** The slave directly interacts with the environment and gets force responses from it. These force responses are sent to all the users, no matter the training case or the leader trainee selected.
- (R7) **Compatibility:** The simulator must allow the use of any haptic devices with any degrees of freedom (DoF), no matter their kinematic configuration.
- (R8) **Stability:** The simulator must be stable at any moment, regardless the user's motions or slave interactions with the environment.

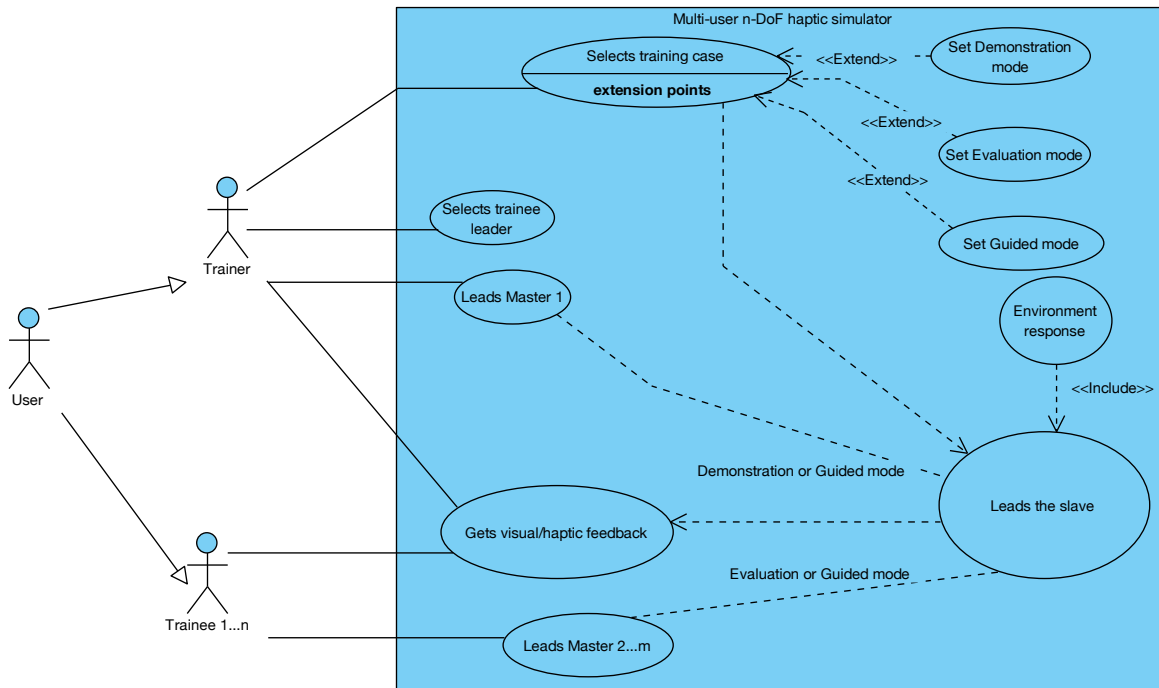


Figure 1-4 – Guided training simulator: Case Diagram

From these 8 requirements, we deduce that a control architecture that connects several haptic devices with a slave must be developed for the accomplishment of the multi-trainee simulator.

1.2 Objectives of the thesis

The objectives of the Robotics Working Group of Ampere lab are to design a dual-user haptic training system able to provide guided force and motion hands-on training. In this project, a first architecture has been introduced by Fei Liu between 2013 and 2016 during his Ph.D., the Energy Shared Control (ESC) (Liu et al. [2015]).

ESC utilizes Intrinsically Passive Controllers (IPC) (details in section 3.2) for compliant control, it is composed of a passive interconnection of passive components to ensure its passivity and therefore its stability (more details about passivity in appendix C). Experiments show good transparency and the effectiveness of guided motion and force hands-on training with 1-trainee in a 1-Dof system.

The main advantages of ESC and requirements it already fulfills:

Advantages

- It allows to work in three different training cases (R2).
- It provides an adaptive dominance function (AAA) based on the performance of the trainee (R3).
- It is able to transmit the dosing force from the trainer to the trainee when making the same motions (R5).
- The force feedback of the master devices is a function of the slave interaction with the environment and not an amalgam of signals from other masters and the slave (R6).
- As it is based on a Time Domain Passivity Control (TDPC, details in appendix C) global stability is ensured all time (R8).

The main disadvantages of the ESC architecture are:

Disadvantages

- It has been tested only in 1-DoF.
- The AAA depends on four variables that are difficult to tune by trainers.

Based on the disadvantages of ESC, my PhD objectives are the following:

Objectives

- (O1)** To extend the ESC architecture into any degrees of freedom (R7)
- (O2)** To extend the system to a multi-trainee approach, that allows adding more haptic devices without redesigning it (R1), and provide a leader trainee selection mechanism (R4)
- (O3)** To propose an enhanced AAA function with an easier tuning process for the trainer that preserves (R3).
- (O4)** To ensure that the extended ESC architecture preserves its position and force tracking (R5 and R6).
- (O5)** To permit using haptic devices with different kinematics than slave robots (R7).
- (O6)** To guarantee that the extended ESC architecture preserves stability (R8).

1.3 Contributions

To achieve these objectives, we carried out a state-of-the-art survey (presented in section 2.4.5), which led us to determine the disadvantages of each architecture found in the literature and helped us determine that the extension and improvement of the ESC architecture was the correct approach to achieve a multi-trainee simulator.

First, we extended the ESC architecture to work in n -DoF. As this architecture was only tested in 1-DoF, this did not provide a way to develop a fully operational simulator. The expansion of this architecture is shown in section III-1 and led to a publication in *Automatica* in 2019 (Liu et al. [2019]). We also experimentally validated that architecture can work with devices with different kinematics. For this, we tested it with three haptic devices, two identical for the masters and one with different kinematics for the slave.

The original Adaptive Authority Adjusting (AAA) function proposed with the ESC architecture (Liu et al. [2015]) could permit the trainer to get back authority over the slave when the trainee is performing badly. We propose a new adaptive function that only uses one parameter gain and a profile. This new AAA function adapts the dominance based on the performance of the trainee compared in real-time to one of the trainers, for each task. We also carry out experimental tests with 20 subjects to determine if the AAA function improves training efficiency. .

Finally, once the new architecture was able to work in 3-DoF (Articular and Cartesian space) and had a newly implemented AAA function, we decided to expand it into an m -trainee architecture. This expansion has the property to preserve the stability and transparency of the original ESC. This work has been presented at IROS 2019 conference (Licona Rodriguez et al. [2019]).

1.4 Document Outline

This thesis document is organized as follows:

In chapter II we introduce haptic devices, how they work, and a classification of them. Then we introduce the actual state of the art of computer-based simulators that use haptic devices for medical training, and the architectures that are used to develop them. Also, we explain why these architectures were not chosen for our application and why we opted for a previously proposed one corresponding to the different cases proposed for training.

In section III-1 first, we introduce the initial ESC architecture with its controller. We also explain

how it works, and its fundamentals. We then introduce its expansion and validation to n -DoF according to two approaches in Articular and Cartesian spaces.

In section III-4 we introduce how we enhanced the Adaptive Authority-Adjusting function that allows changing the dominance of the trainer and the trainee in real-time, initially proposed in Liu [2016]. We explain how this new version provides a simpler way to use for the trainer. We then detailed the evaluation study of the benefits of this function and discuss the obtained results.

In chapter III-5 we expand the architecture presented in previous chapters into a m trainee system and we expose the experimental results.

Finally, in part IV, we present our conclusion about this thesis work and future directions for the improvement of this training system.

Chapter 2

State of art

2.1 Introduction

Humans can remember quite accurately the position of their extremities; this is called Kinesthesia (Feygin et al. [2002]). Kinesthesia comes from the Greek *kinein*, “to set in motion”, and *aisthesis*, “perception” which is interpreted as the perception of motion. Kinesthesia is a conglomeration of three different sub-senses: orientation and position, motion, and force (Rosker and Sarabon [2010]).

Using kinesthesia, humans can get cognitive knowledge from a *motorpsycho* approach and learn to perform gestures or postures that serve for a particular task (Xu and Ke [2014]). One example of this approach is learning how to play an instrument. Musicians learn how to read music sheets and how every sound conjoint together to form a pleasant melody, but only through repetition, they master the ability to reproduce the music (Galvao and Kemp [1999]). Using kinesthesia, humans can have focused kinesthetic training, added to a visual or auditory one, and therefore get more significant learning, especially for learning precise gestures (Panait et al. [2009]).

Considering this, the technological advances in virtual reality, haptic devices, and augmented reality have helped to the development of haptic simulators where a student can interact with a virtual environment via a joystick, glove, or any input device to get force feedback of her/his actions. These simulators are used in several fields like aeronautics, engineering, and any other field of study where manual gestures are applied. Their use for training is becoming widespread.

2.2 Haptic devices

A haptic device is a system that intends to replicate the sensation of feeling or touching a virtual object. For instance, these devices are used for the creation of virtual worlds where vision, hearing, and touch are replicated to provide a more immersive experience (Figure 2-1).



Figure 2-1 – FundamentalVR immersive simulator for Knee Arthroplasty ¹

The interaction between haptic devices and the user is a closed-loop process that begins with the motion input of the user on the haptic device. The haptic device senses its new position and sends it to the virtual world. In the virtual world, the cursor or virtual tool updates its position and, depending on if there is or not an interaction against a virtual object, force feedback is calculated as well the aftermaths of the collision like deformation, displacement, or sectioning. The updated virtual world is rendered and displayed on the screen. The user gets the visual feedback, and at the same time, the force feedback is commanded to the haptic device, which in turn exerts it to the user. For better comprehension, figure 2-2 shows a diagram of this process.

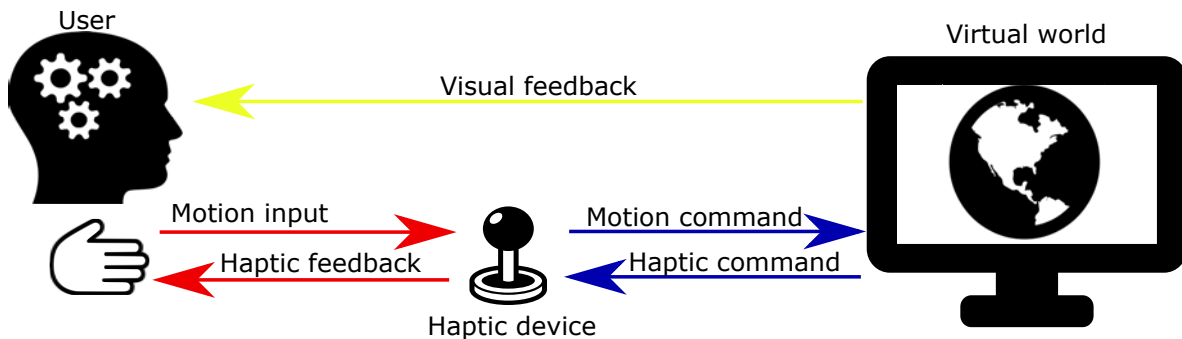


Figure 2-2 – Haptic environment diagram

¹The Verge. Consulted April 21, 2020 URL: <https://www.theverge.com/2018/8/14/17670304/virtual-reality-surgery-training-haptic-feedback-fundamentalvr>

2.2.1 Haptics device classification

Haptic devices are used in industrial tasks, research assignments, and daily life gadgets. Because of this, there are multiple ways of categorizing. The first form of classification is based on their use: Consumer or Industrial (Sreelakshmi and Subash [2017]).

The first category covers low-cost haptic devices added to daily use devices like smartphones, joysticks, and steering wheels. They work by using motors with augmented inertia like the one presented in Figure 2-3. This one is used to alert or give a vague sensation of a collision, for example when something occurs on the screen, like a notification from an application or the explosion in a videogame.



Figure 2-3 – Videogame controller with force feedback motors

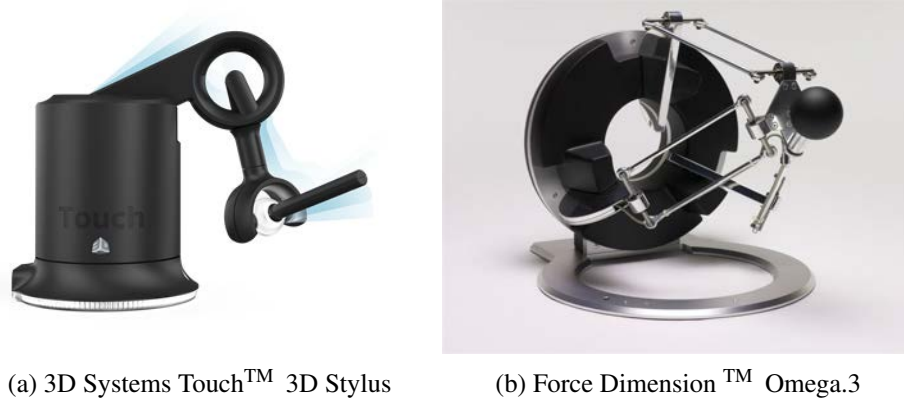
The second category is the industrial haptic devices that include a wide range of commercial or designed devices. These devices are used for complex tasks like scientific applications, virtual sculpture, training simulators. There is a wide variety of industrial haptic devices on the market. Most of them have a kinematic chain configuration (Figure 2-4a) although there are also parallel mechanism systems (Figure 2-4b), all of them providing up to 6-DoF of motion.

The industrial haptic devices are also classified by their use and feeling objective (Culbertson et al. [2018]).

Graspable systems: Haptic devices that allow giving the sensation of pushing or pulling an object.

These systems can also be used for deformation and dissection sensation. These devices are normally grounded and have a staff or holder, which is called a tool or pen and used by the user to activate it and get the force feedback from the virtual object (Figure 2-5a).

Weareble systems: Haptic devices that allow giving the sensation of grabbing objects. These systems



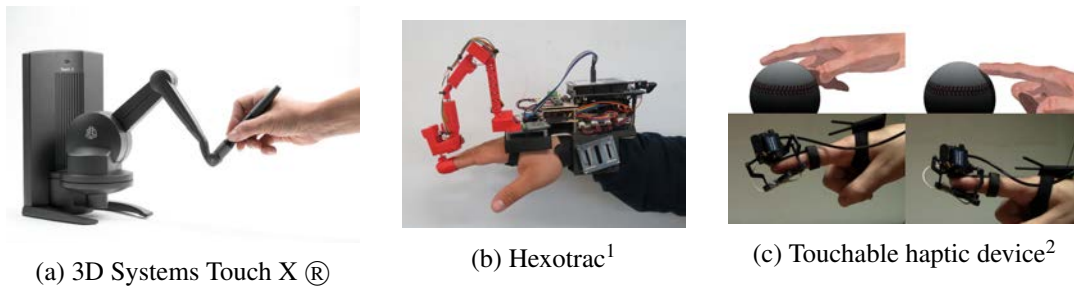
(a) 3D Systems Touch™ 3D Stylus

(b) Force Dimension™ Omega.3

Figure 2-4 – Serial and parallel mechanism haptic devices

are normally attached to the body and work by limiting the movement of the user using different types of actuators (Figure 2-5b).

Touchable systems: Haptic devices that allow giving the sensation of exploring a surface. Albeit the graspable devices also allow giving this sensation by using the tool, touchable systems achieve this by interacting directly with the cutaneous nerves that the human skin has (Figure 2-5c).



(a) 3D Systems Touch X®

(b) Hexotrac¹(c) Touchable haptic device²

Figure 2-5 – Haptic Devices Classification

2.2.2 Advantages and disadvantages

The advantages of each configuration depend on the specific task, but common limitations are:

- costly investment
- limited force and unequal in all directions
- limited bandwidth for the data transmission

¹(Sarakoglou et al. [2016])

²(Perez et al. [2015])

- the haptic signal must be rendered at least at 1 Kilohertz (Tavakoli et al. [2007]), which could limit the simulation loop for very complex virtual environments.

2.3 Medical haptic training simulators

As medicine is a practical area, many procedures use the touch sensitivity of the doctor. Haptic training simulators are developed to aid medical students to practice procedures and repeat an endless amount of time without being short of supplies, unlike the one-time-use training solutions like corpses. There is a wide variety of medical simulators that cover various medicine areas (Escobar-Castillejos et al. [2016]). Some examples of these haptic simulators and their use are:

Stitching simulators: used to teach the medical students how to stitch skin for healing wounds. It uses two 6-DoF haptic devices to guide a virtual needle and scissor. The haptic devices allow positioning and rotating the tool in the Cartesian space. This simulator uses finite element meshes to calculate the force feedback sent to the student. An example of these simulators is the Suturehap (Figure 2-6) presented in (Escobar-Castillejos et al. [2016]).

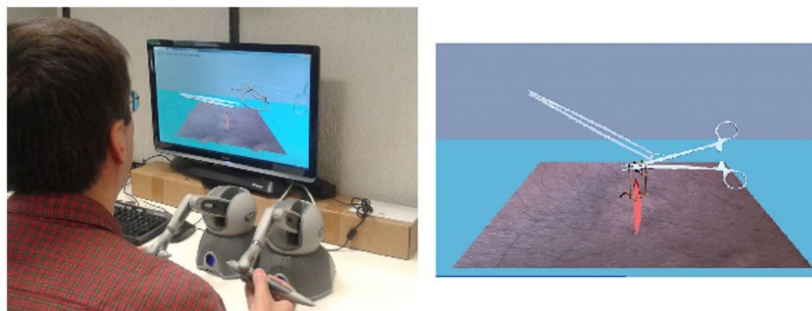


Figure 2-6 – Suturehap: stitching simulator

Palpation simulators: used to aid medical students in perfecting their touch sense to localize imperfections within the patient's body like tumors or hernias. As it happens with stitching simulators, palpation simulators use two haptic devices, one per hand. On the screen, the doctor guides her/his virtual hands and gets force feedback from the virtual body. The medical student must learn the different pressures he can exert to get an accurate analysis and avoid hurting the patient. An example of these simulators is the one presented by (Ullrich and Kuhlen [2012]) (Figure 2-7).

Dental procedures simulators: these simulators use a virtual 3D model of teeth a must offer deformation, sectioning, and roughing of it. The dentist can change between different tools depending

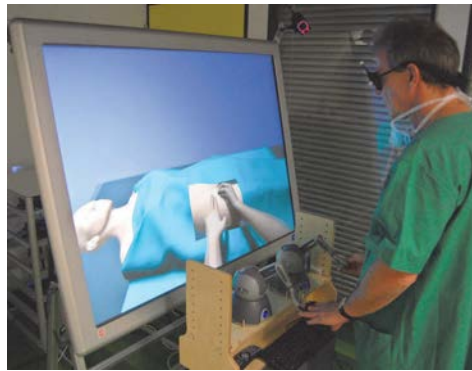


Figure 2-7 – Palpation simulator

on the procedure. The force feedback she/he gets from both devices, and the model must offer accurate deformation force feedback as the dentist must learn how to proceed without hurting the patient. One example of this is the one presented by (Tse et al. [2010]) which allows training caries removal, tooth restoration and dental drilling (Figure 2-8)



Figure 2-8 – hapTEL: Dental procedure simulator

Endoscopy simulators: used to teach medical students how to perform minimally invasive surgery like neurosurgery. These simulators teach medical students how to safely manipulate the instruments and how much force they must apply to the patients, as these are high-risk procedures for the patient. One example of these simulators is NeuroTouch [®] (Figure 2-9) which is a simulator presented by (Delorme et al. [2012])

Laparoscopy simulators: used to teach laparoscopy procedures to medical students. The simulator must be able to represent cutting forces, deformation, and sectioning. One example of this simulator is the Lapsim [®] (Figure 2-10) which is a commercial simulator introduced first by (Woodrum et al. [2006]). It uses a specifically made haptic devices to give direction and orientation to the user

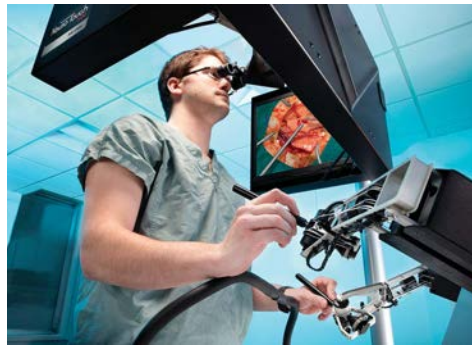


Figure 2-9 – NeuroTouch simulator: Endoscopy simulator

while manipulating the virtual tools.



Figure 2-10 – *Lapsim*®: Laparoscopy simulator

2.4 Simulator control architectures

Although some simulators use two haptic devices at the same time, the virtual tools are detached, which means that the motion of each device is individual and does not affect the others unless a collision occurs. This configuration does not allow guided training. To achieve it, two or more haptic devices must be interconnected using a communication layer and control laws that ensure the position and force tracking from the trainer device to the student. With the use of such control architectures, it is possible to implement a guided haptic simulator.

2.4.1 Multilateral configurations for control architectures

For a better understanding of the control architectures for haptic simulators, we need to study the more basic connection scheme: Teleoperation.

Teleoperation means operating a system remotely by interchanging data via another system (Cui et al. [2003]). The basic configuration of teleoperation allows a user to manipulate a device with the information being sent one-way forward. In this case, the user never gets any feedback from the environment. To solve this, it is necessary to implement a bilateral teleoperation scheme, where Master and Slave devices can perform position and force tracking (Niemeyer and Slotine [1991]).

For haptic simulators, the slave device, which can be real or virtual, represents the tool that the user will handle in a real task. In a basic haptic simulation scheme, there are only five elements: The user or human operator, the Master device, the slave device, the controller, and the remote environment (Secchi et al. [2007]). Figure 2-11 shows the haptic simulation scheme



Figure 2-11 – Haptic simulation architecture

where \dot{x}_m and \dot{x}_s are the velocities of the master and slave devices. F_h and F_e are the forces exerted by the user on her/his haptic interface by the environment on the slave device. Moreover, F_{c1} and F_{c2} are the control signals sent from the controller to the master and slave devices. In this scheme, the user inputs her/his motion and receives force feedback from her/his master device mimicking the forces applied by the environment on the slave. Using bilateral teleoperation schemes allows the development of different multilateral architectures for haptic simulators. Several devices, masters, and slaves can be added. These configurations are called multilateral configurations, and as (Shahbazi et al. [2013]) propose, they can be classified as:

Single Master with Single Slave (SM/SS): a single user manipulates a single slave.

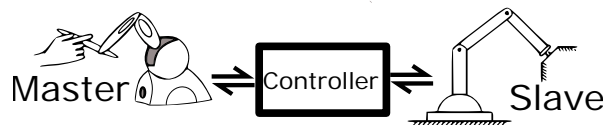


Figure 2-12 – SM/SS configuration

Single Master with Multi Slave (SM/MS): a single user operates several slaves. This architecture can be used for training or cooperation through multiple devices.

Multi Master with Single Slave (MM/SS): several users manipulate at the same time a slave device. This configuration can be used for guided training and cooperation tasks as well.

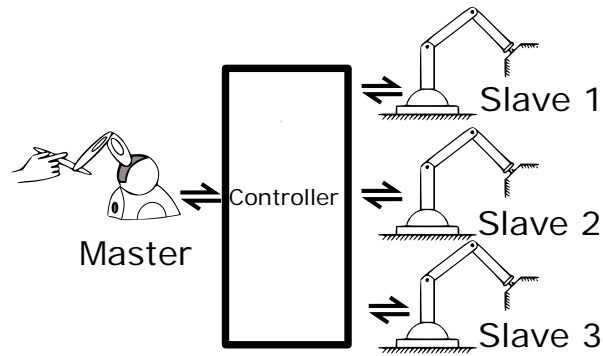


Figure 2-13 – SM/MS configuration

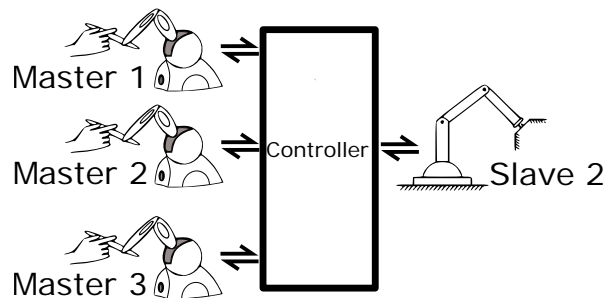


Figure 2-14 – MM/SS configuration

Multi Master with Multi Slave (MM/MS): several users manipulate several slaves. This configuration is made for cooperation tasks

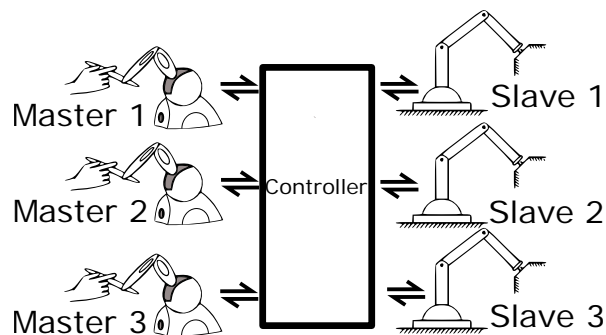


Figure 2-15 – MM/MS configuration

2.4.2 Dual user scheme

A typical approach to implement a guided training simulator is the use of the Dual-user scheme, which is a case of a Multi-Master/Single Slave configuration.

In this scheme, two users, manipulate a haptic device identified respectively Master 1 and Master 2. A Slave device interacts directly with the environment, which can be real or virtual, and is interconnected with each Master device (Figure 2-16).

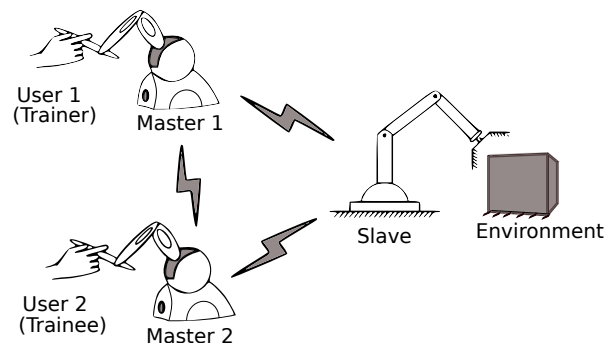


Figure 2-16 – Dual-user system

2.4.3 Dominance sharing in a training context

In the use of multilateral architecture, problems of dominance surge as the slave tries to follow all the master devices at the same time. Because of this, it is necessary to establish a mechanism that allows all the devices to share dominance over the slave.

(Khademian and Hashtrudi-Zaad [2007]) proposed the use of a variable named α for solving the dominance problems in the dual-user scheme. α establishes the dominance of each master device over the slave. In a training context, three possible cases are for setting the value of dominance (Figure 2-17).

Demonstration mode

In this mode, $\alpha = 1$ and the trainer has all the dominance over the slave, and the trainee only receives haptic feedback from the slave. The trainer can teach the trainee how to perform different gestures, and how to properly dose force without being disturbed by trainee's movements. The trainer can also be a virtual one, which in this case, can be a saved trajectory for the virtual tool on the Master device.

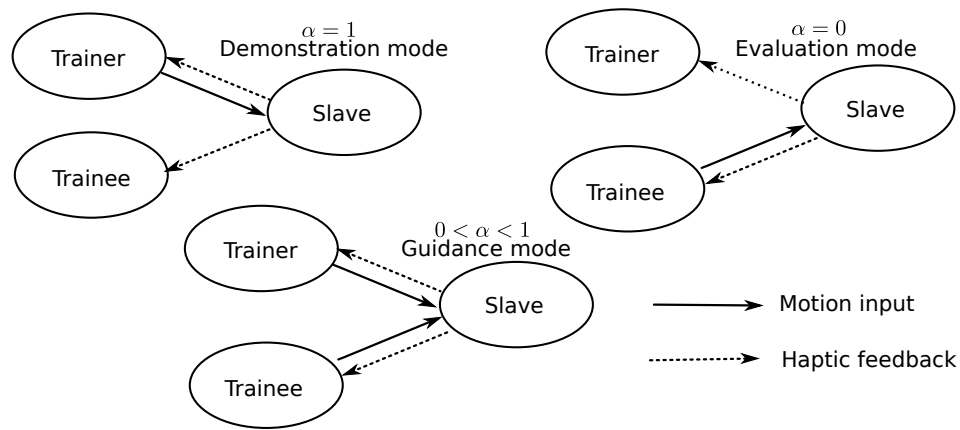


Figure 2-17 – Dominance sharing training cases

Guidance mode

In this mode, $0 < \alpha < 1$, trainer and trainee share the dominance of the system. Both master devices receive haptic feedback from the slave, and both devices input their relative dominance. How much dominance has each device is set by α . As $\alpha \rightarrow 1$ the dominance opts to the trainer. On the other hand, as $\alpha \rightarrow 0$, the dominance opts to the trainee. An example of this is when $\alpha = 0.3$, which means that the trainee has 70% of the dominance while the trainer has only 30% of it.

Evaluation mode

In this mode, symmetrical to demonstration mode, $\alpha = 0$, and the trainee now has all the dominance over the slave. The trainer only receives haptic feedback from the slave but cannot input his motion. This mode is used for the trainee to present its advances to the trainer. The trainee can perform the trajectory without any perturbation from the involuntary movements of the trainer.

2.4.4 Adaptive dominance

The trainer can change the value of α during the process. However, this process can be tedious as she/he needs to adjust the amount of dominance and concentrate on the task at the same time.

To make the change of dominance an easy task, an adaptive dominance can be used as well. The dominance is then, a function that changes the value of α from 0 to 1, indirectly depending on the trainee. If the trainee performs the task according to the trainer's movements, the function will grant him/her more dominance. On the contrary, if he is moving away from the trainer's position, the

function will return the dominance to the trainer so he can correct him in real-time.

A first adaptive dominance function was presented by (Shahbazi et al. [2013]) along with their dual-user architecture. Other authors as (Huang and Lu [2015]) and (Liu et al. [2015]) have presented different approaches for this function. Each one is based on the position error between the trainee and the trainer devices.

2.4.5 Bibliographical study of control architectures for haptic simulators

Several control architectures have been proposed for guided training simulators. As they are based on MM/SS and dual-user system, we present a study on the more relevant architectures found in the literature and provide which requirement(s) they do not fulfill for the desired simulator.

1. (Katsura et al. [2007]) proposed a multilateral force feedback control for cooperation tasks. Their solution is dedicated to systems that do not share the same kinematics (for instance, one translation for master 1, two translations for master 2 and 3 translations for the slave). The authors used an acceleration control with disturbance observers and spatial transformations to facilitate the cooperation between two master users. This solution is interesting, but our goal is that trainees reproduce the same gestures as the trainers (**Does not fulfill R5**).
2. (Moghimi et al. [2008]) developed an architecture for cooperation in virtual environments. This architecture works in a dual-user scheme and uses an adaptive controller with filters for the force and position. The architecture can work with impedance and admittance haptic devices. The authors prove their stability with a Lyapunov function and made experimental test but only with 2-DoF robots. The authors addressed that the architecture will be tested for training purposes. So, for a change of devices, it is necessary to recalculate the control law (**Does not fulfill R1 and R7**).
3. (Shahbazi et al. [2011]) presented an architecture for dual-user systems with different kinematics. They used a decentralized, shared adaptive controller to overcome the uncertainties on the robot's dynamics and to cope with the uncertain delays in the communication line. This solution works for cooperative use but is limited for training purposes as when the operator #1 has the lead on the slave, the controller does not ensure that the operator #2 gets the same force feedback (**Does not fulfill R1, R5, and R6**).

4. (Li et al. [2013]) presented a novel architecture that uses an adaptive fuzzy control for motion/force synchronization. The control was created to ensure stability regardless of the delays, the uncertainties of the dynamics/kinematics of the devices and the disturbances. In this architecture, the force and position tracking are two different objectives that are reached by one device at the time. This means that while one master devices perform only position tracking, and the second one only force tracking. This architecture is not usable for guided training, as both users do not share the same position and haptic feedback, which prevents that the trainer can correct the trainee in real-time (**Does not fulfill R1 and R6**).
5. (Khademian and Hashtrudi-Zaad [2013]) studied the transparency with two users and three degree-of-freedom on planar devices. They enhanced the control to guarantee the stability of this system against uncertainties in the environment and the user's dynamics, with the help of Llewellyn's unconditional stability criterion. An experimental user study was carried out to assess the effectiveness of the proposed architecture in terms of transparency in a training context. However, their studies with planar robots are not validated to work with devices where gravity affects directly the haptic devices. This could be solved by adding low-level gravity compensation control laws. However, the architecture interconnects directly both masters, without any treatment of dominance, invalidating the Demonstration and Evaluation modes (**Does not fulfill R1 and R2**).
6. (Shahbazi et al. [2013]) proposed a dual-user architecture in conjunction with the use of virtual fixtures to teach Minimal Invasive Surgery (MIS) to medical students. The authors utilize an impedance control, and the system uses an adaptive dominance factor law. An online evaluation of the trainee performance determines the dominance factor. However as the system uses virtual fixtures to create a path that the trainee must follow, this does not ensure force feedback from the slave to the masters, as the system focuses in path tracking rather than force tool feedback, which it is not desirable for our goal (**Does not fulfill R1 and R6**).
7. (Shahbazi et al. [2014]) introduced an architecture that works with fuzzy controls for training purposes. Authors made this architecture to work with four different levels of trainees: Beginner, Intermediate, Advanced, and Skilled. This architecture focuses on changing the feedback of the trainee based on her/his experience. The more skilled the student is, the more haptic feedback from the slave she/he receives instead of the trainer. As this architecture focuses on

- Demonstration mode, it does not allow the trainer to correct the trainee in real-time like Guidance or Adaptive mode, so it does not work for our objectives (**Does not fulfill R1 and R2**).
8. (Shahbazi et al. [2015]) also introduced a MM/SS teleoperated framework for cooperative task. The passivity of the system is investigated, taking into account time-varying communication delays (but with the assumption that they are all equal, which is not restricting as, in training applications, users are at the same site). A control is developed to provide haptic feedback to every user. Nevertheless, the experiments show that when a user (such as the trainer) has the full control on the slave, the other users are not guided by this user and they get haptic feedback whatever their device's position. This behavior may not be efficient for the training as the trainees will not be able to synchronize position and force in their gesture learning (**Does not fulfill R1, R5 and R6**).
 9. (Shamaei et al. [2015]) developed a trilateral architecture for training purposes. They propose two dominance factors as an extension of the one introduced by (Shahbazi et al. [2014]), but with fuzzy controllers (α and β , to establish Demonstration and Evaluation mode). The implementation of two dominance factor allow the system to separate and give priority of the force feedback from the environment and the trainer are applying to the trainee. The architecture requires six forces, and velocities as desired values. And, as it is remarked, the lack of the third mode, guidance or adaptative mode avoids that the architecture can be used for correct the trainee's movements in real-time (**Does not fulfill R1 and R2**).
 10. (Huang and Lu [2015]) introduced an architecture that splits the dominance factor into three different ones, to enable a shared control of the slave with one user protecting the gestures of the second one. They took the opportunity of these two new variables to provide some adaptive behavior to help users in their tasks. They provided experimental results including two different master devices (a Phantom Omni (3-DoF), and an Omega (2-DOF)). Nevertheless, this architecture carries out the drawbacks of Shahbazi's solution (lack of obvious and efficient force training capability) (**Does not fulfill R1, R2**).
 11. (Zakerimanesh et al. [2017])proposes an expansion to the dual-user scheme: a nonlinear Proportional plus Damping (nP+D) controller, previously proposed for SM/SS configurations. This control strategy considers the saturation of the motors and depends on two dominance factors. These dominance factors are α , which determines the supremacy of the trainer over the trainee,

and β that determines the supremacy of each user over the slave. Both dominance factors provide weight gains over the proportional-damping part of the control, which calculates the torque signal in function of the dominance, position error, and performance of the trainee. They proved the stability of this new architecture and simulated it with a 3-DoF planar robot, but still not with a robot affected by gravity (**Does not fulfill R1 and R7**).

12. (Panzirsch et al. [2017]) developed a dual user bilateral architecture for cooperative tasks, where two users must complete complicated labors using remote slave systems. The architecture proposed by the authors has the property to replicate the sensation of movement from each partner. As an example, the authors perform the task of loading a box. In a real-life situation, two humans can feel the movement of their respective partners when they are loading the box. Thanks to this, it is possible to synchronize the forces and motions without any other feedback more than the kinesthetic feeling of their body. This architecture tries to imitate this kinesthetic feeling using haptic devices and slaves in remote environments. However, the cooperation task is not well suited for the training approach studied in this thesis, as no leader gives the path to follow and the expected input force into the slave (**Does not fulfill R1, R2, R5, and R6**).

Table 2.1 sums up the characteristics of all the aforementioned architectures. The table is ordered by year and we use the authors' names as identification as some architectures do not have a proper name. The DoF column indicates the number of degrees of freedom that have been tested or proposed by the authors; in case the authors do not mention it, this is considered as an n-DoF architecture. We recall the important advantages and disadvantages of each architecture. The column "Training purpose" defines if the architecture has been used or can be used for training purposes compliant with our use case.

Table 2.1 – MM/SS architectures comparison table

No.	Authors	DoF	Advantages	Disadvantages	Training purpose
1	(Katsura et al. [2007])	n	Solution dedicated to systems that do not share the same kinematics at all	Trainees do not reproduce the same gestures as trainers	

Table 2.1 continued from previous page

No.	Authors	DoF	Advantages	Disadvantages	Training purpose
2	(Moghimi et al. [2008])	2	Works with admittance and impedance devices	Only 2 DoF Recalculation of controllers.	✓
3	(Shahbazi et al. [2011])	1	Works with delays. Shared adaptive control that overcomes uncertainties.	Controller does not ensure that trainee gets the same force feedback	
4	(Li et al. [2013])	n	The controls ensure stability regardless the delays, the uncertainties and the disturbances.	The force and position tracking are two different objectives replied by different devices	
5	(Khademian et al. [2013])	3	The controller guarantees the stability against uncertainties on the environment and the user's dynamics.	Only works with Planar systems	✓
6	(Shahbazi et al. [2013])	n	Online evaluation of the trainee performance	Focused on path tracking rather than force	

Table 2.1 continued from previous page

No.	Authors	DoF	Advantages	Disadvantages	Training purpose
7	(Shahbazi et al. [2014])	1	Changes the feedback of the trainee based on their experience	Focused in Demonstration mode	✓
8	(Shahbazi et al. [2015])	1	Works with constant delays	Trainees cannot synchronize position and force in their gesture learning	
9	(Shamaei et al. [2015])	1	Two dominance factors	Lacks guided mode	✓
10	(Huang and Lu [2015])	2	Solution dedicated to systems that do not share the same kinematics at all	Lacks force tracking	
11	(Zakerimanesh et al. [2017])	n	Calculates torque in function of the dominance, position error, and trainee's performances	Only tested with planar devices	
12	(Panzirsch et al. [2017])	n	Works with complicated tasks	Cooperation only	

Most of the architectures mentioned above work in a trilateral or dual-user scheme and have been tested for 1-DoF or more, but in planar robots, which does not fully illustrate the feasibility in 3D of haptic devices. Also, the use of trilateral architectures limits the requirement **R1**, due to the dif-

faculty of expanding the number of implied devices without a deeper redesign of the controllers and architecture structure.

Most of the architectures do not offer position and force tracking, or a multi-trainee approach, as they are designed to work only with one trainer and one trainee at the same time.

2.5 Conclusion

Haptic devices are used in computer-based simulators for medical training. They allow students to get kinesthesia training and aim at training on surgeries without touching a real patient while getting the same feeling.

The current commercial simulators offer only one trainee consoles. Although they accomplish their objective, they lack a console that allows a trainer to monitor and correct the trainee movements in real-time.

Researchers have proposed control architectures that allow a trainer and a trainee to interact directly during a training session. Some of these architectures allow position/force tracking and calculate the dominance of the slave device based on the trainee's performance. However, they have not been fully developed to work with n-DoF devices to allow several trainees to be instructed at the same time. With this in mind, we choose to expand the previously proposed ESC architecture into an n-DoF multi-trainee architecture.

Part III

Contributions

Chapter 3

Expanding to n -DoF (eESC architecture)

3.1 1-DoF ESC architecture

The ESC architecture embeds three Intrinsically Passive Controls (IPC) to provide compliance to the user haptic interfaces and to ensure that the interfaces with users (on masters' side) and the environment (on the slave's side) remain independently passive.

The ESC architecture utilizes an impedance-admittance controller configuration. This arbitrary choice of causality provides better transparency in practice, as explained in section 3.2.

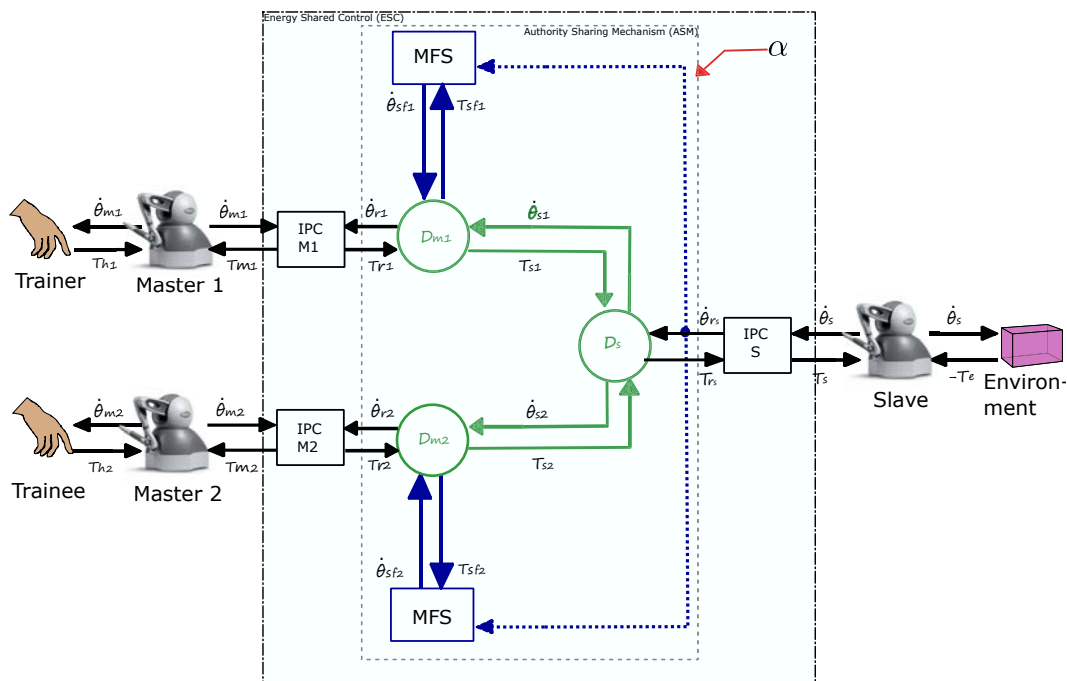


Figure 3-1 – ESC architecture dual-user haptic system for one dof

Design details are provided by Liu [2016]. Figure 3-1 depicts this architecture. The inputs/outputs of this architecture are Articular velocities ($\dot{\theta}_{m1}$ and $\dot{\theta}_{m2}$ for the master interfaces and $\dot{\theta}_s$ for the slave) and torques (T_{h1} and T_{h2} are applied by the users on the master haptic devices and T_e is the torque applied by the environment on the slave tool). D_{m1} , D_{m2} and D_s (in green) are lossless leveraging interconnections (Dirac structures) to perform authority sharing (modeled as skew-symmetric matrices in equations 3.1 and 3.2). They control the energy exchanged between both the masters and the slave devices according to the dominance α .

$$\begin{aligned}
 D_{m1} : \begin{bmatrix} \dot{\theta}_{r1} \\ T_{s1} \\ T_{sf1} \end{bmatrix} &= \begin{bmatrix} 0 & \alpha & 1-\alpha \\ -\alpha & 0 & 0 \\ \alpha-1 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_{r1} \\ \dot{\theta}_{s1} \\ \dot{\theta}_{sf1} \end{bmatrix} \\
 D_{m2} : \begin{bmatrix} \dot{\theta}_{r2} \\ T_{s2} \\ T_{sf2} \end{bmatrix} &= \begin{bmatrix} 0 & 1-\alpha & \alpha \\ \alpha-1 & 0 & 0 \\ -\alpha & 0 & 0 \end{bmatrix} \begin{bmatrix} T_{r2} \\ \dot{\theta}_{s2} \\ \dot{\theta}_{sf2} \end{bmatrix} \tag{3.1}
 \end{aligned}$$

$$\begin{aligned}
 D_s : \begin{bmatrix} \dot{\theta}_{s1} \\ \dot{\theta}_{s2} \\ T_{rs} \end{bmatrix} &= \begin{bmatrix} 0 & 0 & \beta_1 \\ 0 & 0 & 1-\beta_2 \\ \beta_1 & 1-\beta_2 & 0 \end{bmatrix} \begin{bmatrix} T_{s1} \\ T_{s2} \\ \dot{\theta}_{rs} \end{bmatrix} \\
 \beta_1 = \begin{cases} \alpha, & \alpha = 1, 0 \\ 1, & 0 < \alpha < 1 \end{cases} & \beta_2 = \begin{cases} \alpha, & \alpha = 1, 0 \\ 0, & 0 < \alpha < 1 \end{cases} \tag{3.2}
 \end{aligned}$$

The advantage of this approach is to provide a model linking passive blocks together in a passive manner, which results in a global passive system. The IPC controllers feature several parameters to tune their compliance to provide transparency without compromising the overall passivity. They do not introduce any steady-state error on (Articular velocities and torques), Each master IPC provides the interconnection part with an "estimation" of the torque provided by the corresponding master device. This torque reflects, in steady-state, the user's hand interaction force, and gravity, friction, and inertia effects. The slave IPC provides an estimation of the position of the slave. During manipulations, in steady-state, the input torque T_{rs} indirectly reflects the interaction force between the slave and the environment, added with gravity, friction and inertia effects. This architecture does not rely on a force sensor to measure the forces; the only forces used are the ones calculated by the inner model of each

IPC. This architecture was designed with three haptic devices that were kinematically identical and with very close motions. In practice, the disturbances are nearly the same for the three of them, so that, they offset one another and the leading user only feels the tool-environment interaction forces ($T_{r1} \approx T_{rs} \Rightarrow T_{h1} \approx T_e$ in practice). To get the real force feedback, it is possible to install a costly force sensor on each device effector or provide a force estimator and update the IPC model to guarantee its passivity.

By default, the $D_{m1,m2,s}$ interconnection has an important limitation in demonstration ($\alpha = 1$) and evaluation ($\alpha = 0$) modes: the following user does not get any feedback from the slave, for energy balancing reasons. A solution, proposed by Liu et al. [2015], consists of adding two Modulated Flow Sources (MFS) providing the necessary complementary energy to the following device. To ensure the global passivity preservation that these additional sources of energy could compromise, a Time Domain Passivity Observer (TDPO) measures the energy $E_p(t)$ injected by the two MFS into the system:

$$E_p(t) = \int_0^t [(\alpha - 1)T_{r1}(\tau) - \alpha T_{r2}(\tau)]^T \dot{\theta}_{rs}(\tau) d\tau \quad (3.3)$$

A solution to preserve the system passivity (and therefore stability) consists in ensuring that $E_p(t) \leq 0$. The outputs of the MFS are accordingly modulated (3.4), acting as a Switched Time Domain Passivity Controller:

$$\dot{\theta}_{sf1}(t) = \dot{\theta}_{sf2}(t) = \begin{cases} \dot{\theta}_{rs}(t), & \text{if } E_p(t) \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

In practice, with a standard passive environment (free or static "walls"), no passivity loss was observed so far by the authors.

3.2 The Intrinsically Passive Control

To better understand how the ESC architecture achieves the position and force tracking, we need to introduce the Intrinsically Passive Controller (IPC) propose in (Stramigioli [1996]). This control uses the concept of passivity to bring global stability to the system (more details in Appendix IV-C).

There exist two configurations for this controller: with impedance and admittance causality. Figure 3-2 shows the concept of an Impedance IPC, while Figure 3-3 shows the Admittance IPC.

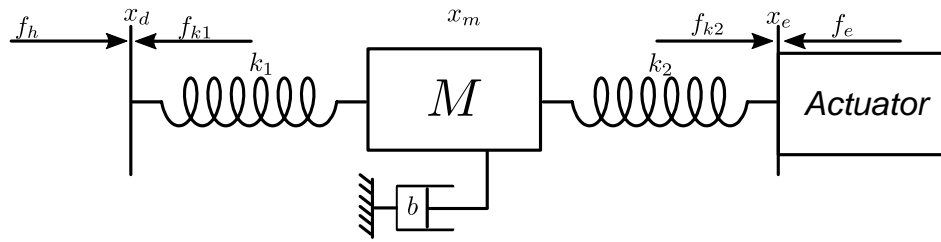


Figure 3-2 – Impedance IPC controller for the master

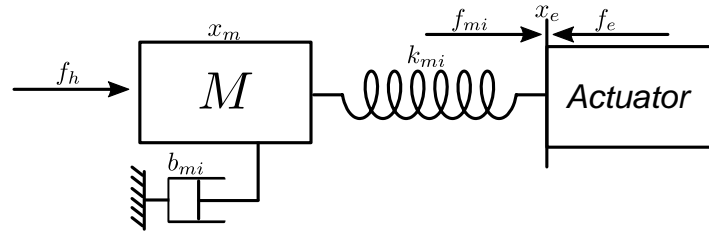


Figure 3-3 – Admittance IPC controller for the slave

Both configurations simulate the dynamics of a mass-spring-damper system. The core of the control is the inner mass M , whose position is denoted by x_m . The springs, k_{mc} , and k_{mi} offer proportional control, the damper b_{mi} ensures the passivity of the system by consuming the energy generated by the momentum of the mass M (Arcara and Melchiorri [2002]). x_d is the desired position reference, and x_e is the position of the environment. f_h is the input force of the user, and f_{mc} is the reaction force used to give the haptic feedback to the user, and f_{mi} is the force exerted by the control. Therefore $f_h = -f_{mc}$ and $f_e = -f_{mi}$ in steady-state. The values of the springs and dampers are constant, so they would not change during computing. The model that rules the dynamics of the Impedance IPC is

$$\ddot{x}_{mc} = \frac{f_{mc} - f_{mi} - \dot{x}_m b_{mi}}{M} \quad (3.5)$$

And for admittance

$$\ddot{x}_{mc} = \frac{f_h - f_{mi} - \dot{x}_m b_{mi}}{M} \quad (3.6)$$

The dynamic of the force exerted by f_{mc} and f_{mi} are based in Hooke law and shown in equation 3.7

$$\begin{aligned} f_{mc} &= k_{mc}(x_m - x_d) \\ f_{mi} &= k_{mi}(x_d - x_e) \end{aligned} \quad (3.7)$$

The force f_{mc} is returned to the Master device. The force f_{mi} is the commanded force and applied in the slave side in order. In the case of the impedance IPC, in a teleoperation scheme where position/force tracking is the objective to achieve, $k_{mc} = k_{mi}$. In case we do not respect this equality restriction, the controller would compute the desired force with a scale factor of

$$n_f = \frac{k_{mc}}{k_{mi}} \quad (3.8)$$

This is not desirable for our case, as n_f must be equal to 1 to preserve the transparency of the system by exerting the same force from the environment onto the master device.

3.2.1 Impedance-Admittance IPC

By using two IPC controllers, it is possible to achieve a bilateral teleoperation system. It exists four configurations for this: Impedance-Impedance, Admittance-Admittance, Impedance-Admittance, and Admittance-Impedance (Hashtrudi-Zaad and Salcudean [2001]). The Impedance-Admittance IPC is the one used in the ESC architecture and is the union of both controllers presented in Figure 3-2 and 3-3. Its diagram is shown in Figure 3-4.

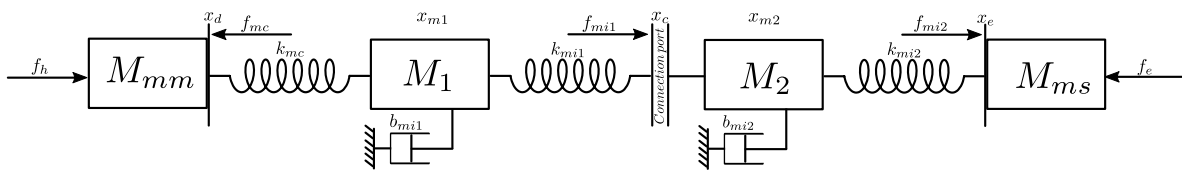


Figure 3-4 – Impedance-Admittance IPC for ESC

Where M_{mm} and M_{ms} are the masses of the master and slave device respectively. The model of the

control is set as

$$\begin{aligned}
 M_{mm}\ddot{x}_u &= f_u - k_{mc}(x_u - x_{m1}) \\
 M_1\ddot{x}_{m1} &= -k_{mc}(x_{m1} - x_u) - k_{mi1}(x_{m1} - x_{m2}) - b_{mi1}\dot{x}_{m1} \\
 M_2\ddot{x}_{m2} &= -k_{mi1}(x_{m2} - x_{m1}) - k_{mi2}(x_{m2} - x_e) - b_{mi2}\dot{x}_{m2} \\
 M_{ms}\ddot{x}_e &= -k_{mi2}(x_e - x_{m2}) - f_e
 \end{aligned} \tag{3.9}$$

As the impedance control reads the position of the device and the admittance of the force/torque, both signals are incompatible. Because of this, position x_e , from the impedance control, changes to x_c , and it is now considered as the connection port. With this, the force f_{mi} from the impedance IPC is equal to the input force f_u from the admittance IPC, allowing to preserve the transparency and the interchange of energy. This point can also be used to implement a communication layer for remote teleoperation as well or application of the dominance.

3.2.2 Transparency and stability of the IPC

Transparency

The objective of any bilateral teleoperation is to give the user the sensation of interacting directly with the final environment, regardless of the number of devices and communication layers implied. This important characteristic is named transparency and specifies the fidelity of the haptic feedback that the user gets from the teleoperator.

For impedance devices, between the interaction of the user, haptic devices, and environment, it exists a mechanical impedance Z that relates the force F and velocity $X \cdot s$ of the system (Raju et al. [1989]).

$$Z = \frac{F}{X \cdot s} \tag{3.10}$$

The user moves her/his master device, which commands the slave device, and this last interacts with the environment. With this in mind, we defined a teleoperator as a black box that consists of the master-controller-slave system described in Figure 3-5.

So the impedance coupling of the user with the environment can be described as a hybrid matrix

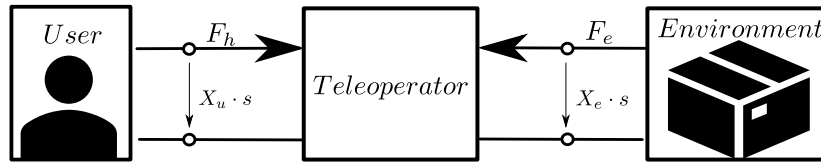


Figure 3-5 – Teleoperator scheme

in eq. (3.11), which considers the velocity and force of both parts.

$$\begin{pmatrix} F_h \\ -X_e \cdot s \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} \cdot \begin{pmatrix} X_u \cdot s \\ F_e \end{pmatrix} \quad (3.11)$$

Where h_{11} is the unconstrained movement impedance and gives the equivalent of inertia and damping that the user feels when the system is in free motion. h_{12} is the backward force gain and the force tracking in contact. h_{21} is the forward velocity gain and the position tracking during free motion. And h_{22} is the slave output admittance and the position tracking during contact (Aliaga et al. [2004]).

The velocity of the environment is always negative as must enter into a port according to the two-port convention. A teleoperator that offers full transparency must fulfill the conditions of (3.12)

$$F_h = F_e \quad \text{and} \quad X_d \cdot s = X_e \cdot s \quad \therefore Z_u = Z_e \quad (3.12)$$

So the total force applied by the user is:

$$F_h = h_{11} \cdot X_h \cdot s + h_{12} F_e \quad (3.13)$$

And the velocity of the slave is:

$$-X_e \cdot s = h_{21} \cdot X_u \cdot s + h_{22} f_e \quad (3.14)$$

Analyzing (3.11) and (3.12), the perfect transparency, without scaling forces or positions, can be

deduced as:

$$\begin{pmatrix} F_h \\ -X_e \cdot s \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} X_h \cdot s \\ F_e \end{pmatrix} \quad (3.15)$$

IPC's transparency

To determine the ESC's IPC transparency, we first determined its hybrid matrix. Based on the framework presented by Hannaford [1989], we use the declared model in (3.9) from the Impedance-Admittance IPC and apply the Laplace transform with initial conditions equal to zero.

$$\begin{aligned} M_{mm}X_u \cdot s^2 &= F_h - k_{mc}(X_u - X_{m1}) \\ M_1X_{m1} \cdot s^2 &= -k_{mc}(X_{m1} - X_u) - k_{mi1}(X_{m1} - X_{m2}) - B_{mi1}X_{m1} \cdot s \\ M_2X_{m2} \cdot s^2 &= -k_{mi1}(X_{m2} - X_{m1}) - k_{mi2}(X_{m2} - X_e) - B_{mi2}X_{m2} \cdot s \\ M_{ms}X_e \cdot s^2 &= -k_{mi2}(X_e - X_{m2}) - F_e \end{aligned} \quad (3.16)$$

From this, we declare the mechanical impedances (Z) of the system and its gains (C), through the connection channels between master and slave device as

$$\begin{aligned} Z_1 &= M_{mm} \cdot s & Z_2 &= M_1 \cdot s & Z_3 &= M_2 \cdot s \\ Z_4 &= M_{ms} \cdot s & C_1 &= \frac{k_{mc}}{s} & C_2 &= \frac{k_{mc} + k_{mi1} + B_{mi1} \cdot s}{s} \\ C_3 &= \frac{k_{mi1}}{s} & C_4 &= \frac{k_{mi1} + k_{mi2} + B_{mi2} \cdot s}{s} & C_5 &= \frac{k_{mi2}}{s} \end{aligned}$$

So the model of (3.16) becomes

$$X_h Z_1 \cdot s = F_h - C_1 X_{mm} \cdot s + C_1 X_1 \cdot s \quad (3.17)$$

$$X_1 Z_2 \cdot s = C_1 X_{mm} \cdot s - C_2 X_1 \cdot s + C_3 X_2 \cdot s \quad (3.18)$$

$$X_2 Z_3 \cdot s = C_3 X_1 \cdot s - C_4 X_2 \cdot s + C_5 X_{ms} \cdot s \quad (3.19)$$

$$X_e Z_4 \cdot S = C_5 X_2 \cdot s - C_5 X_{ms} \cdot s - F_e \quad (3.20)$$

And the new diagram that describes the two-channel connection between the master and the slave device is shown in Figure 3-6, where the $IPC1$ is the Impedance IPC used by the Master 1 or Master 2, and the $IPC2$ is the Admittance IPC used by the slave device.

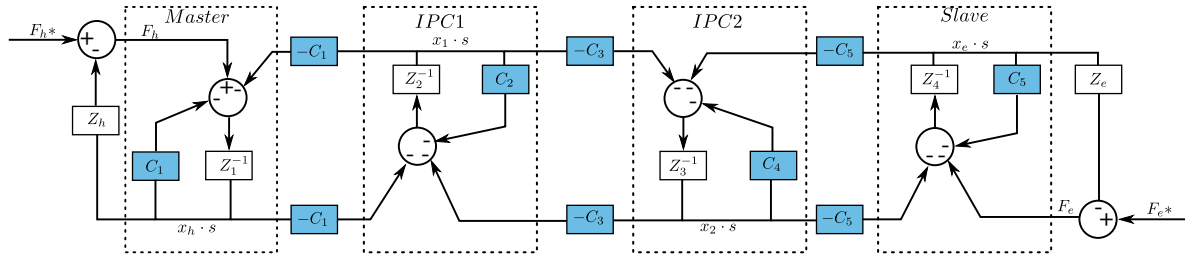


Figure 3-6 – ESC's IPC transparency diagram

Where F_h^* and F_e^* are the exogenous inputs from the user and the environment. Meanwhile Z_h and Z_e denote the impedance of the user's hand and the environment. Once the equations are set, we need to determine F_h and x_e , both in function of x_h and F_e . For F_h , we first isolate (3.19) and (3.20) for \dot{x}_{ms} , matching them we get

$$\dot{x}_2 = \frac{-C_3 Z_{t1} \dot{x}_1 + C_5 F_e}{-Z_3 Z_{t1} - C_4 Z_{t1} + C_5^2} \quad (3.21)$$

With $Z_{t1} = Z_4 + C_5$. Substituting (3.21) into (3.18) we get

$$\dot{x}_1 = \frac{C_1 Z_{t2} \dot{x}_{mm} + C_3 C_5 F_e}{Z_{t3}} \quad (3.22)$$

With $Z_{t2} = -Z_3 Z_{t1} - C_4 Z_{t1} + C_5^2$ and $Z_{t3} = Z_2 Z_{t2} + C_3^2 Z_{t1} + C_2 Z_{t2}$. Now, substituting (3.22) into (3.17) and isolating for F_h we get

$$F_h = \frac{(C_1^2 Z_{t2} - Z_1 Z_{t3} - C_1 Z_{t3}) \dot{x}_{mm} + \frac{C_1 C_3 C_5}{Z_{t3}} F_e}{Z_{t3}} \quad (3.23)$$

Now we isolate (3.18) and (3.19) for \dot{x}_1 , matching them and clearing for \dot{x}_2 we get

$$\dot{x}_2 = \frac{C_1 C_3 \dot{x}_{mm} + C_5 Z_{t4} \dot{x}_{ms}}{Z_3 Z_{t4} + C_4 Z_{t4} - C_3^2} \quad (3.24)$$

With $Z_{t4} = Z_2 + C_2$. Substituting (3.24) into (3.20) and isolating for \dot{x}_{ms} we get

$$\dot{x}_{ms} = \frac{-C_1 C_3 C_5}{C_5^2 Z_{t4} - Z_4 Z_{t5} - C_5 Z_{t5}} \dot{x}_{mm} + \frac{Z_{t5}}{C_5^2 Z_{t4} - Z_4 Z_{t5} - C_5 Z_{t5}} F_e \quad (3.25)$$

With $Z_{t5} = Z_3 Z_{t4} + C_4 Z_{t4} - C_3^2$. So with (3.23) and (3.25), we determine the hybrid matrix as

$$\begin{aligned} h_{11} &= \frac{(C_1^2 Z_{t2} - Z_1 Z_{t3} - C_1 Z_{t3})}{Z_{t3}} & h_{12} &= \frac{C_1 C_3 C_5}{Z_{t3}} \\ h_{21} &= \frac{-C_1 C_3 C_5}{C_5^2 Z_{t4} - Z_4 Z_{t5} - C_5 Z_{t5}} & h_{22} &= \frac{Z_{t5}}{C_5^2 Z_{t4} - Z_4 Z_{t5} - C_5 Z_{t5}} \end{aligned} \quad (3.26)$$

For a complete expansion of the hybrid matrix, refer to Appendix B. The transparency of the system depends on the parameters of the IPC and by simple inspection, it is possible to determine that the system does not meet the condition established in (3.15) for full transparency. However, it is possible to determine if the architecture is transparent for different inputs in low frequency (LF) and high frequency (HF). We evaluate $s \rightarrow 0$ for LF and $s \rightarrow \infty$ for HF. The results of the evaluation are visible in Table 3.1.

Parameter	LF ($s \rightarrow 0$)	HF ($s \rightarrow \infty$)
h_{11}	$B_{mi1} + B_{mi2}$	M_{mms}
h_{12}	1	$1/s^6 \rightarrow 0$
h_{21}	-1	$-1/s^6 \rightarrow 0$
h_{22}	0	$M_{ms}/s^6 \rightarrow 0$

Table 3.1 – Analysis of the hybrid matrix in Low and High Frequency

From the analysis in LF, we can observe that h_{12} , h_{21} and h_{22} behave ideally respecting the full transparency condition given by (3.15). This means that the position and force tracking in free motion and contact are met without any added force. In the case of h_{11} , due to the non-zero value of the dampers is not possible that $h_{11} \rightarrow 0$, which means that the user always feels the damping of the architecture even in very smooth movements. As the damping factors cannot have negative values, one clear solution is to set $B_{mi1} = B_{mi2} = 0$, but it is important to remark that with $B_{mi1}, B_{mi2} \rightarrow 0$, the passivity of the system becomes compromised, and therefore the stability. From this, the damping factors should mitigate the input \dot{x}_u in low frequencies. To get a reduction, we know that the gain must be below zero decibels (dB). So we consider that $B_{mi1} + B_{mi2} < 1$ (as $20 \log(1) = 0$), so the input speed of the \dot{x}_u of the user is mitigated, allowing to approach the system as close as possible to the desired full transparency.

In HF, h_{11} indicates that the user will always feel the inertia of the haptic device in free motion. $h_{12}, h_{21} \rightarrow 0$, so the added force for force tracking during contact and position tracking during free motion is mitigated, which is desirable. h_{22} indicates that the position tracking during contact adds the inertia of the slave device.

To illustrate these observations, we drew the Bode diagram (Figure 3-7) for five possible cases of damping: $B_{mi1} = B_{mi2} = 0.01$, $B_{mi1} = B_{mi2} = 0.1$, $B_{mi1} = 0.3$ and $B_{mi2} = 0.4$, $B_{mi1} = B_{mi2} = 0.49$ and $B_{mi1} = B_{mi2} = 1$. For all the different values of B_{mi1}, B_{mi2} we arbitrary consider that $k_{mc} = k_{mi1} = k_{mi2} = 100$, $M_1 = M_2 = 0.01$ and $M_{mm} = M_{ms} = 0.1$ as the main subject of interest is to demonstrate that the value of the dampers affects the transparency of the system.

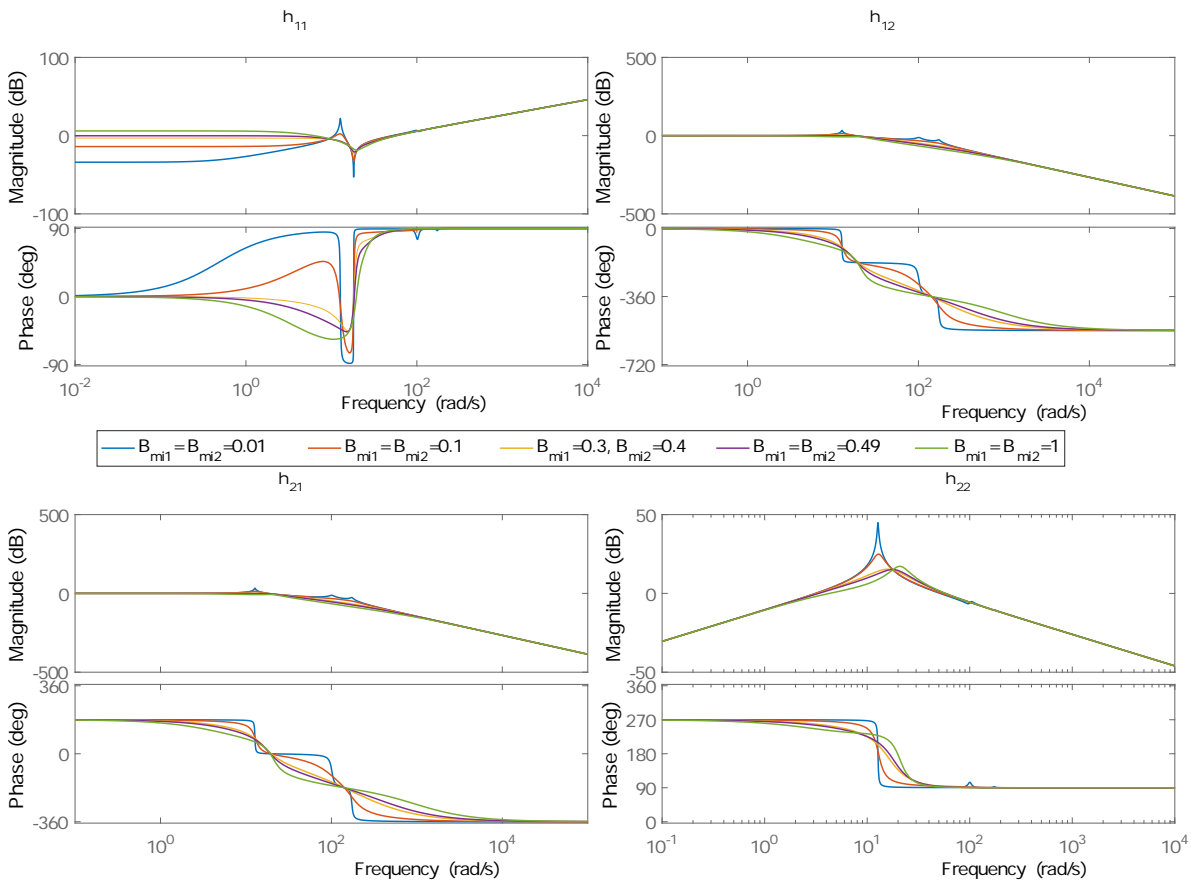


Figure 3-7 – Bode diagram for the ESC hybrid matrix with different dampers

Figure 3-7 shown the following behavior for each member of the hybrid matrix

h_{11} : For $B_{mi1} + B_{mi2} \leq 1$, the inertia felt by the user is mitigated in frequencies lower than 100rad/s, which is desirable for a transparent system. After this, it exists a change in the phase of the output force, which is more abrupt for lower damping factors. From this point, the inertia is

amplified in higher frequencies, and the values of the dampers are irrelevant, as it is shown how the output has a slope of gains independently of the parameters used. This indicates that the user feels the inertia of the mass, augmented by the speed she/he applies to the tool.

h_{12} : For frequencies lower than 100rad/s there is no gain or nor mitigation of the force tracking. This indicates that the output that the user feels is the same that the one given by the environment. After this point, force tracking is mitigated, which is translated to the user as a reduction in the force felt from the environment.

h_{21} : This one as a similar behavior as h_{12} , explained by the fact that their values are the same with just a change of sign. This indicates that the position tracking in free motion is achieved in frequencies lower than 100rad/s. After this point, the position tracking is mitigated if the user applies inputs with higher frequency.

h_{22} : The position tracking during contact is the only element of the hybrid matrix that behaves as desired with low and high frequencies. The position tracking is only amplified around 100rad/s, which indicates the beginning of amplification for h_{11} , reduction for h_{12} and h_{21} and the loss of transparency from the environment.

The final assumption from this analysis is that ESC is transparent in lower frequencies, and at higher frequencies (around 100rad/s for the test parameters) the transparency is lost. This does not represent a problem, as for guided-training purposes, the trainer does not effectuate sudden or quick movements while the trainee is learning how to do the task.

Finally, we tested ESC vs two other architectures to see how well the performance of this was. The two architectures selected are Position-Error-Based (PEB) and Direct-Force-Reflection (DFR) from Tavakoli et al. [2008]. PEB was chosen for its similitude to ESC of being a two-channel architecture and both only read the position of the user. DFR was chosen for being a simpler architecture and incorporate a force sensor, which could improve transparency. The hybrid matrix of PEB is given by

$$H = \begin{bmatrix} Z_1 + C_m \frac{Z_A}{Z_{ts}} & \frac{C_m}{Z_{ts}} \\ -\frac{C_s}{Z_{ts}} & \frac{1}{Z_{ts}} \end{bmatrix} \quad (3.27)$$

Where $C_m = (k_{mc} + B_{mi1}s)/s$, $C_s = (k_{mi2} + B_{mi2}s)/s$, $Z_{tm} = Z_m + C_m$ and $Z_{ts} = Z_s + C_s$. For DFR,

the hybrid matrix is set as

$$H = \begin{bmatrix} Z_m & 1 \\ -\frac{C_s}{Z_{rs}} & \frac{1}{Z_{rs}} \end{bmatrix} \quad (3.28)$$

The parameters for three architectures are same as the one use for the Bode diagram of Figure 3-7, with $B_{mi1} = B_{mi2} = 0.49$. Figure 3-8 show the comparison of the three architectures. From the

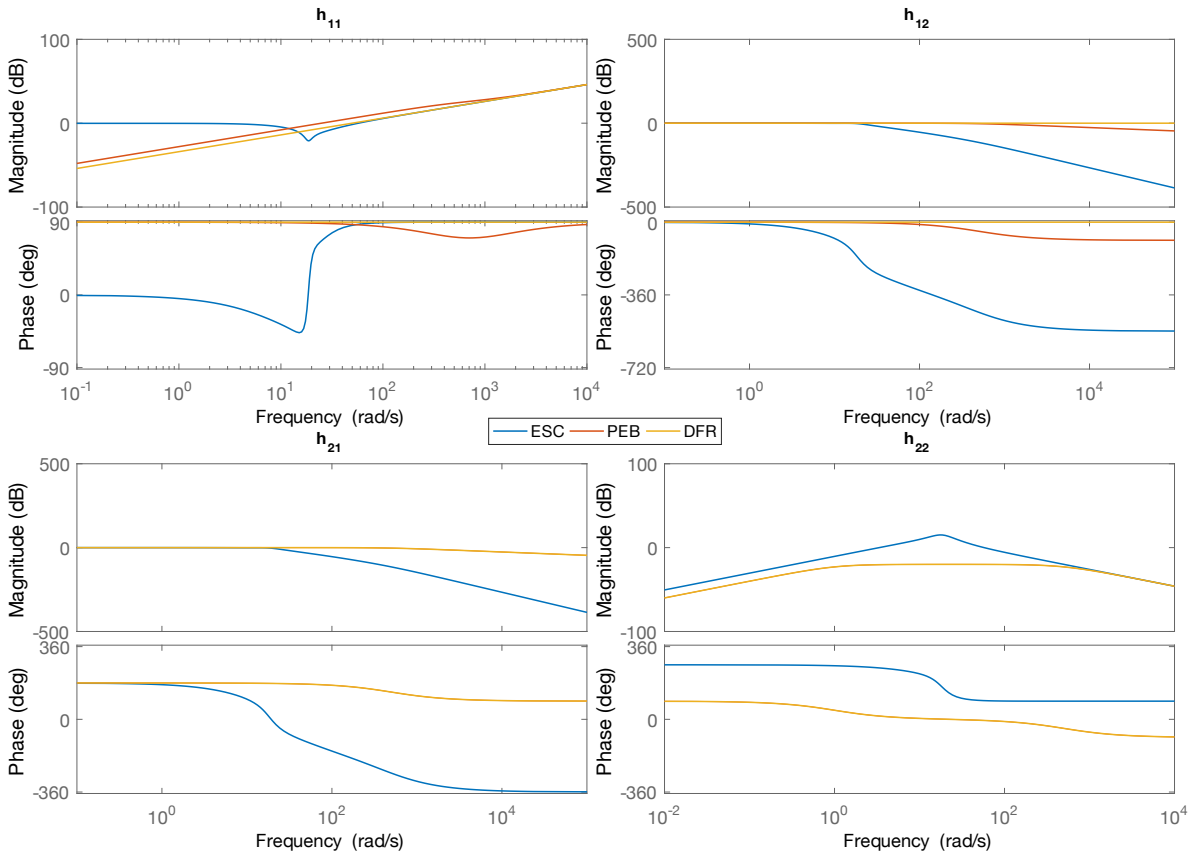


Figure 3-8 – Comparison between different architectures

comparison we deduce the following:

h_{11} : PEB and DFR mitigate the inertia felt by the user better than ESC, with a similar behavior between the first two. However, in higher frequencies, the amplification of the inertia is the same for three architectures.

h_{12} : DFR is the best in force tracking, as the frequency of the input does not change the value of the force exerted by the environment. This is because DFR has a force sensor attached. Between PEB and ESC, this latter behaves worst, as it mitigates the force tracking sooner as PEB.

h_{21} : Once again ESC mitigates the position tracking in free motion before PEB and DFR. Between these last two architectures, the behavior is similar and there is no noticeable difference between them.

h_{22} : For position tracking during contact, PEB and DFR do not amplify the signal, while ESC only around 100rad/s as it is explained in the previous analysis.

As the results show, ESC has the worst transparency compared with PEB and DFR. The addition of a force sensor can increase the transparency of the system; however, the majority of the haptic devices lack one. A force sensor can be placed at the end of the tool of the haptic devices, but it must be instrumented and adapted to the simulation software, which can increase the cost of the simulator. Once ESC is demonstrated to be transparent (in low frequencies), we proceed to explain the process to expand it into n -Dof.

3.3 From 1-DoF to n -DoF

3.3.1 Problem

To create a functional simulator, the aforementioned 1-DoF ESC architecture must be expanded to n -DoF. The principal problem with the expansion of the system is to ensure its stability all the time. The architecture has been tested in 1-DoF, in a planar configuration without gravity. The IPC controller must be able to maintain the system's stability, regardless of the position of the links. The two principal problems with the expansion of ESC to n -Dof is the non-linearity that occur because of the addition of more degrees of freedom. Also, the gravity term complicates the model, as a continuous force/torque must be applied on each link moving direction in the vertical direction. Another difficulty of this expansion is to find the correct parameters for the IPC controllers that can ensure an accurate position/force tracking avoiding side-effects like shattering and loss of transparency. The only method proposed to tune the IPC parameters was by trial and error.

3.3.2 Assumptions and proposal

IPC has been demonstrated working with Multi-Input Multi-Output systems. Arcara et al. [2001] presents simulation results of a planar robot using IPC in parallel for control. In their experimentation, a 3-Dof robot is used for teleoperation. Their results have shown that interconnecting several IPC controllers in parallel Cartesian position and orientation purposes, bilateral teleoperation is reachable.

Based on this, it is assumed that the compliance of the IPC controller absorbs the disturbances (nonlinearities such as frictions, Coriolis forces), and also that every device has the same kinematics and very close so positions and orientations that the gravity applied on each device is the same. By expanding the ESC architecture independently for each joint (Figure 3-9 for articular and Figure 3-10) a n -DoF dual user scheme, named (eESC) can be achieved. It permits connecting devices with the same kinematics. In a second approach, we duplicated the ESC for each Cartesian axis (positions and rotations).

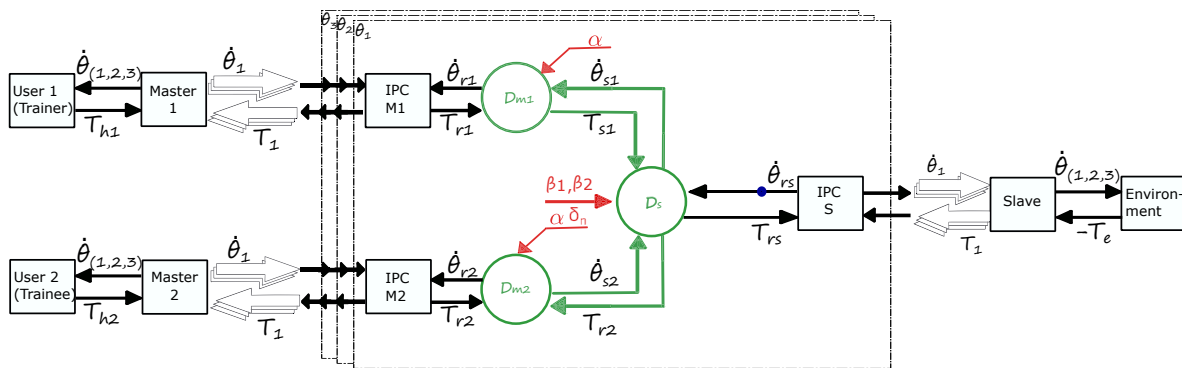


Figure 3-9 – Multiple DoF Articular eESC architecture

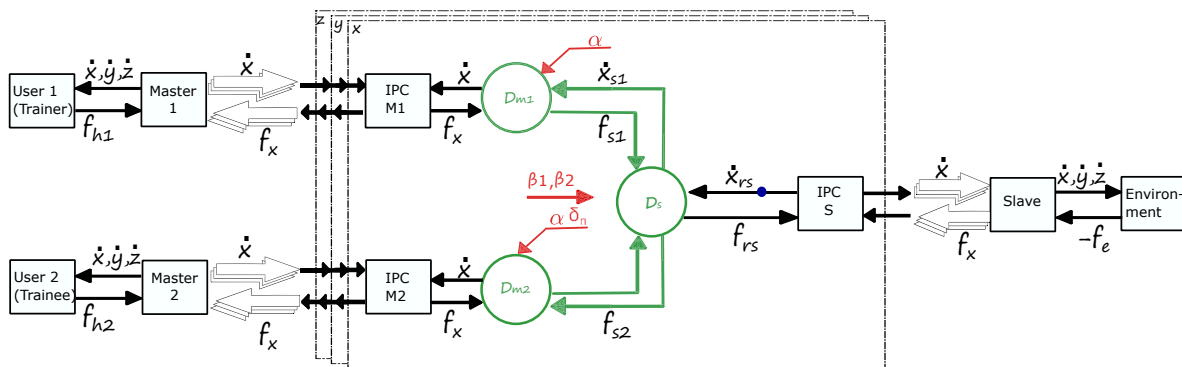


Figure 3-10 – Multiple DoF Cartesian eESC architecture

As the ESC architecture is based on the TDPC approach (Franken et al. [2011]), the passivity of every link is achieved independently. Therefore every source of energy inside each device is passivated. For each degree of freedom, the dynamic effects produced by the motions, are seen as external input torques and transmitted as is to the communication side (Authority Sharing Mechanism - ASM). So, the IPC controls must be tuned to get a desirable behavior. This expanded ESC architecture is named **eESC**. To determine the eESC reliability and precision, a series of experimental tests were performed.

3.3.3 Simulation

We first simulated the system by expanding the architecture into three DoF: θ_1, θ_2 and θ_3 for Articular (aESC), and x, y and z for Cartesian (cESC) expansion. The main advantage that Cartesian expansion offers versus the Articular one is the ability to work with devices that do not share the same kinematic configuration. This means that a haptic master device designed for human interaction can be used with a robotic manipulator on the slave side, as long as both systems have a common Cartesian workspace.

Simulation setup

The models used for Master 2 and the Slave are based on the dynamic model of the Geomagic 3D Touch (Appendix A.1). The model only considers 3 actuated DoF, which implies controlling only of the translations of the devices, so not orientations are considered with this setup. In this case, the system is set in Demonstration mode, so Master 1 is the leader device, which inputs the desired position, and therefore, it is represented by a trajectory generator, as no dynamic model is required for this. The simulation is timed from 0s to 10s and the trajectory is set as a circular path with a radius $r = 10\text{cm}$ defined by

$$\begin{aligned} x &= r * \sin(t) \\ y &= r * \sin(t) \\ z &= r * \cos(t) \end{aligned} \tag{3.29}$$

For the Articular approach, we used the inverse kinematics of the Geomagic Touch 3D (Appendix A.1.2) to establish the desired Articular positions and the direct kinematics transform (Appendix A.1.1) to compare the performed path between both approaches (Articular vs Cartesian). Figure 3-11 shows the diagram of the simulation.

The Cartesian expansion requires higher gains than the Articular expansion to get a good performance. We opted to use the same numerical value for the gains on both devices. The values of the virtual spring stiffness and masses were set empirically as there is no method to obtain them. In the case of the damper determination, as the IPC resembles a mass-spring-damper system, we assume that it is desirable to have it as a critically damped system, as an underdamped setting involves free vibrations of the mass (which leads into chattering), and an overdamped setting leads to slow reactions (Ogata [2003]).

To determine the damping factor of the master IPCs, first, we needed to determine the total stiffness

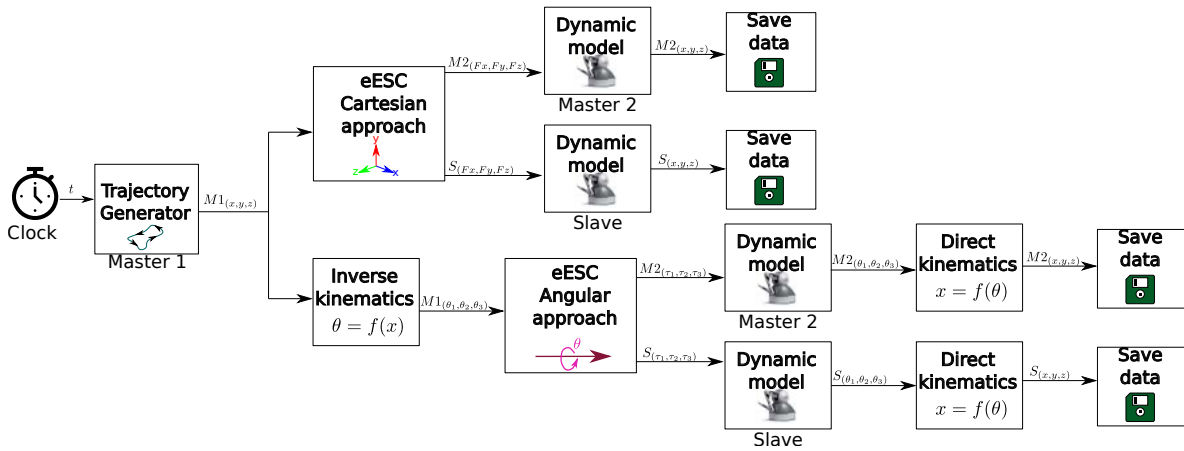


Figure 3-11 – Simulation diagram: Cartesian vs Articular approaches

of the system. The master IPCs have more than one spring, so their global stiffness is

$$k_t = \frac{1}{\frac{1}{k_1} + \frac{1}{k_2} + \dots + \frac{1}{k_n}} \quad (3.30)$$

as $k_{mc} = k_{mi}$ to provide bilateral transparency, the total stiffness is

$$k_t = \frac{k_{mc}}{2} = \frac{k_{mi}}{2} \quad (3.31)$$

In the case of the slave IPC, it only exists one spring, therefore $k_t = k_{mi}$. Then the damping coefficient for a critically damped system is given by

$$B_{mi} = 2\sqrt{M * k_t} \quad (3.32)$$

With this, Table 3.2 shows the IPC parameters obtained employing simulation. These gains have been obtained empirically through several iterations to get perfect tracking (negligible error) in both approaches.

Table 3.2 – IPCs' parameters for simulation validation: Cartesian vs Articular

Impedance (Master IPC)				Admittance (Slave IPC)			
Parameter	x	y	z	Parameter	x	y	z
	θ_1	θ_2	θ_3		θ_1	θ_2	θ_3
$M_1 kg$	0.01	0.01	0.01	M_2	0.01	0.01	0.01
$k_{mc}(N/m)$	2000	2000	2000	k_{mi2}	2000	2000	2000
$k_{mi1}(N/m)$	2000	2000	200	B_{mi2}	8.94	8.94	8.94
$B_{mi1}(Ns/m)$	6.32	6.32	6.32				

The parameters used for the Articular and Cartesian space are the same and correspond to the minimum of them that allow to get an underdamped reaction in the case of the Cartesian approach, as the following simulation shows.

Simulation results

From the results, we determined that the Articular approach performs a perfect tracking of the trajectory. There are no noticeable deviations or overshoots during the task, as it is seen in Figure 3-12.

In the case of the Cartesian approach, the Master 2 device has overshoot problems during the beginning of the trajectory. Once the system has established, the architecture follows the position in with the same level of performance that the Articular, as it is seen in Figure 3-13.

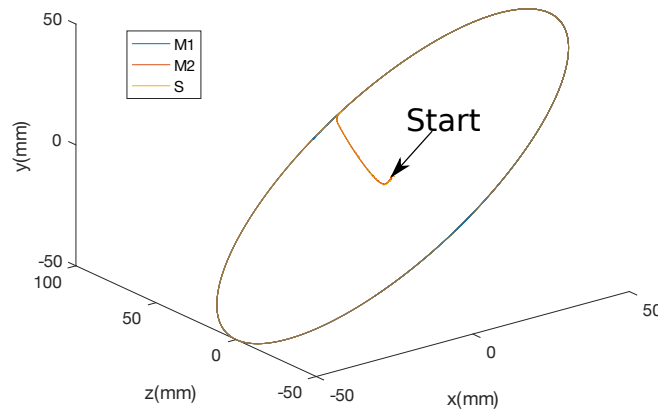


Figure 3-12 – Expansion n -DoF: simulated Articular approach trajectory

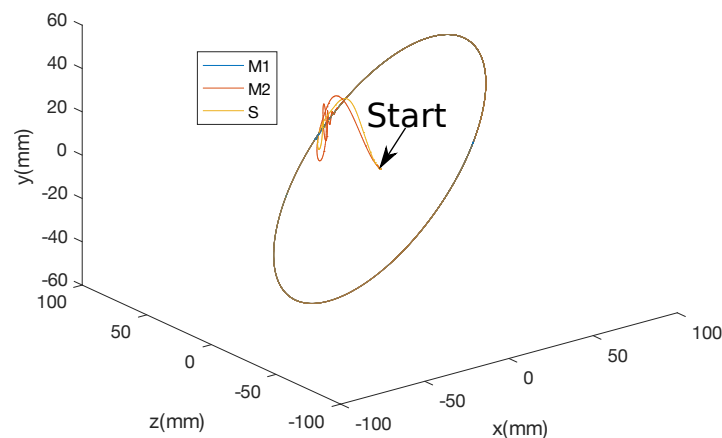


Figure 3-13 – Expansion n -DoF: simulated Cartesian approach trajectory

For a better appreciation, Figure 3-14 shows the position tracking for the Articular approach, and Figure 3-15 the error in the Cartesian space. In this case, the Articular architecture can reach the

desired position without any undesired behavior in a smooth and fast way. For the Cartesian approach, Figure 3-16 shows that the system suffers from overshooting at the beginning of the task, mainly in the axis Y due to the gravity term. Once the system is stabilized, the tracking of the trajectory is perfect as it is with the Articular.

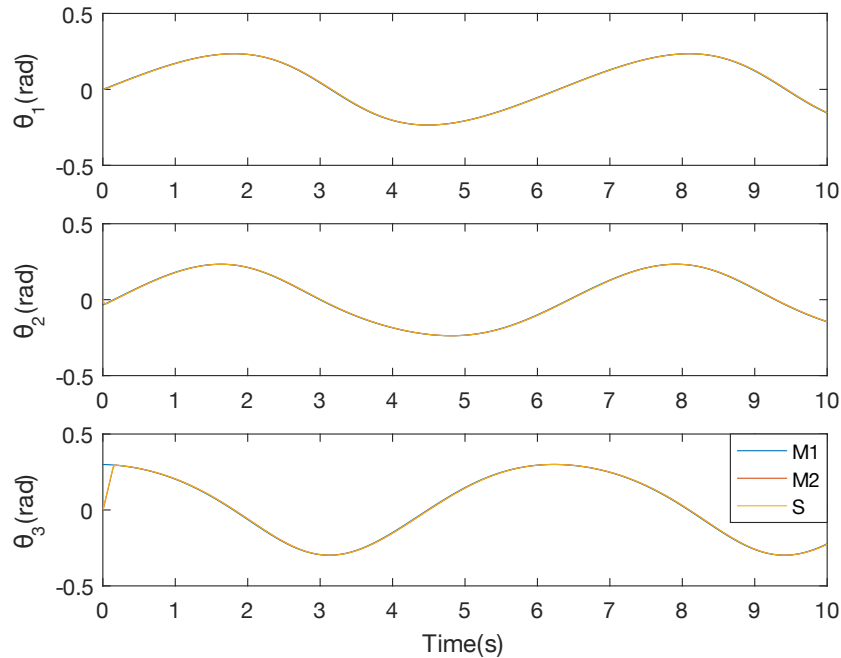


Figure 3-14 – Expansion to n -DoF (Cartesian approach): simulated Articular position

Based on the obtained results, the Cartesian and Articular approaches have similar performance, with an RMS error of 1×10^{-6} m for the Cartesian and 4×10^{-6} m for the Articular positions errors. Unlike with the Articular Approach, with the Cartesian approach, the position tracking shows an initial transient mode during around a tenth of a second. This is the only drawback of this approach. It may be due to errors obtained during the first computations of the Lagrange matrix.

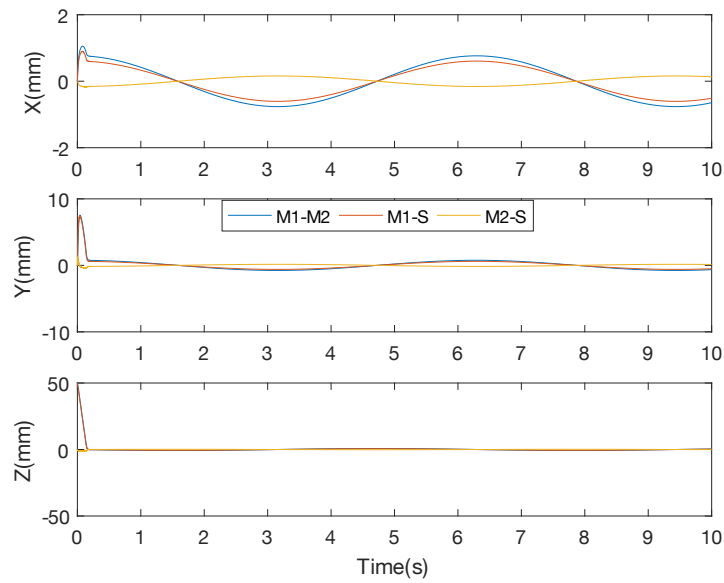


Figure 3-15 – Expansion to n -DoF (Cartesian approach): simulated position error

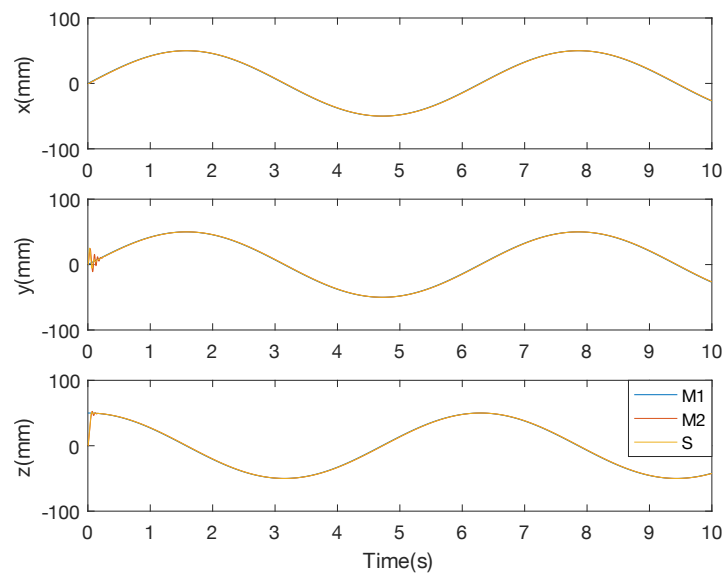


Figure 3-16 – Expansion to n -DoF (Articular approach): simulated Cartesian position

3.4 Test bench for experimental results

3.4.1 Core Setup

The following setup is used during the results shown in Part II. Some variations are made to adapt the setup for new experiments, but the base core is not changed at all. The changes made for the new experiments are addressed in their corresponding sections. The core of the setup is composed of three Geomagic 3D Touch™ haptic devices, shown in Figure 3-17 with the considered degrees of freedom (θ_1 , θ_2 and θ_3). Table 3.3 shows the specifications of the device (3DSYSTEMS [2019]).

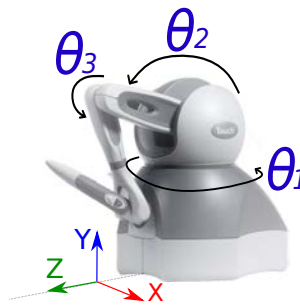


Figure 3-17 – Geomagic 3D Touch™

Table 3.3 – Geomagic 3D Touch specifications

Haptic Device	Geomagic 3D Touch
Degrees of freedom	6
Actuated Dof	3
Workspace (mm)	160x120x70
Nominal position resolution	55 μ m
Max. Normal Force	3N
Force/Torque sensors	No
Communication protocol	Ethernet
API Software	Openhaptics

The three devices are connected to one computer with two screens in duplicate configuration. The slave is set close to the testing bench, which represents the environment and consists in a 3D printed model disc with three possible paths: a plain one, a guided one and one with peaks that are used to get the sensation of touching a rough pattern (Figure 3-18).

For the Master 1 and Master 2 devices, both are located close to a screen of the computer. A wall is set between each device to avoid the users can directly see their partner device or the slave. A webcam pointing to the slave provides visual feedback for each user. The screens show the streaming video

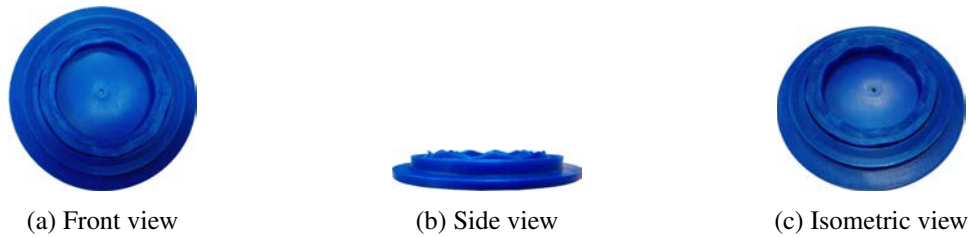


Figure 3-18 – 3D printed environment

from the webcam in conjunction with the position and force plots of Master 1 and Master 2. Figure 3-19 shows the organization of the setup.



Figure 3-19 – Setup organization

As these devices are 6-DoF systems but only three are actuated, the three non-actuated joints, which are the wrist joints, are immobilized.

To communicate with the haptic devices, we use the Open Haptics® software library on the computer. This library is the official provided API for the utilization of the Geomagic 3D Touch devices. The Open Haptics API connects with the Phantorque library introduced by Aldana et al. [2014], which consists of a Mex file¹. As Phantorque can communicate with only two haptic devices at the same time, a modification to this library was made to enable the communication with more devices. Also, the modified library grants access to the physical buttons of the Geomagic 3D Touch can activate the control in the Cartesian space initially, or the control in Articular space was the only one available. Phantorque communicates with Simulink, where the ESC architecture was implemented in. During all the tests, the real-time clock was supplied by Phantorque.

Figure 3-20 shows the connection framework for a better comprehension of the communication between the ESC architecture and the haptic devices. In this case, the only shared values are the

¹An executable file that Matlab uses to perform the process with high priority in the Operative System environment

position signals $P_{m1,m2,s}$ (which can be Articular, Cartesian or a transformation matrix of the haptic device) and the commanded signal $C_{m1,m2,s}$ (which can be force or torque values to be applied by the haptic device). The torque and force controllers are provided by the Geomagic Touch 3D internal software.

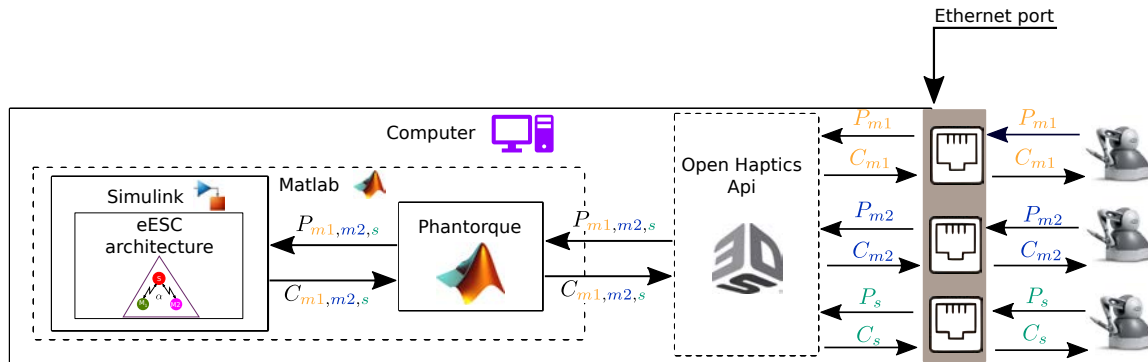


Figure 3-20 – Framework connection

The parameters of the IPCs (masters and slave, independently for each joint) were computed according to the method of section 3.5.1. The value of the parameters are listed in Table 3.4 and are used in all the experiments of part II, unless the contrary is indicated.

Impedance (Master IPC)				Admittance (Slave IPC)			
Parameter	θ_1	θ_2	θ_3	Parameter	θ_1	θ_2	θ_3
$M_1 (kg)$	0.002	0.002	0.002	$M_2 (\frac{kg}{m^2})$	0.002	0.002	0.002
$k_{mc} (\frac{Nm}{rad})$	20	40	15	$k_s (\frac{Nm}{rad})$	20	40	15
$k_{mi} (\frac{Nm}{rad})$	20	40	15	$B_{mi} (\frac{Nms}{rad})$	0.4	0.56	0.34
$B_{mi} (\frac{Nms}{rad})$	0.4	0.56	0.34				

Table 3.4 – IPCs' parameters for Articular approach

3.4.2 User interface

During the test, the trainer tries to keep the same position in each axis, but as soon he corrects in one axis, the others get misaligned. Because of this, we designed a new user interface to correct this problem.

Simulink is a very flexible tool that allows modifying the architecture and control scheme but does not provide the tools to create a user-friendly interface. For this reason, we create a new user interface using the Chai3D framework and Visual Studio 2017 as the compiler.

The Chai3D framework is an open-source framework, written in C++, developed for the creation of 3D virtual environments for haptic devices (Conti et al. [2003]). Chai3D provides all the functions, libraries, and tools needed to connect with several haptic devices, including the Geomagic Touch 3D. This framework also provides a set of functions for the manipulation of virtual objects as well as the calculation of collisions, deformation, and forces. Ideally, Chai3D is only used to create these virtual environments, lacking the tools offered by other frameworks to develop control schemes.

Instead of translating the ESC architecture and the IPC control into C++ language for being used with the Chai3D framework, we opted to create memory zone location in the RAM that allows Simulink and the developed user interface to share data. By implementing the boost libraries into the Chai3D framework (Boost [9 02]), we create the memory zone. This memory zone is managed by the operative system (Windows 10 in this case) and allows that both processes run in parallel. As the tests are with real environments, it is not necessary to get force calculation from them, and thus the user interface works as a visualization of the actual position of both devices. To better understand this, Figure 3-21 shows the new framework connected with the use of the new user interface.

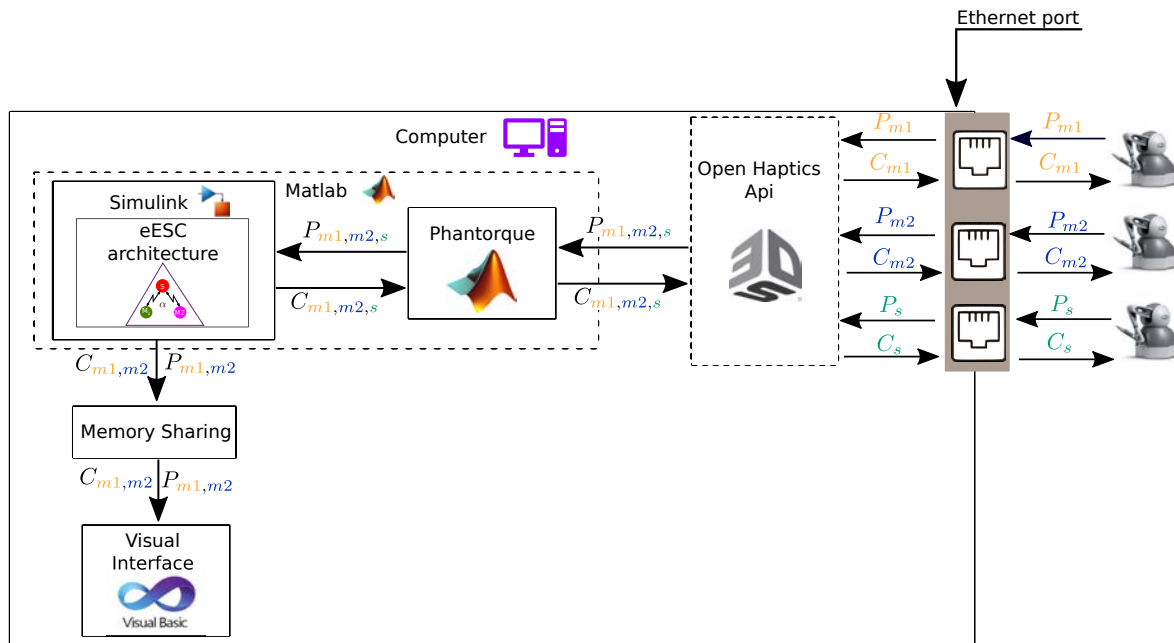


Figure 3-21 – Framework layer hierarchy with the new user interface

The user interface consists of two spheres, blue for Master 1 and green for Master 2, two force bars and an isometric plane with texture. The spheres provide an intuitive way for the users to locate their device respecting her/his partner, in a 3D Cartesian environment. A gray line links both spheres to

help determine how far is each from the other. To give a perception of depth, the spheres are displayed over an isometric plane with a brick-wall texture. Once the slave interacts with the environment, the follower user needs to exert the same effort onto his device. To help them apply the same force, on both sides, two side-bars are colored according to the force exerted by each device. Figure 3-22 shows a screenshot of the resulting user interface.

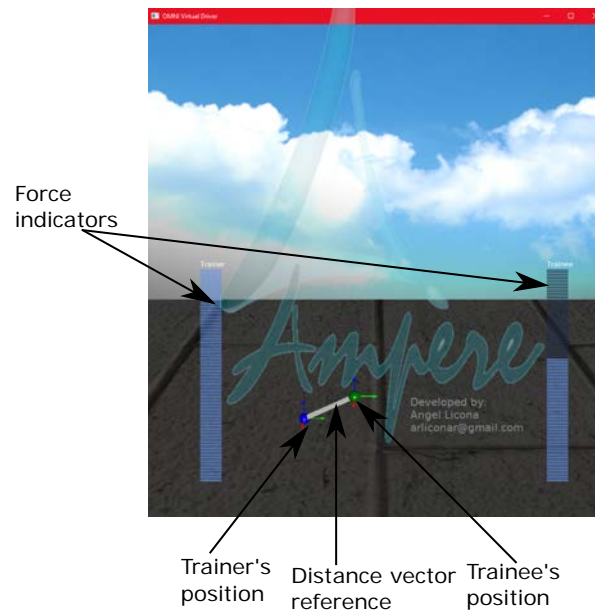


Figure 3-22 – Chai3D user interface

3.5 Experimental validation for Articular approach

3.5.1 Determining control parameters

As established in section 3.3.3, the only parameter that must be determined for the IPC controllers is the damping factor $B_{mi1,mi2}$. We propose here a method to determine the stiffness of the IPC controllers, based on the reaction of the haptic device.

In this method, masses $M_1 > 0$ and $M_2 > 0$. Stramigioli [1996] also states that for better performance, the virtual mass in an IPC must be lesser than the one of the devices to be controlled. In our tests, we observed that both masses must be between 0.001 and 0.01 kg to reduce the chattering problem and accords with the parameters of our haptic devices (detailed in Appendix A.1). In the case of Master and Slave devices having the same kinematics, we propose to use $M_1 = M_2$. If these are not set as suggested, the system remains stable due to its passive nature. However, the difference between

both masses can change the momentum generated by them, leading to a change of the selected springs, dampers, and different establishment time of the haptic device.

The stiffness of the springs can be set by determining the minimum reaction force/torque (R_f). R_f is the minimum input signal that is necessary to take it off from the stand position. To acquire it, we made an experimental test by injecting a ramp signal of force/torque as an input in the haptic device. By plotting the position and the input signal, it is possible to determine R_f by observing at which value of the ramp signal the system starts to move. Figure 3-23 shows the determination of R_f in the axis X with a Geomagic® 3D touch. As one can observe the relation between the position and the exerted force is not linear, with the presence of a threshold that increases the steady-state error of the system. To ensure the proper position tracking of the IPC controllers, we determined the minimum force at which the system reacts to reduce this gap as short as possible.

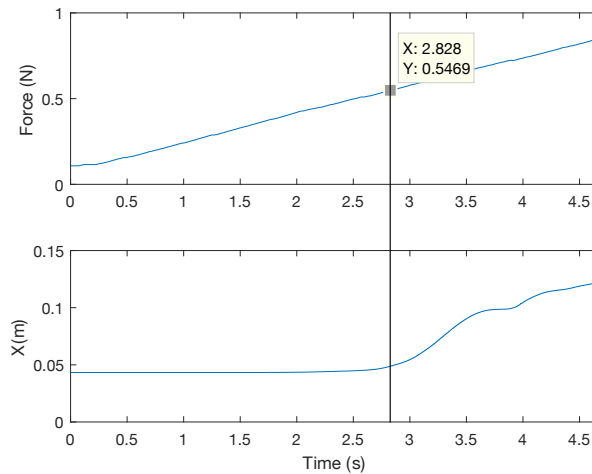


Figure 3-23 – Determining R_f in X axis with a Geomagic 3D Touch

From our test, the reaction force is $R_f = 0.54\text{N}$ as the system starts to actuate when a force of around 0.5N is injected. Also, the system has a non-linear behavior, especially noted from $t > 4\text{s}$, but this result is neglected as it has been supposed in section 3.3.3 IPC controller is robust enough to compensate it. Although $R_f > 0.5\text{N}$ indicates that the system requires at least this force to be actuated. This implies that the haptic device can not exert forces lesser than this. R_f only defines the necessary force required to set off the system from the static repose. Once the system is in motion, it is possible to render forces lesser than it. With R_f it is then possible to get the value of k_{mc}/k_{mi} with

$$k_{mc} = k_{mi} = \frac{R_f}{\Delta_r} \quad (3.33)$$

Where Δ_r is the minimum error allowed before the system can be actuated, this value is in meter/radian, and it is directly inverse proportional to the value of the springs. The lower Δ_r is, the more accurate the controller is, but this can produce chattering effects. From our tests the value that performs the best is $1 \times 10^{-4} < \Delta_r < 5 \times 10^{-4}$ m/rad. In the case of the admittance control for the slave, k_{mi2} can be equal to their Master counterpart k_{mi1} as long as the Slave device shares the same kinematic configuration. If this is not possible, a proper R_f must be determined.

3.5.2 Types of experiments

To validate the hypothesis that the eESC architecture can perform position and force tracking, two experiments were done in a training proposes context, with the Articular and Cartesian approaches.

The experiments performed in this thesis are categorized into two types: Free motion and Wall Contact.

In Free Motion, the slave device can move freely in the workspace without interacting with the environment. This test is made to evaluate the position tracking of the slave and the follower master according to the leader device. During the test, the follower device attempts to move away from the trajectory given by the leader (a circle in this case). The architecture responds by injecting a correctional force/torque into the lead device until it returns to the desired position. The haptic forces the user gets from the slave are the ones created by the inertia of the slave, which are considered negligible, and in the case of the lead device, the correctional torque.

In Wall Contact, the slave device works in free motion during a few seconds. The leader device takes the slave into a collision with an established wall or environment that can be soft or hard. This test is made to validate the torque/force tracking of the slave to the master devices. The forces/torques that the master devices exert on the user's hand must be equal to the one that the slave is exerting to the wall, assuming the slave or the master has not reached their force/torque saturation point. For the follower devices, if it is overpassing the position of the leader, the architecture will exert more force/torque to correct with the leader. From this point, it is not possible to distinguish the difference between the environment force feedback and the correctional force. It is required to use the human interface to compare the force exerted by the slave and the correctional one.

To get a better analysis of the results of both tests, we compute the Root Mean Square (RMS) error for the position and force error signals (Appendix A.2). The computation of RMS values helps us quantify the tracking performance between the follower device and the leader devices. This RMS

error can be computed for position tracking in Cartesian space, Articular space, and to analyze the force and torque tracking.

3.5.3 Articular expansion

Free Motion for eESC architecture

In this first experiment, the system worked in Demonstration mode ($\alpha = 1$) at first and changed it to evaluation ($\alpha = 0$) at $t \approx 11.5$ s.

To show that the device of the following user leads him on the track of the leader, each user performed a gesture to take his device away from the prescribed trajectory (the trainee in demonstration stage (a) and then the trainer during the assessment stage (b) in Figures 3-24 and 3-25).

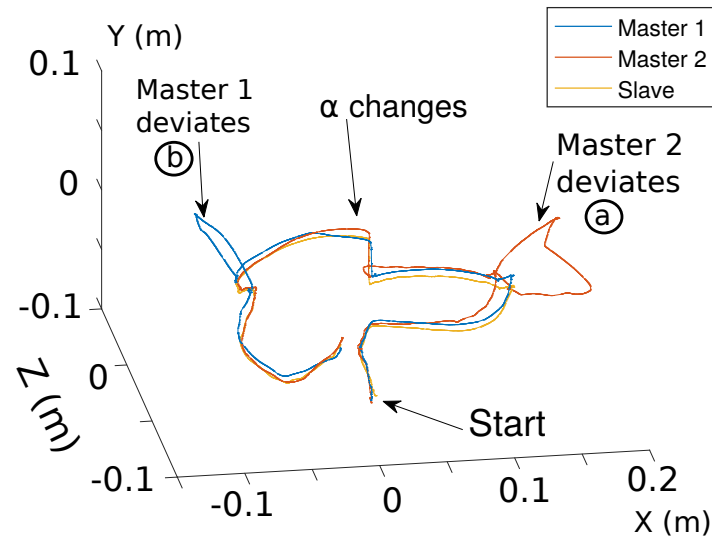
Figure 3-24 depicts the global trajectory performed by the users in free motion. This plot shows that, out of the two attempts to take the device away, the three trajectories follow close trajectories. The results are presented in Articular and Cartesian approaches. Although the control is in the Articular approach, the Cartesian position of the system can be obtained directly by the Phantorque library or by using the direct kinematic presented in Appendix A.1.1. The RMS error value of the tracking error (computed from the positions expressed in the Cartesian space) between the Master 1 and slave devices during demonstration mode was 8.2 mm (and 6 mm between the Master 2 and the slave during demonstration mode). This level of precision is to be taken into account when choosing training applications. It can suffice for first hands-on training but may not be precise enough for more accurate training tasks. To enhance precision, it is suggested to use more precise haptic devices, which could, in turn, allow to raise the gains of the IPC and enhances the precision of the whole system.

Figure 3-25 shows the Articular positions of the three actuated joints. The error of all the trajectory between Master 1-Master 2 (M1-M2), Master 1-Slave (M1-S), and Master 2-Slave (Slave) are shown in Figure 3-27. One can observe that the precision is the best for θ_1 (tracking error of 17.10^{-3} rad RMS between Master 1 and slave when $t < 5$ s) and the worst for θ_2 (55.10^{-3} rad RMS).

This can be explained by the weight of the arm, which disturbs the IPC controller. A weight compensation approach should be envisaged to raise the precision of this joint control.

Besides, it is logical that the follower user position does not overlap with the two others as this user is led by their device to follow the slave trajectory. According to the resistance she/he brings (the stiffness of the hand), the follower position is shifted from the reference trajectory.

Figure 3-26 shows the exchanged torques $T_{r1,2,s}$ for the three actuated joints.

Figure 3-24 – Expansion n -DoF: trajectory in free motion

The three plots show a proper torque tracking for the three joints. As the motion is performed in free motion, these torques should be close to zero if neglecting the aforementioned disturbances. Indeed, in demonstration mode, the master 1 (resp. master 2) and slave torques overlap.

Moreover, one can observe the torque generated by the devices to bring back the following user to the reference trajectory at marks (a) and (b).

Also, the sudden change of α results in a small sudden change of T_1 and T_2 , but in practice, it is hardly perceived by users. It has not any adverse effect on energy levels (see Figure 3-28), which remain negative.

Wall contact for eESC architecture

On wall contact for eESC, the dominance factor α is kept to 1, corresponding to the demonstration mode where the trainer (with master 1) shows an interaction to the trainee (with master 2). The trajectory is visible in Figure 3-29.

The trainer leads his device to apply the slave effector on a solid surface (for the force exerted by the haptic device), to show the behavior of the system in contact with a hard organ, like a bone, that it may encounter in a medical context.

Once in contact, he asks the trainee to come in the same position to be able to feel the same interaction force. This can be very useful in a training context to learn how to dose forces to apply.

To better understand this experiment, it is necessary to understand how the force tracking works

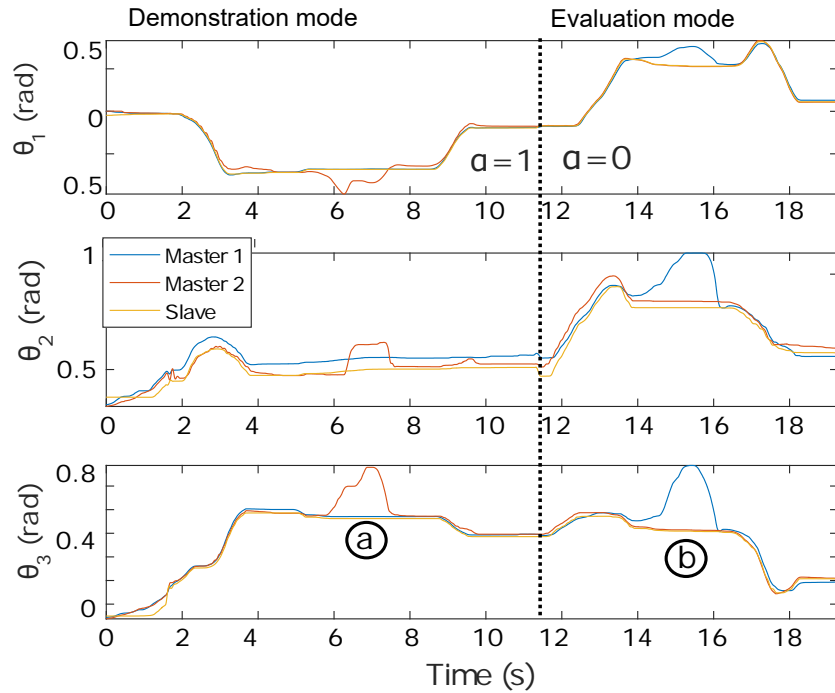


Figure 3-25 – Expansion to n-DoF: Articular positions in free motion

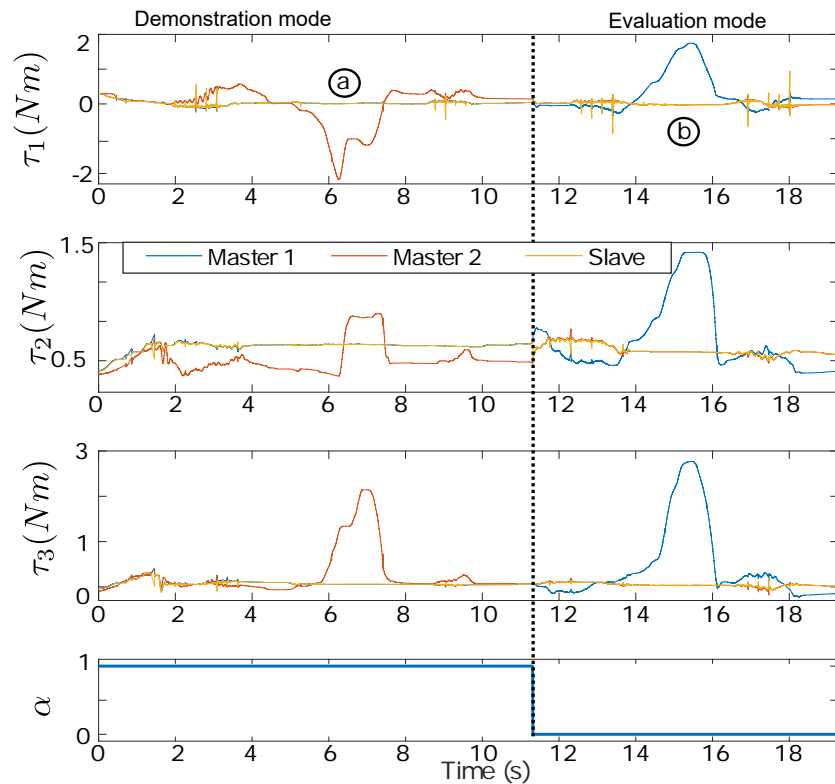
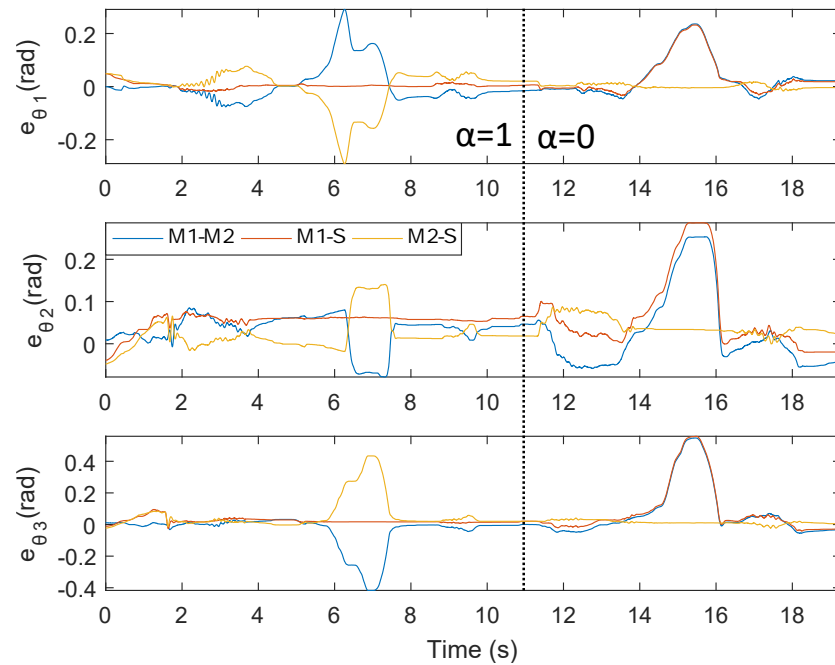
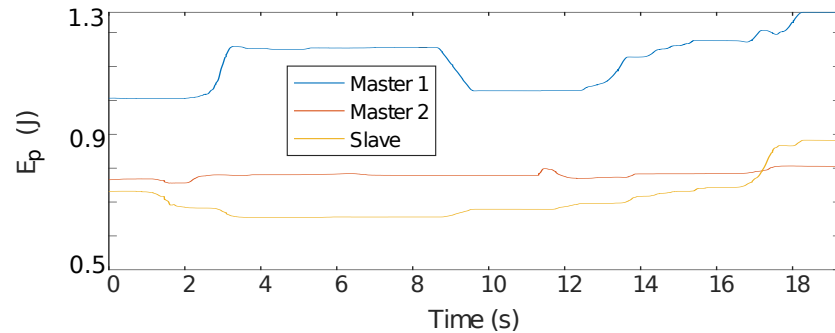


Figure 3-26 – Expansion to n-DoF: torques and α in free motion

Figure 3-27 – Expansion to n -DoF: Articular error in free motionFigure 3-28 – Expansion to n -DoF: Energy E_p in free motion

in teleoperation systems. First, when the slave contacts the wall, the leader user applies some effort on the leader device, which positions its device beyond the position of the surface and permits him to get haptic feedback from the slave. This phenomenon depends on the stiffness of the surface, the power of the haptic device, and the controllers' gains. The more they are elevated, the less the leader master device will go beyond the real surface position, and the transparency will be better. This limitation is due to a compromise between stability and transparency met in every teleoperation system.

The positions of the masters are virtually beyond the surface (around 4 cm) while the slave effector stopped in contact, which corresponds to stiffness of 250 N/m.

Figure 3-30 shows the position of the effectors along the x axis in Cartesian space and the resulting

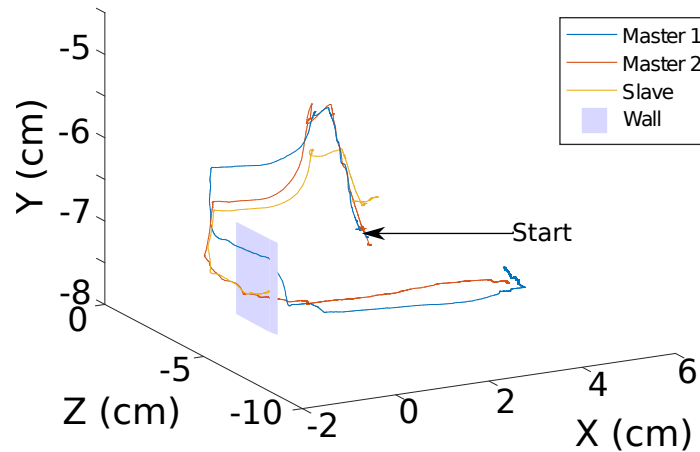


Figure 3-29 – Expansion to n -DoF: Trajectory in wall contact

forces in this direction. At $t = 3$ s, the slave comes into contact with the vertical surface of the piece located at $x = 0$ in the plane yz .

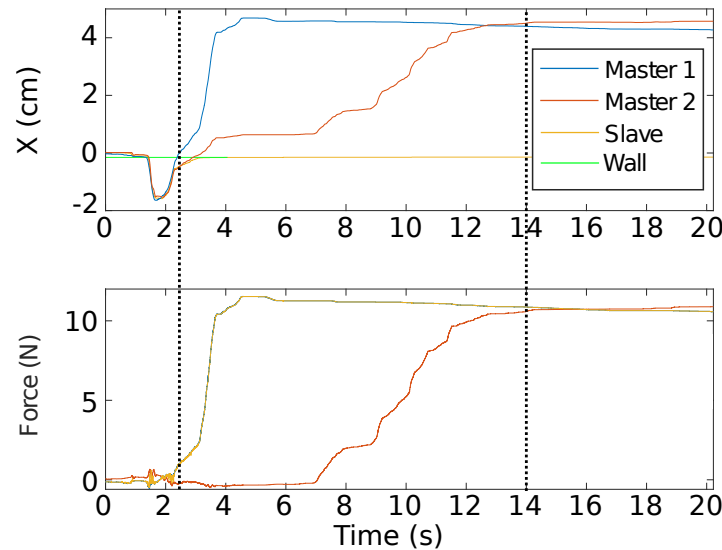


Figure 3-30 – Expansion to n -DoF: Position and force along Cartesian x in wall contact

As long as the trainer pushes his own device "inside the piece" the force feedback increases accordingly. In this plot, the slave force overlaps the Master 1 feedback force.

In steady-state, the devices being immobile and as users move along the horizontal x axis, there is no torque disturbance in this direction. In the case of vertical interactions, both torque plots would be shifted together to compensate for the gravity (when touching a surface from bottom).

Despite the aforementioned disturbing positioning, the force tracking error is 0.35 N RMS, corresponding to 3.5% of the magnitude of the forces.

When the trainer decides that this corresponds to the right level of force to apply in a training context, he stops his motion and asks the trainee to join him at the same location, which the latter does at $t \approx 13$ s.

The follower aligns his device at the same location as the trainer's (error of 2 mm RMS in x direction when $t > 13$ s) only by way of the video feedback and the real-time plots displayed on his monitor. He then feels force feedback which is very close to the interaction force (error of 0.2 N RMS (2%) when $t > 13$ s) as far as we can estimate it.

Figure 3-31 shows that MSF energy levels are not influenced by the contacts.

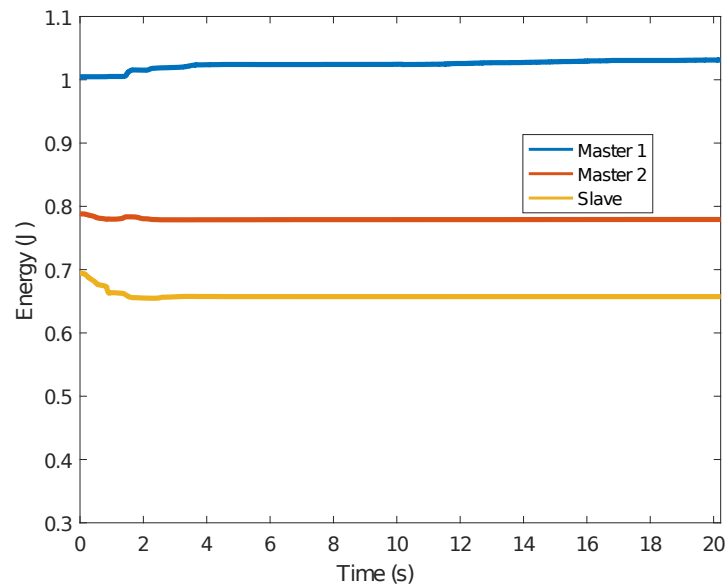


Figure 3-31 – Expansion to n -DoF: Energy E_p in wall contact

3.5.4 Cartesian expansion

Experimental setup for Cartesian expansion

For the experimental validation of the architecture in the Cartesian expansion, we implemented two identical Geomagic Touch 3DTM and one VirtuouseTM 6D (Fig. 3-32) as the slave device. Table 3.5 shows the characteristics of the Virtuouse device.

The control and architecture were implemented using Matlab 2016a. For the communication with Geomagic Touch 3D the library, we use our modified Phantorque library, and for the case of the Virtuouse, the Virtuouse Matlab library (Virtuose Mex) developed by Alamilla Daniel et al. [2018]. As the Haption Virtuouse device does not have a torque/force sensor, the measured effort is obtained directly

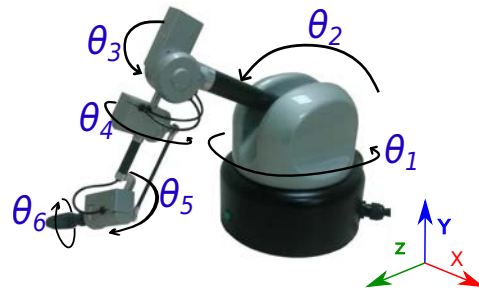


Figure 3-32 – Haption Virtuose 6-Desktop

Table 3.5 – Haptic Device’s specifications

Haptic Device	6Dof Desktop
Actuated Dof	6
Workspace (mm)	185x200x260
Max. Force	10N
Force/Torque sensor	No
Communication protocol	Ethernet
API Software	Virtuose

from the control signal that the architecture delivers. Figure 3-33 shows the framework connection for this test bench.

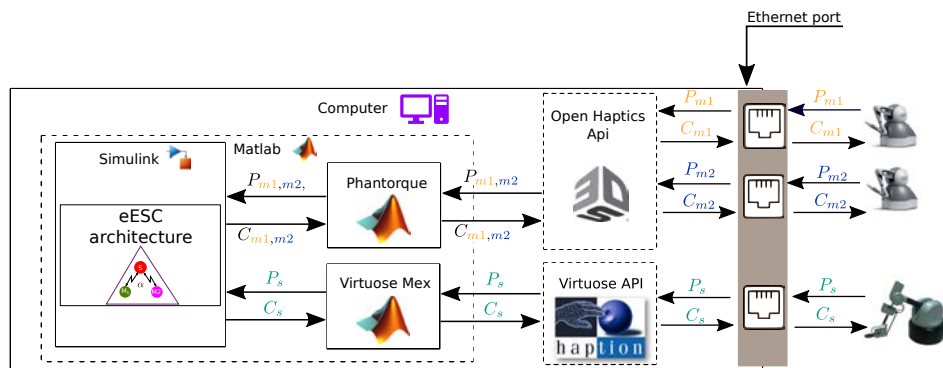


Figure 3-33 – Framework connection for Haption Virtuose slave

Table 3.6 shows the parameters of the Cartesian IPC for all the experiments made with the Cartesian expansion in Part II.

Three experiments were proposed to validate the architecture: Position tracking, Position tracking with eAAA, and Force Tracking. In this case, we are using a two master device, so the multi-trainee eESC architecture works in a dual-user approach.

Impedance (Master IPC)			Admittance (Slave IPC)				
Parameter	X	Y	Z	Parameter	X	Y	Z
$M_1 (kg)$	0.002	0.002	0.002	$M_2 (kg)$	0.002	0.002	0.002
$k_{mc} (\frac{N}{m})$	400	800	60	$k_s (\frac{N}{m})$	400	800	60
$k_{mi} (\frac{N}{m})$	400	800	60	$B_{mi} (\frac{Ns}{m})$	1.78	2.52	0.69
$B_{mi} (\frac{Ns}{m})$	1.26	1.78	0.48				

Table 3.6 – IPCs' parameters for Cartesian approach

Free motion

This test evaluates if the follower devices can follow the trajectory proposed by the leader. The test consists of a trajectory in free motion given by the leader device. In the first half of the test, $\alpha = 1$ which means the system is running in demonstration mode and the trainee will try to oppose the trainer in a determined time. The architecture responds by exerting a feedback force to correct the position of the guided device. Then the dominance is turned by the trainer to $\alpha = 0$, and the system works in evaluation mode, and as it was made with the first half, the process is repeated but now with the roles changed.

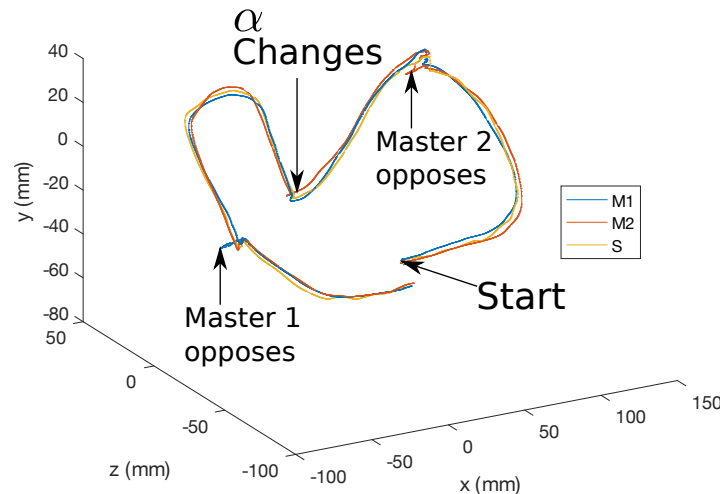
Figure 3-34 – Expansion to n -DoF (Cartesian approach): Trajectory in free motion

Figure 3-34 shows the trajectory performed by during the test, while Figure 3-35 shows the Cartesian position of each device and Figure 3-37 the position errors. During $0 < t < 11$ s and $20 < t < 25$ s Master 2 (M2) follows the trajectory of Master 1 (M1) without exerting any effort. In these periods the total Cartesian RMS errors are 0.96mm and 2.18mm between M1-M2, and for M1-S are 1.29mm and 1.44mm. From $11 < t < 20$ Master 2 opposes the position of Master 1, and the architecture reacts by

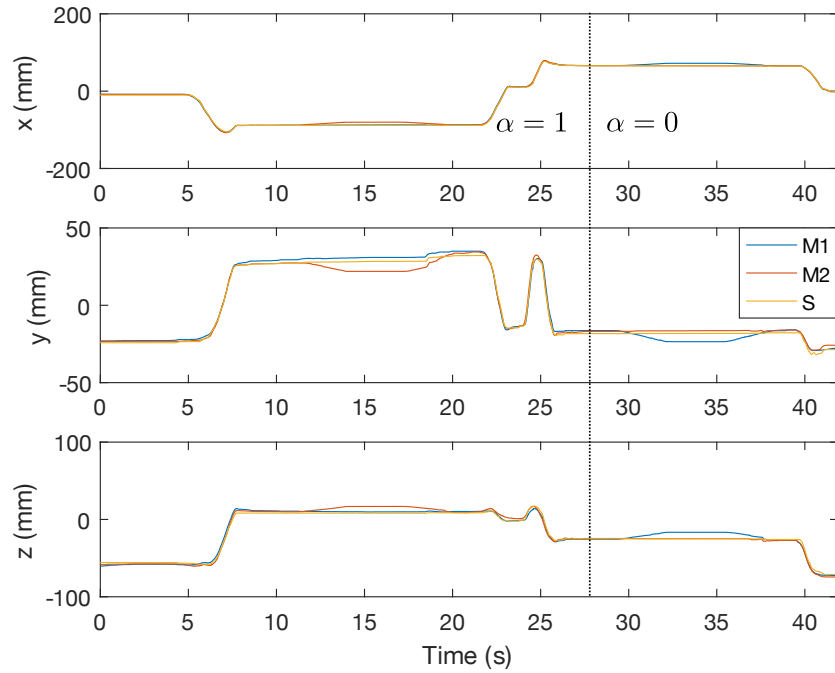


Figure 3-35 – Expansion to n -DoF (Cartesian approach): Positions

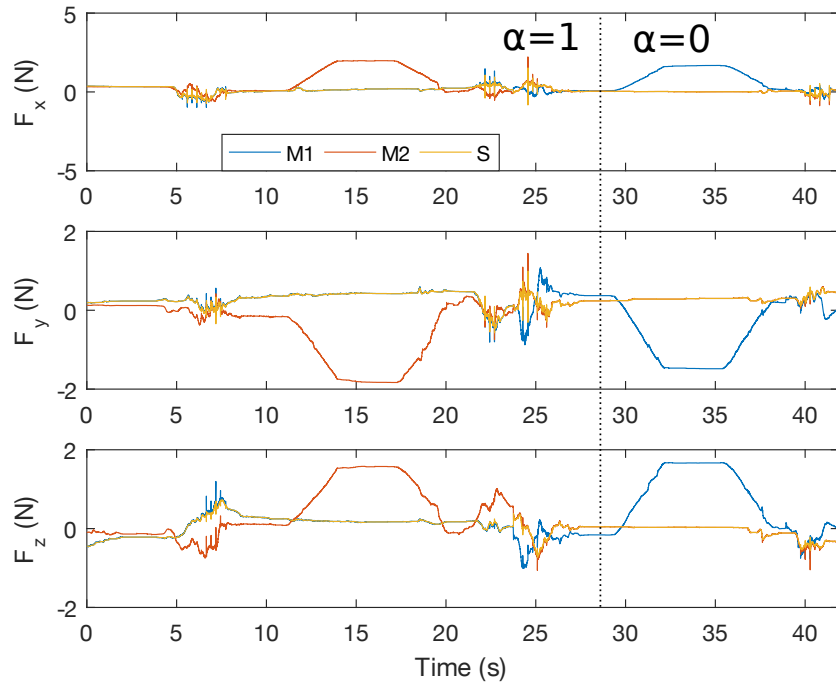


Figure 3-36 – Expansion to n -DoF (Cartesian approach): Forces

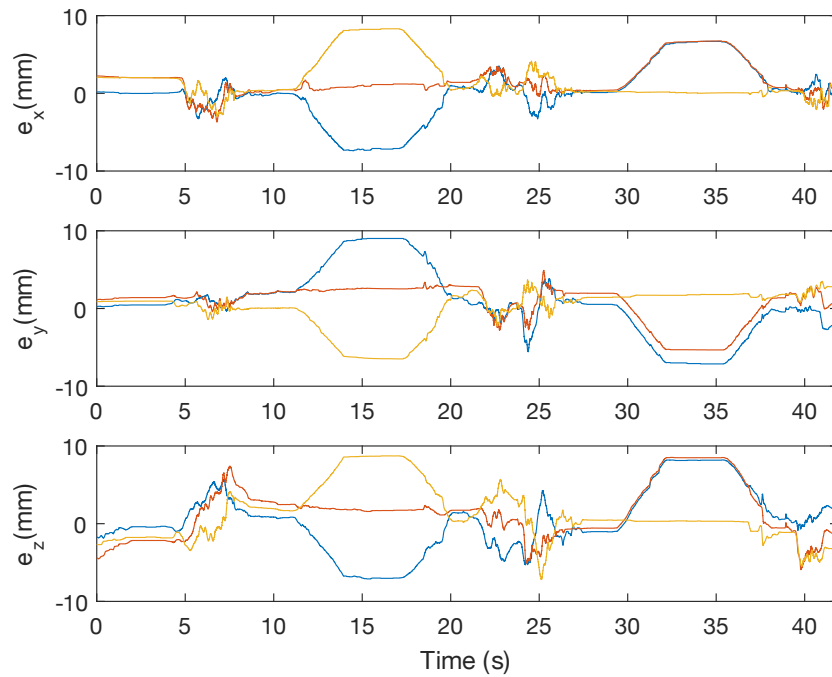


Figure 3-37 – Expansion to n -DoF (Cartesian approach): Positions error

exerting a correction force at it is shown in Fig. 3-36.

At $t = 26$, the change of dominance occurs ($\alpha = 1$) and the system works in evaluation mode. From $26 < t < 29$ and $38 < t < 41$ the RMS errors between M2-M1 are 0.31mm and 0.35mm. In the case of M2-S, the RMS is 0.26mm and 1.35mm. As it is shown in the experiment, the average error is less than 2.5mm, which can be considered good in a surgery performed by a human being.

Wall contact

Once position tracking is validated, the system must be able to perform force tracking as well. For this, a contact test was performed using three axial walls (once for each axis). For this test, the system works in demonstration mode. The trainer exerts a force on the wall while the slave and trainee remain at the edge. After a few seconds, the trainee tries to match the trainer's position. As soon the trainer reaches it he/she can feel the same interaction force as the trainer. Figure 3-38 shows the position, Figure 3-40 the position errors, Figure 3-39 the force plots and Figure 3-41 the forces errors.

The test is divided into three sections (a), (b), (c) where the device interacts with a "wall" each time set at different position and orientation. In section (a) the test is performed on a wall $X = 50$ mm. At $17 < t < 19$ s the trainee applies a force on his device to reach trainer's position. The RMS errors in this section are 0.1mm for position and 0.02N for force. In the section (b) the test is performed into

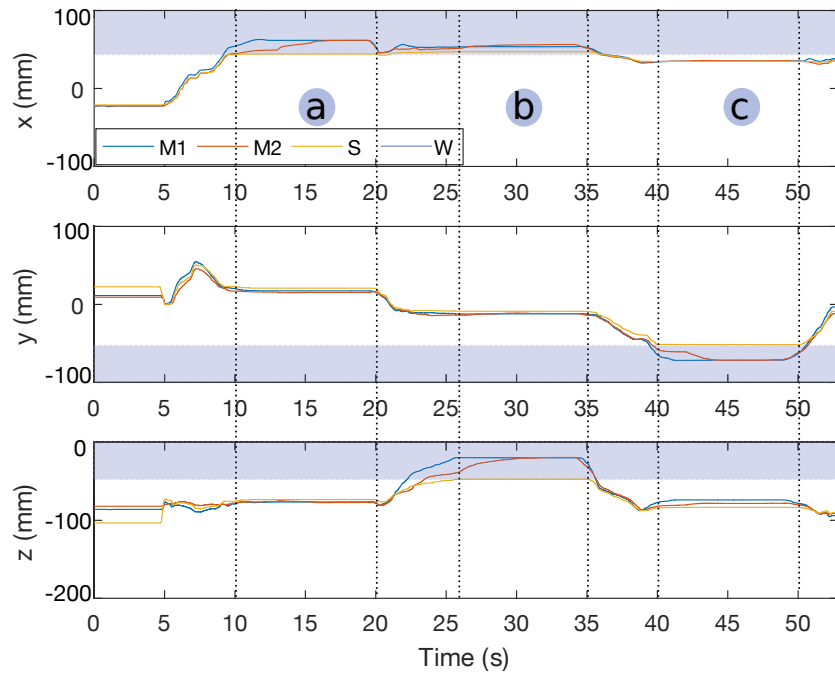


Figure 3-38 – Expansion to n -DoF (Cartesian approach): Positions in wall contact

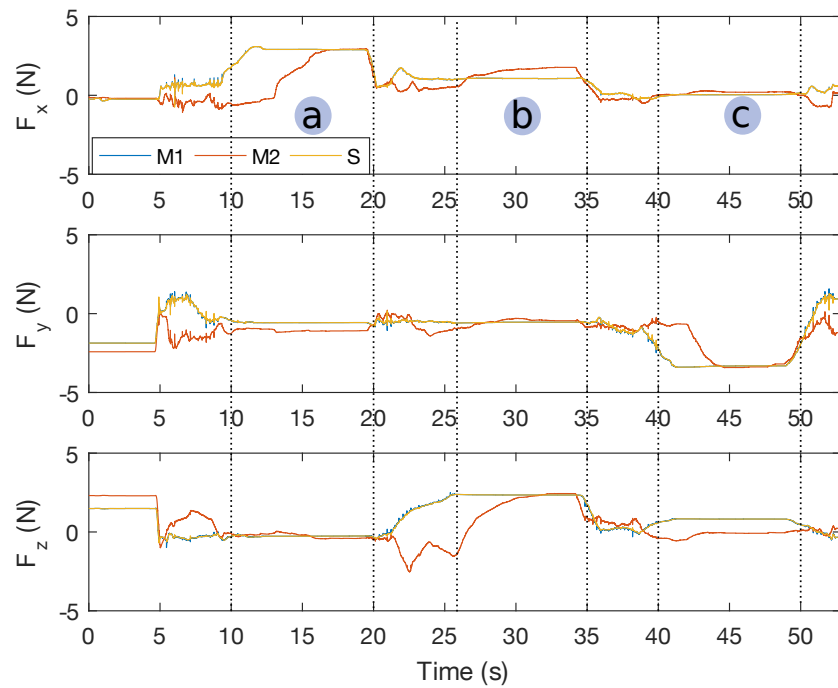


Figure 3-39 – Expansion to n -DoF (Cartesian approach): forces in wall contact

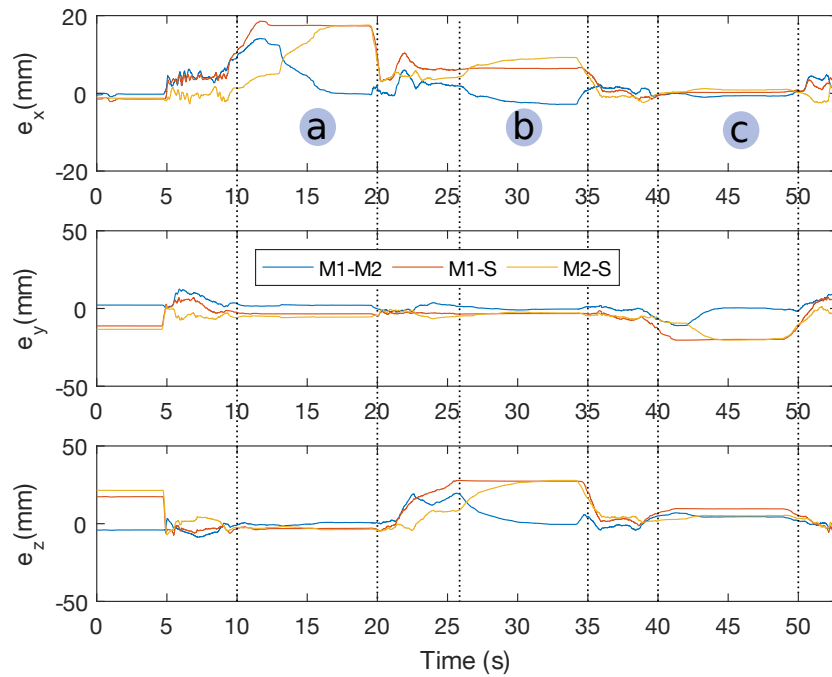


Figure 3-40 – Expansion to n -DoF (Cartesian approach): position errors in wall contact

the wall $Z = -50\text{mm}$. When $31 < t < 34\text{s}$ the RMS errors are respectively 0.37mm and 0.74N . Finally, in section © the test is performed on a wall located at $Y = 50\text{mm}$. At $44 < t < 49\text{s}$ the position errors are 0.34mm , and force 0.08N .

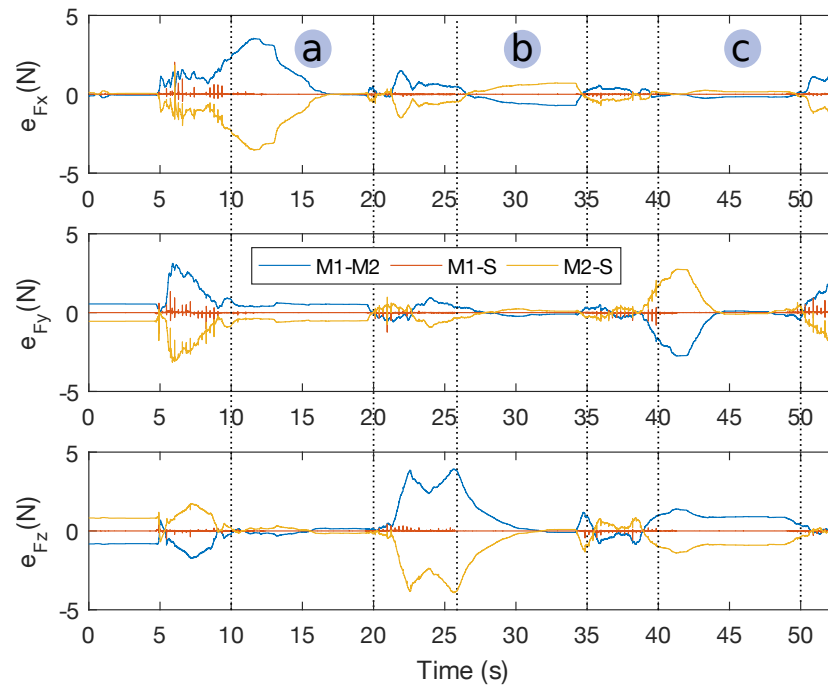


Figure 3-41 – Expansion to n -DoF (Cartesian approach): force errors in wall contact

3.6 Discussion

For the Free-motion test, the results have shown that the eESC is able to achieve a full hands-on training system. The resulting precision level can be considered as a drawback (lowest error of 6 mm) when practicing precise incisions. Despite this, its use as an introductory simulator can be beneficial, as the trainees can get used to the process, the forces, and the implied methods in the task. Then when students dominate it, they could try a more accurate simulator to improve their performance. Also, the torque plots show that the eESC offers a minimal force that is activated during the motion. This torque is exerted by the system as a result of the IPC's inner dynamic. We called this a parasite torque, that although it is felt by the users, it does not disturb the movement and interaction as the force and frequency of the torque is almost negligible.

For the wall contact test, the system is limited by the balance between transparency and stability that comes with every TDPC system. The trainee must have the same position as the trainer to get the exact amount of torque feedback.

It is validated that eESC architecture can work in Cartesian expansion to provide a haptic simulator able to work with devices of any kinematic configuration. The system can perform trajectory tracking. For the contact case, the system can also perform force tracking when both master devices share the

same position. With this system, it is now possible to create simulators that could use a haptic device with different kinematic configurations. An optimal configuration for a training simulator would be with all the users manipulating the same kind of haptic devices, and a slave device more adapted for the training task.

Chapter 4

Enhanced AAA

4.1 AAA for ESC

Although the dominance factor α can be changed by the trainer during the training task, it is also desirable that the system can adapt the dominance of the system between trainer and trainee based on how well performs the second one. The ESC architecture comes with an inner Adaptive Authority Adjusting (AAA) function proposed by Liu et al. [2015] as well. The proposed AAA uses three functions, Adaptive Virtual Boundary, Virtual Boundary Torque and the Adaptive Control Authority, that allow the architecture to determine the best value of α .

4.1.1 Adaptive Virtual Boundary

This function is in charge of determining the region where the trainee can move. It is almost impossible that trainer and trainee share the same position at the same time (as factors like the precision of position sensors and involuntary movements made this an arduous task) it must exist a boundary zone where the user can move freely without disturbing the dominance.

To establish the boundary, first, it is deduced the maximum allowed distance B using equation 4.1 (Shahbazi et al. [2013]).

$$B = B_0 - k_b \cdot \ln(\alpha_a + \varepsilon) \quad (4.1)$$

where B_0 is the maximum allowed distance and α_a is the Adaptive dominance factor. The use of the $\ln(\cdot)$ function allows the function to have a non-linear and smooth boundary. k_b is a parameter,

and ε is a minimal value that avoids zero cases for the logarithmic function.

Equation 4.2 set the error between Master 1 and Master 2.

$$d_{\theta_m}(t) = (\theta_{m1}(t) - \theta_{m2}(t))^2 \quad (4.2)$$

Where $\theta_{m1}(t)$ is the position of the Master 1 and $\theta_{m2}(t)$ is the position of the master 2. With the error between devices and the maximum allowed distance, it is possible to calculate the Virtual Boundary Torque.

4.1.2 Virtual Boundary Torque

This function is used when the trainee moves beyond the boundary, it will exert a virtual torque into the Adaptive Dominance factor in order to correct the trainee position. The torque is modeled by using the Hooke law as follows:

$$T_v = \begin{cases} 0, & d_{\theta_m} \leq B \\ -k_v(\theta_{m1} - \theta_{m2}) & d_{\theta_m} \geq B \end{cases} \quad (4.3)$$

Where k_v is a virtual stiffness, and equation 4.3 evaluates wherever the AAA function injects the virtual torque, only if the position error does not overpass the maximum allowed distance the virtual torque remains zero.

Once the virtual torque is calculated, we can establish the adaptive dominance factor.

4.1.3 Adaptive Dominance Factor

This last function is the one in charge of changing the global dominance of the system. It is based on a Gaussian function of the Virtual Boundary Torque T_v . The dominance is determined as follows:

$$\alpha_a = 1 - e^{-\frac{T_v^2}{2 \cdot (T_0)^2}} \quad (4.4)$$

Where T_0 is a task parameter, given by the trainer based on the complexity of the selected training, the range of α_a is from 0 to 1.

4.1.4 Issue with current AAA

The principal limitation of the AAA is the tuning process of four parameters (B_0 , T_0 , k_v and k_b), which can result in a complicated process for the trainer. B_0 is used to establish the maximum distance allowed for the trainer to the trainee; this distance must be determined according to the experience of the trainee and the performed task. T_0 is a variable tied to the type of training. k_v and k_b are variables that must be set to get the desired behavior of the boundary and the virtual torque. Out of these four parameters, the trainer specifies B_0 and T_0 based on the type of training, the remaining ones are set, based on the configuration of the environment and devices. Because of this, the tuning process can be challenging for the trainer depending on the selected task and used haptic devices.

4.2 Proposed eAAA

The training of medical students is varied due to the numerous surgeries that doctors must realize with expertise. Previous adaptive dominance factors (Shahbazi et al. [2013], Liu [2016]) base the change of the dominance on the analysis of the path generated by the trainer during the training session and applies a virtual fixture to limit the trainee movements. However, some training tasks can require a higher level of precision, or a novice student needs more freedom to start familiarizing herself/himself with the task.

To solve this, we need a new AAA function that can work by using only the maximum error, which is the error that indicates the system when a trainee has deviated enough from the desired position and gives the trainer the total dominance of the system. For the proposed function, we choose to implement geometric profiles that change the dominance in real-time. These profiles allow linear and non-linear behaviors for the dominance and are in function of the error determined by the euclidean norm ($\|e(t)\|$).

The profile selection is made by the trainer before the training session begins and is based on the difficulty of the task and the experience of the trainee. The only parameter that the trainer needs to provide for the profiles is the e_{max} , which is the maximum allowed error between Master 1 and Master 2. This maximum error limits how much the trainee can be far from the trainer before $\alpha = 1$. The first four proposed profiles are a line, a concave, a convex, and a Bézier function, all of them shown in Figure 4-1.

Using the profiles in the calculus of an adjustable α value allows delivering a soft change during the training task, as long as $e < e_{max}$. As the trainee directs the task, the dominance is adjusted,

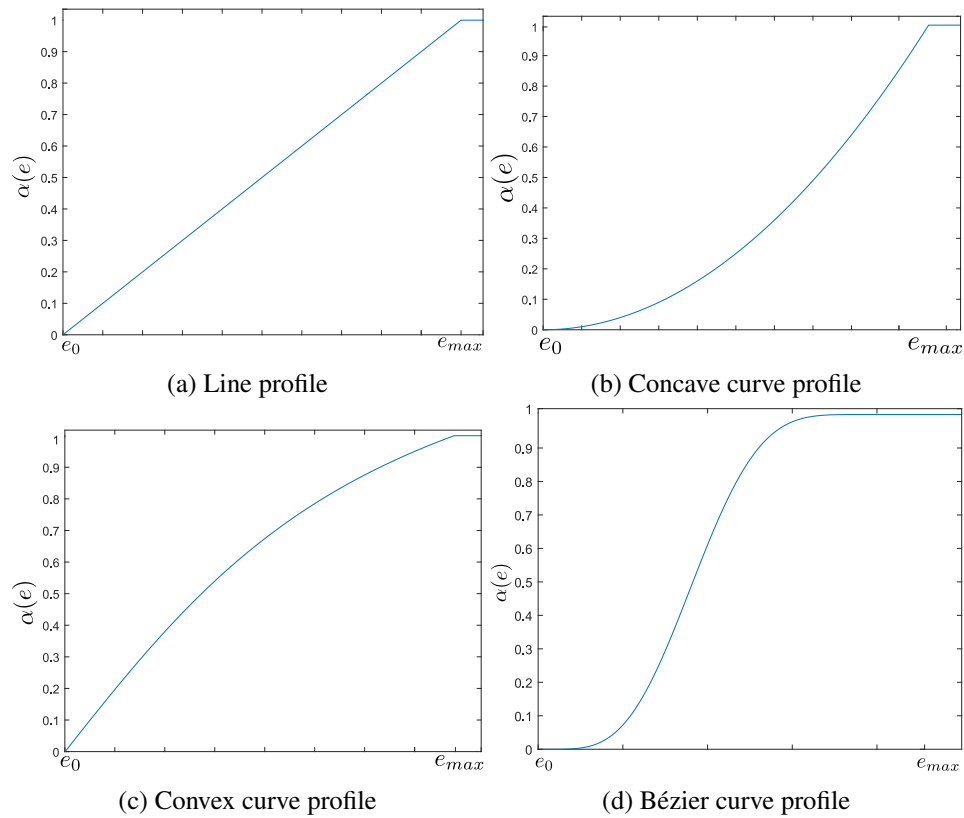


Figure 4-1 – eAAA default profiles

given more dominance to the trainer if it is necessary so he can apply corrections in real-time to the trainee's movements. On the other hand, when $|e| \geq e_{max}$, it is assumed that the error between Master 1 and Master 2 is big enough to allow the trainer to get the total dominance over the system to apply correctional movements.

More profiles can be implemented as well, but we consider that the concave, linear and convex profiles are enough for novice, intermediate and expert trainees, with the one dedicated case of the Bézier curve for daily practice.

4.2.1 Line profile

To determine the line profile shown in Figure 4-1a, the system first set the slope m as

$$m = \frac{1}{e_{max}} \quad (4.5)$$

Then the dominance α is set as

$$\alpha(e) = \begin{cases} m * e(t) & \text{if } e(t) < e_{max} \\ 1 & \text{otherwise} \end{cases} \quad (4.6)$$

As the dominance between the trainer and the trainee changes directly proportional to the error among them, this profile can be used for intermediate trainees, where the minimal change of the trajectory is reflected as an opposition force exerted by the trainer, indicating them that they need to correct the trajectory.

4.2.2 Concave curve profile

To determine the concave curve profile from Figure 4-1b, the same slope from equation (4.5) is used. Then, the dominance of α can be set as

$$\alpha(e) = \begin{cases} (m * e(t))^2 & \text{if } e(t) < e_{max} \\ 1 & \text{otherwise} \end{cases} \quad (4.7)$$

In this profile, the dominance changes at a slower rate than the linear profile, which brings to the trainee more freedom of movement during the training before they can feel the correctional force. This profile can be used for novice trainees, introducing them to the simulator and the training task.

4.2.3 Convex curve profile

The convex profile of Figure 4-1c, first we determine the slope of a tangential function as

$$m_t = \frac{\tan(1)}{e_{max}} \quad (4.8)$$

Then, the dominance α can be set as

$$\alpha(e) = \begin{cases} \arctan(e(t) * m_t(t)) & \text{if } e(t) < e_{max} \\ 1 & \text{otherwise} \end{cases} \quad (4.9)$$

The convex slope reacts faster than all the profiles. This profile can be used for an expert trainee, where the movement is very restrictive due to the sudden increase of torque.

4.2.4 Bézier curve profile

The Bézier curve profile of Figure 4-1d is a particular case selected because it is a junction of the three previously proposed profiles. The Bézier one begins acting like a concave curve, followed by a linear profile and finishes with a convex curve. This enables us to provide as more freedom to move for small position errors, a more corrective force for an intermediary error, and a very restrictive correctional force for a higher error. We assume that this profile can be used for continuous training and practice, where the trainees are familiarized with the task but do not have enough experience to face a linear or convex curve profile.

This Bézier curve was proposed by Beltran-Carbajal [2012], and it is used to give motors a soft change of velocity when they are controlled. This tenth-order polynomial curve is used because of the non-changing zone presented at the beginning and the end of the curve, allowing to give the trainee and the trainer an offset zone where both can move without disturbing the actual status of the dominance. This characteristic is very crucial since as mentioned before, it is a challenging task that trainers and trainees share the same position at the same time especially when our haptic devices have a nominal resolution of $55\mu\text{m}$. Also, the polynomial allows generating a continuous curve during all the segments until $e(t) = e_{max}$.

The dominance changes using the Bézier profile according to

$$\alpha = \begin{cases} b_z & e(t) < e_{max} \\ 1 & \text{otherwise} \end{cases} \quad (4.10)$$

Where b_z is

$$b_z = e_r(t)^5 [r_1 - r_2 * e_r(t) + r_3 * e_r(t)^2 - r_4 * e_r(t)^3 + r_5 * e_r(t)^4 - r_6 * e_r(t)^5] \quad (4.11)$$

And e_r is

$$e_r(t) = \frac{e(t)}{e_{max}} \quad (4.12)$$

With $r_1, r_2, r_3, r_4, r_5, r_6$ as constants with values of 252, 1050, 1800, 1575, 700 and 126 respectively.

4.3 Experimental validation

Using the eAAA algorithm, experimental validations to test the change of dominance were performed. The purpose is to prove that the system's dominance is changed according to the student's performance, based on the selected profile without any stability issue.

4.3.1 Correctional force validation

First, we wanted to check that the proposed profiles offer different changes of dominance and a correctional force behavior under the same scenario. To prove this, we used our Slave device from the core setup of section 3.4 and the eESC architecture in the Cartesian approach.

For this experiment only, we implemented a perfect trainer and trainee, as it is impossible to replicate the same trajectory in both haptic devices with human users. These two perfect users are calculated values of trajectories introduced directly to the eESC architecture as a position value. The trajectory is set as

$$\begin{aligned}x(n_t) &= r * \sin(n_t) \\y(n_t) &= r * \cos(n_t) \\z(n_t) &= r * \sin\left(\frac{n_t}{2}\right)\end{aligned}\tag{4.13}$$

With

$$n_t = \frac{2 * \pi * t}{t_t}\tag{4.14}$$

Where $t_t > 0$ and is the total time of the experiment and r the radius of the trajectory. The trajectory is a sloped circle used, so all the system axes are actuated during the test. The parameters of the experiment are $t_t = 25\text{s}$, $e_{max} = 1\text{cm}$ and $r = 6\text{cm}$. This experiment was repeated four times with the same parameters by only changing the used profile. In the beginning, the perfect trainer and the trainee follow the same trajectory, but during $5 < t < 15\text{s}$ the perfect trainee deviates from the trajectory by a 10% of the trainer's position. Figure 4-2 shows the described trajectory.

When a deviation occurs, the correctional force is injected into the Master 2 device as an effort to correct the position. Figure 4-3 shows the correctional force and the continuous change of α for each

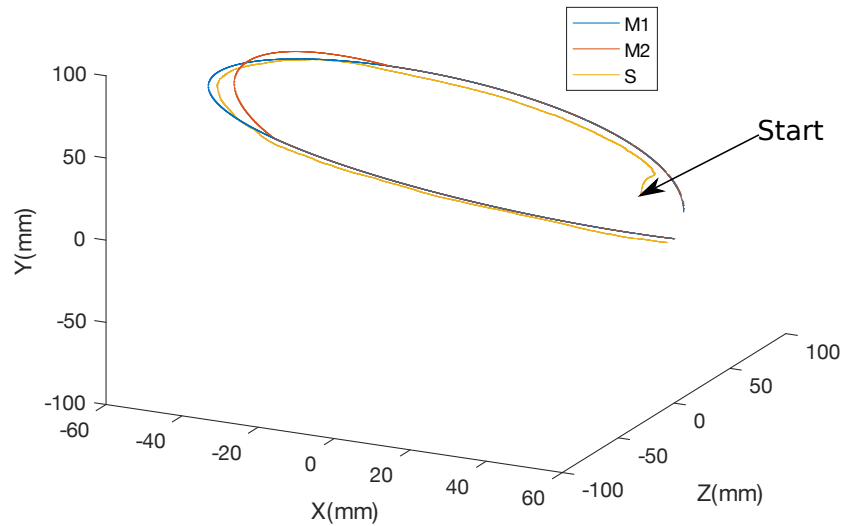


Figure 4-2 – Validation trajectory

profile.

One can observe that the convex curve profile reacts faster than the others in the change of the dominance, followed by the linear, Bézier and the concave curve profiles. However, the correctional force F for the Bézier curve raises in a more consistent way which provides the trainee in a smooth and continuous increase of the correctional force. In the case of the concave Curve profile, which reacts the slowest, the correctional force also changes more abruptly, so the trainee can feel this like a more sudden change of the correctional force. In the case of the convex curve and Line profiles, both increase the correctional force in a more continuous change than the convex curve but in a more authoritative way than the Bézier curve.

4.3.2 Free motion

We tested the eAAA with real users. We chose the Bézier curve profile due to the smooth change of dominance and correctional force it offers. We set $e_{max} = 3\text{mm}$, and used the core setup of section 3.4 and the eESC in Cartesian approach. The experiment was performed in free motion show the resulting position tracking and correctional force.

Master 2 was the leader ($\alpha = 0$) at the beginning. The trajectory was performed by the trainee until $t = 7\text{s}$ when the Master 1 started to correct the trainee's trajectory voluntarily. As the position between both users changed progressively, the dominance of the system did as well, resulting in the

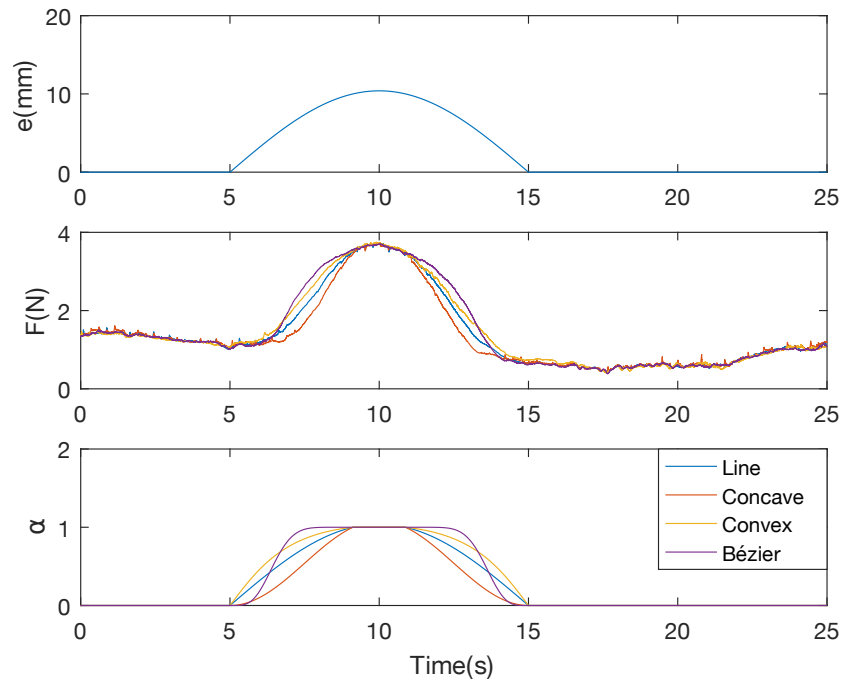


Figure 4-3 – Forces and dominance change for various profiles

evolution of the dominance α and consecutively in an increase of the force to correct the trainee's position. Figure 4-4 shows the position of the devices during the test and Figure 4-5 the correctional force of the haptic devices. As soon as $|e(t)| > e_{max}$, the dominance reached its top value of 1 and remained so until the trainee corrected the position of her/his device, to equal the trainer's one as of $t = 17\text{s}$. Then the system returned the entire dominance to the trainee, and the trajectory ended.

Fig. 4-4 shows the positions of the devices. From $0 < t < 9$ and $18 < t < 24$, the RMS errors are 1.66mm and 1.75mm for M1-M2, in the case of M2-S the RMS are 1.86mm and 2.42mm for M2-S. As the position difference between trainer and trainee for these two periods of time is negligible, the dominance does not change significantly $\alpha \rightarrow 0$.

When $9 < t < 18$ the trainee deviated from the trainer's position. As the position of Master 2 differed from Master 1, the value of α raised to 1. As the dominance changed the system exerted a correctional force (in the three axes) on the trainee's device, as visible in Fig. 4-5.

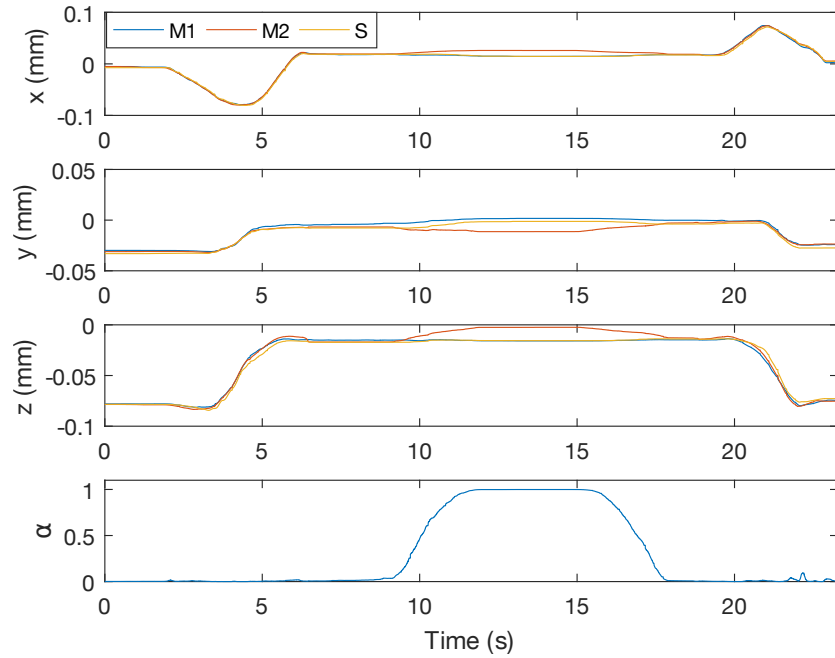


Figure 4-4 – AAA position tracking test: positions

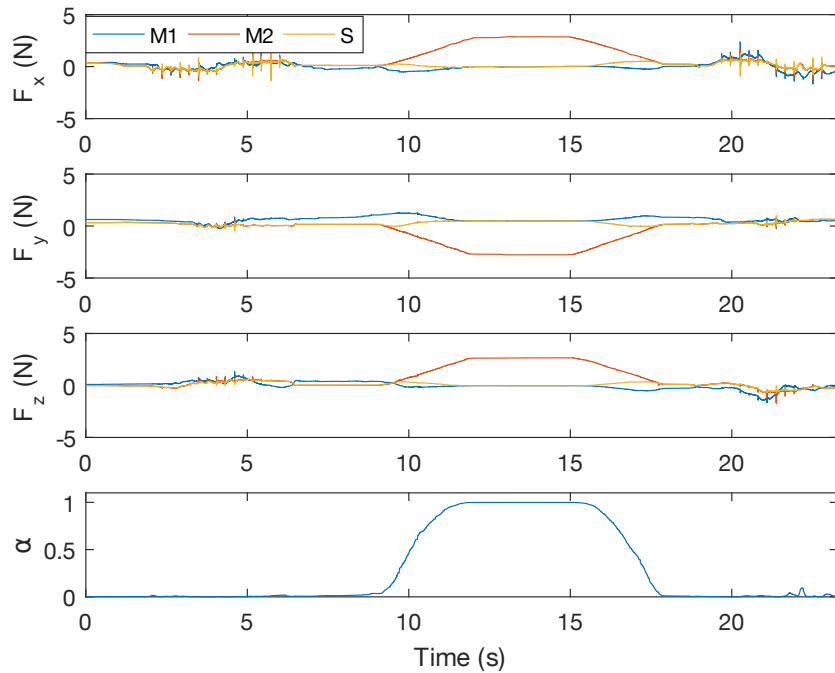


Figure 4-5 – AAA position tracking test: forces

4.4 Adaptive dominance validation

Albeit the proposed AAA methods are introduced to improve the trainee learning by adapting the dominance based on their performance, there is still no method or study that proves this hypothesis. Furthermore, there is no way to test and compare the different existing AAA methods. Because of this, it is necessary to establish a test bench to analyze the advances of the trainee using an AAA method vs a just training.

4.4.1 Exercise selection

To validate the improvements of an AAA method vs just training, we proposed a series of real exercises that were performed by test subjects. Two different groups were made and instructed to replicate the same exercises, with the only difference of how they were instructed to do it. The exercises consist of real tasks rather than virtual ones, which made them easy to perform. The principal drawback of the attempts is the lower repeatability, as users cannot repeat the process the same way, which complicates the analysis of data.

To choose the exercises implemented in the study, we first brainstormed possible options that could be performed by the trainee following the instructions of the trainer. The exercise must fulfill two principal characteristics to be considered: the exercise must provide useful force feedback to the user, so she/he can feel like she/he interacts directly with the environment. The exercise must give the sensation to trainees that they are actually modifying the environment and not just touching it. Based on these considerations, eight exercises were proposed:

Surgical game: based on the popular board game, Operation, the trainee tries to recover a piece of plastic, representing an organ, by using a gripper in the slave device in the fastest way possible. The trainee must avoid touching the edges of a hole containing an organ.

Tie a knot: the trainee tries to complete a knot using the slave's tool in the less time possible.

Dictate Dimensions: the trainer receives instructions to draw a 3D object.

Cutting task: the trainee cut an object following the instructions of the trainer.

Following a path: the trainee must guide a hook from point A to B, following the profile of the winding metal path. The hook must not touch the path, or a sound alarm will be activated.

Building a Jenga: based on the popular board game, Jenga, the trainee must move bricks from a blocky tower, avoiding collapsing it.

Writing: the trainee tries to write non-Latin characters instructed by the trainer. The trainee must replicate the trainer’s character as close as possible.

Stiching: the trainee must follow a given pattern from the trainer in the fastest way possible.

To choose the best-suited exercise for the study we quantitative qualified them by analyzing each exercise regarding the general constraints of the simulator like general feasibility and motion capabilities. Table 4.1 shows the score for general constraints and feasibility, table 4.2 for the motion capabilities and interaction responses. The score of both tables is summarized in Table 4.3.

Exercise	General constraint				General feasibility		
	Slave’s Workspace	Environment	<3N Force	Total	Performable	Expert availability	Total
Surgical game	1	1	1	3	1	-1	0
Tieing a knot	1	-1	1	1	1	1	2
Dictating dimensions	1	1	1	3	1	-1	0
Cutting task	1	1	-1	1	1	-1	0
Following a path	1	1	1	3	1	-1	0
Building Jenga	1	1	1	3	1	-1	0
Writing	1	1	-1	1	1	1	2
Stitching	1	0	1	2	1	1	2

Table 4.1 – General Constraint and General Feasibility criteria for proposed exercises

For General constraints, the criteria are slave’s workspace, which determines if the exercise can be done inside the physical limitations of the slave. The environment determines if the exercise interacts directly with an object that can be modified or displaced to get haptic force feedback. “<3N Force” determines if the exercise is suitable for a maximum force of 3N, which is the top force that the Slave can offer. For general feasibility, the criteria is “Performable” which indicates if the exercise can be done with the current setup, and expert availability, which determines if a user who is considered an expert in the exercise, can be used to work as the trainer.

In the case of Motion and Interaction, the two criteria are “Quantifiable”, which means if the metrics of the exercise can be measured, and “Improves” which determines if the user can improve directly from the exercise.

From the analysis of table 4.3, the best score exercises are “Writing and Stiching”. However, after an analysis of the setup from section 3.4, the lack of an actuated wrist in the master and slave devices limits both activities. Starting from the previous analysis, we proposed a second wave of exercises,

Exercise	Motion (position)			Interaction (Force)		
	Quantifiable	Improves	Total	Quantifiable	Improves	Total
Surgical game	1	1	2	1	-1	0
Tieing a knot	-1	-1	-2	1	1	2
Dictating dimensions	1	1	2	-1	-1	-2
Cutting task	1	1	2	1	1	2
Following a path	1	1	2	-1	-1	-2
Building Jenga	-1	-1	-2	1	-1	0
Writing	1	1	2	1	1	2
Stitching	1	1	2	1	1	2

Table 4.2 – Motion and Interaction criteria for proposed exercises

Exercise	Final Score
Surgical game	5
Tieing a knot	3
Dictating dimensions	3
Cutting task	5
Following a path	3
Building Jenga	1
Writing	7
Stitching	8

Table 4.3 – Final score for proposed exercises

that was simpler to fit better with the slave’s workspace and actuators’ limitations. Table 4.4 shows the criteria of “General constraints” and “General feasibility” for the new set of exercises. The exercises are not explained as the previous ones because of their simpler nature and self-explanatory names.

Exercise	General constraint				General feasibility		
	Slave’s Workspace	Environment	<3N Force	Total	Performable	Expert availability	Total
Push an object (certain distance)	1	1	1	3	1	-1	0
Pull a spring (certain distance)	1	1	1	3	1	-1	0
Pick up something to a point	1	1	1	3	1	-1	0
Scratch an object	1	1	1	3	1	-1	0

Table 4.4 – General Constraint and General Feasibility criteria for proposed exercises: Second wave

Table 4.5 shows the criteria of Motion and Interaction for the second wave of the proposed exercises.

Finally table 4.6 shows the final score for the exercises proposed in the second wave.

The best scored exercises were “Push an object” and “Pull a spring”. To simplify the analysis of data and discussion of the results, the push exercise is focused in the localization of the slave’s tool in the environment, while the pull exercise focuses on the replication of the force feedback from the

Exercise	Motion (position)			Interaction (Force)		
	Quantifiable	Improves	Total	Quantifiable	Improves	Total
Push an object (certain distance)	1	1	2	1	1	2
Pull a spring (certain distance)	1	1	2	1	1	2
Pick up something to a point	1	1	2	-1	-1	-2
Scratch an object	-1	-1	-2	1	1	2

Table 4.5 – Motion and Interaction criteria for proposed exercises: Second Wave

Exercise	Final Score
Push an object (certain distance)	4
Pull a spring (certain distance)	4
Pick up something to a point	0
Scratch an object	0

Table 4.6 – Final score for proposed exercises: Second Wave

environment.

4.4.2 Validation

Test bench setup

First, we established to use the base core test bench of section 3.4. In this case, each participant is a trainee, and the teacher is one of our associates that gives the participants all the necessary instructions and explains how to perform. Due to the simpler nature of the exercises, an expert trainer is not required for both exercises. A wall divides the trainee and the trainer spaces to avoid direct interaction. The trainer has direct visual feedback of the slave while the trainee receives it from a stream of video on their screen. The slave interacts with two elements, a block that is used to evaluate the push, and a spring to evaluate the pull. Figure 4-6 shows the configuration of the setup used.

The push exercise's objective is to guide the block over a slope until a certain distance to evaluate the dexterity of a trainee during the location in the Cartesian space. The trainer guided the trainee into pushing the block over the slope and stopped at a marked landmark. When the block is returned to the original position, the trainer gives the dominance to the trainee. The trainee must replicate the exercise by stopping as close as possible of the landmark.

The pull exercise's objective is to instruct the trainee on how to pull a spring until a certain distance and angle to evaluate the dexterity and kinesthetic memory of the trainee. The trainer taught the trainee how to pull the spring until a specified landmark and then asked him to replicate the exercise as close

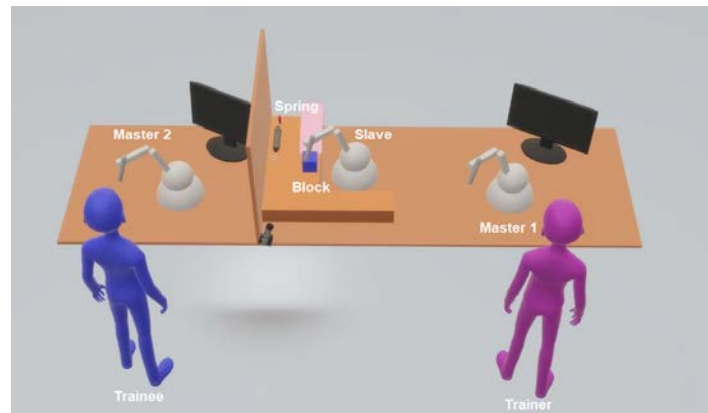
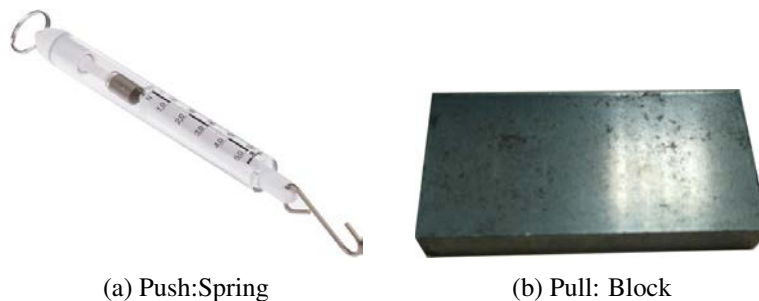


Figure 4-6 – AAA validation setup

as possible. The traversed distance of the spring is the equivalent of 1N exerted by it. The reason for using 1N is because the Geomagic 3D Touch supports up to 3N, and by using this maximum force, the trainee could saturate the actuators quickly, and would not feel any difference on the force feedback after this point. By using 1N as a reference, the trainee must recall how much effort she/he needs put on, to get the same haptic feeling. The angle of 30° is chosen, so the user applies different forces in each axis, making the task a little more challenging rather than just applying the force in one axis or the same in both.

For the environmental objects used during the test, we selected for the pull exercise an RS spring model 27003, which can measure up to 5N of force (Figure 4-7a) and it is fixed to the bench. For the push exercise we used a 6x3x0.8cm cold roll block with a weight of 100g (Figure(4-7b)) that can move over a 45° slope.



(a) Push: Spring

(b) Pull: Block

Figure 4-7 – Environment interaction object

Protocol

For the test, 18 subjects were contacted by one of our associates. The subjects consist of a varied group from different nationalities, ethnicity, and languages (although the test was directed in English). They were divided into two different groups for comparison. Group 1 (AAA group) and Group 2 (Lead group). Group 1 served to validate if the AAA function improves the trainee performance and learning. We used the eAAA with a Bézier profile. The process consisted of apprenticeships (demonstration mode), practiced with the trainer (guidance mode) and then evaluated (evaluation mode). For Group 2, the group who was used to compare results vs Group 1, and their process was taught (demonstration mode) and then evaluated (evaluation mode).

All participants were cited to the Laboratory Ampère with a lapse of 30 minutes between participants, of which 15 minutes were for performing the exercises, with 5 minutes to respond to a survey (Appendix D) and 5 minutes to attend unexpected problems that could occur during the test. Participants were introduced to the test bench and the haptic devices. They were explained about how the simulator works and the final proposal of the study. Also, they were indicated how to use the user interface to compare the same position and force. Depending on the group, participants, performed the exercise in the following way.

For Group 1:

- The trainer carried out the push and pull exercise three times in demonstration mode.
- The trainer changed the dominance to guided mode
- The trainer replicated the exercise three more times and was corrected by the Trainer if necessary.
- The trainer changed the dominance to evaluation mode
- The trainee performed three times more the exercises, but without the intervention of the Trainer

For Group 2:

- The trainer carried out the push and pull exercise 3 times in demonstration mode.
- The trainer changed the dominance to evaluation mode
- The trainee performed six times more the exercises, but without the intervention of the Trainer

It should be noticed that for evaluation mode, for both groups, the user interfaces disabled the force indicators and the position of the Master devices. This is to prevent trainees from replicating the force based on the indications from the screen rather than on their own kinesthetic memory.

Results

For recorded data, we use the position values provided by OpenHaptics and the calculated force of the IPC controls, because of this, the computation force from the IPCs can rise over 3N, which is the saturation force of the Geomagic Touch.

With the recorded data from the tests, we determine the difference between trainee and the landmark (Δ_d), for the push exercise, and the force and angle applied in the pull exercise. Table 4.7 summarizes the position and force difference for push and pull in Group 1 and 2.

Group 1	Push	Pull		Group 2	Push	Pull	
Participant	$\Delta_d(\text{mm})$	$F(\text{N})$	θ_f°	Participant	$\Delta_d(\text{mm})$	$F(\text{N})$	θ_f°
1	0.03	2.13	27.78	2	0.04	2.26	18.45
3	0.03	2.64	22.24	4	0.03	1.74	26.79
5	0.02	1.61	21.88	6	0.02	1.35	29.71
7	0.03	3.55	19.59	8	0.03	3.78	18.82
9	0.02	2.23	26.13	10	0.03	0.85	118.23
11	0.03	1.60	34.62	12	0.03	2.95	6.20
13	0.03	6.53	19.79	14	0.03	0.77	43.83
15	0.03	2.07	14.64	16	0.03	0.37	125.92
17	0.03	1.75	25.40	18	0.03	3.10	16.76
Total mean	0.03	2.68	23.56	Total mean	0.02	2.28	22.93

Table 4.7 – Push and Pull results

It should be noted that participants 10 and 16 were eliminated because of inconsistency in their results, specifically in the θ_f . The survey and the answers of the participants are detailed in Appendix D.

For the push exercise, the results of Δ_d for group 1 and 2, show only a variation of 4% between them, which (considering the low accuracy that trainees have due to their inexperience with haptic systems and the performed task) can be considered as negligible. So, from these tests, we conclude that by the moment there is no significant improvement using an AAA method.

In the case of the pull exercise, we see a variation of 268% for Group 1 and 228% for Group 2 in the force exerted. It appears that Group 2 has better performance overall. However, due to the high deviation from the desired force of 1N, the results for the two groups do not permit us to conclude that the eAAA improves the learning of the trainee. Finally, for the θ_f , the variation is 21% for Group

1 and 23% for Group 2. The slight difference of 2% between both Groups also does not allow us to conclude an improvement or drawback of using an AAA method.

4.5 Results and discussion

The Bézier curve offers a starting point to prove the eAAA, but it can not establish that the function is the optimal one. More test needs to be made, where the Bézier polynomial is compared versus other functions to determine a more qualitative value.

We still have to test these four profiles with real trainees to determine which of those perform better in different approaches of dexterity. Unfortunately, we still can not determine if eAAA performs better than the other methods proposed in the bibliography (Shahbazi et al. [2013] and Liu et al. [2015]) as the parameters of those changes the behavior of the dominance. This can be solved by making a series of experiments with trainers and trainees, where the evolution of several test groups are analyzed to determine which method and profile for our case allows the student to improve faster.

We tried to validate the learning improvement brought by eAAA. Unfortunately, our tests do not offer conclusive results, as the variations between both test groups are almost insignificant. We consider that due to the low number of repetitions and the lack of time, trainees do not adapt and get used to the task. As remarked by Vaughan et al. [2016], in a medical procedure like epidural insertion, it is required at least eighty repetitions to consider the trainee an experienced student. Although the task does not require so much training in our test, it provides a starting point of how much training is needed to notice a difference in learning improvements. For future work and correct validation, two long-term study groups are required to corroborate the potential improvements and drawbacks of using an AAA.

Our proposed testbed for validation of the AAA function must be improved to include more sophisticated tasks to get significant results about learning improvement, as the push and pull experiments only provide simple actions that are easy to master. For more complicated tasks, we need to provide wrist force feedback to the system. By changing the haptic device into a 6 actuated DoF, like the Virtuouse 6D Desktop, and implementing the eESC architecture onto all the joints.

For a more faithful validation of eAAA and the ESC architecture, experiments including a test group should be performed on a real surgical hands-on training to be able to determine which one is the best between single-user, dual-user, and manual training.

Chapter 5

Expanding eESC to $m - 1$ trainees

5.1 Problem

Based on the literature review from section 2.4.5, we determined that none of the proposed architecture allows several trainees to be instructed by one trainer at the same time. By making a simulator that can perform this, it is possible to reduce the time that takes every trainee to get instructed in Demonstration mode. This also allows students to present their advances to several trainers at the same time (in evaluation mode) and get feedback from all of them. The main problem with a multi-trainee architecture is to find a correct configuration that enables this.

The difficulty of adapting eESC to a multi-trainee approach is the proposal of a new set of MIMO¹ nodes that ensures the stability and transparency of the system.

5.2 Expansion

In a dual-user scheme, it is supposed that all the devices are linked to each other, but as it is stated in section 3.1, ESC only connects Master devices with the Slave. Leaving the Slave device as an intermediary point between Master 1 and Master 2 has the drawback of a potential accumulative error between them, but this also opens the possibility of expanding the architecture to more users.

Expanding the eESC architecture to an $m - 1$ trainee system allows for the development of new simulators that support several trainees, receiving the same guidance at the same time. The main perspective is to provide one device for one trainer (Master 1) and $m - 1$ devices for the $m - 1$ trainees

¹Multiple Input Multiple Output

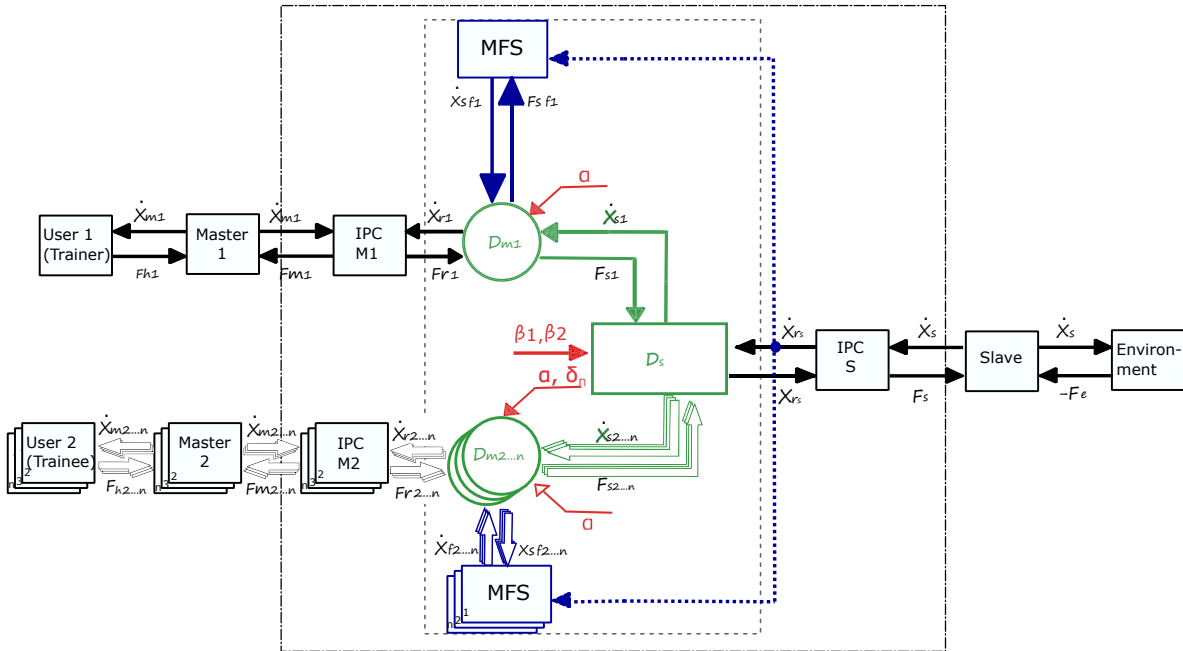


Figure 5-1 – ESC expanded to m devices for only one dof (Cartesian example)

(Master 2 to Master m).

The ESC architecture interconnects the haptic systems, and the controllers using Multiple-Input-Multiple-Output (MIMO) nodes D_{m1} for Master 1, D_{m2} for Master 2 and D_s for the Slave. During the expansion of the ESC architecture, the node D_{m1} remains unaltered, as it is intended that Master 1 will always be the device used by the trainer. The node D_s serves as a union point between the implicated devices and the Slave, this connects directly to D_{m1} and D_{m2} . Expanding the node D_s allows us to use it as a union point between the trainees' devices and the Slave. For this, first, it is necessary to modify the node D_{m2} and duplicate it in parallel with their corresponding IPC controller. It is also necessary to update D_{m2} and D_s in a way that allows the introduction of the $m - 1$ complementary devices. Figure 5-1 shows the expansion of the ESC architecture, augmenting the number of nodes.

The nodes D_{m2} to D_{mm} are identical and allow the connections between each trainee device and their corresponding IPC controller to the node D_s (eq. 3.1).

Each trainee device is identified from Master 2 to Master m , and each trainee device works in parallel to the others. As D_s is the node that interconnects all the IPC controllers and establishes the dominance factor of the system, it is necessary to update it to be used with $m - 1$ new devices. D_s reuses the factors β_1 and β_2 introduced in Eq. (3.2) in order to avoid the appearance of α^2 terms in the relations between angular velocities and torques of devices. The principal restriction of the expansion

is the preservation of the transparency matrix presented in Eq. (3.11). To achieve this, the matrix must be skew-symmetric while adding the m input forces and output velocities from the master devices.

The expansion of node D_s is set as:

$$\begin{bmatrix} \dot{\theta}_{s1} \\ \dot{\theta}_{s2} \\ \vdots \\ \dot{\theta}_{sn} \\ T_{rs} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 & \beta_1 \\ 0 & 0 & \cdots & 0 & 1 - \beta_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 - \beta_2 \\ -\beta_1 & \beta_2 - 1 & \cdots & \beta_2 - 1 & 0 \end{bmatrix} \begin{bmatrix} T_{s1} \\ T_{s2} \\ \vdots \\ T_{sm} \\ \dot{\theta}_{rs} \end{bmatrix} \quad (5.1)$$

where D_s remains a skew-symmetric matrix and thus preserving the transparency property shown in section 3.2.2. $\dot{\theta}_{s1..n}$ are the angular velocity outputs from the Slave towards all master IPC respectively, $T_{s1..n}$ are the torques transmitted by each master IPC to the slave IPC, and T_{rs} is the torque transmitted, to the slave IPC, taking into account the dominance factor.

In the new architecture dealing with $m - 1$ trainees devices, the factor α cannot establish the dominance for all the trainee devices at once. As α indicates the grade of interaction which coexists between the trainer and the trainees, we have to consider it as the **global dominance factor**. It is desirable that only one trainee, the active trainee, interacts at the same time with the trainer and the slave in evaluation and guidance mode, to avoid possible interposition input problems. The remaining trainee devices can only receive haptic feedback from the Slave.

The selection of the active trainee is manually established in real-time by the trainer. To provide this option a new trainee selection mechanism has been introduced in conjunction with α : $\Delta \in [2, m]$. This discrete variable establishes which of the trainee devices interacts with the Slave and the trainer. Figure 5-2 illustrates how the new hierarchy of dominance works.

Δ has been split in $(m - 1)$ boolean variables δ_i whose value is Kronocker delta provided by Eq. (5.2), in order to integrate them in each D_{mi} node (initially D_{m2} in the initial dual-user architecture), with $i \in [2, m]$.

$$\delta_i = \begin{cases} 1 & \Delta = i \\ 0 & \text{otherwise} \end{cases} \quad (5.2)$$

When $\delta_i = 0$ the i master device receives haptic feedback, and it is positioned according to the slave device, whatever value of α . When $\delta_i = 1$, i the master device has a dominance established by

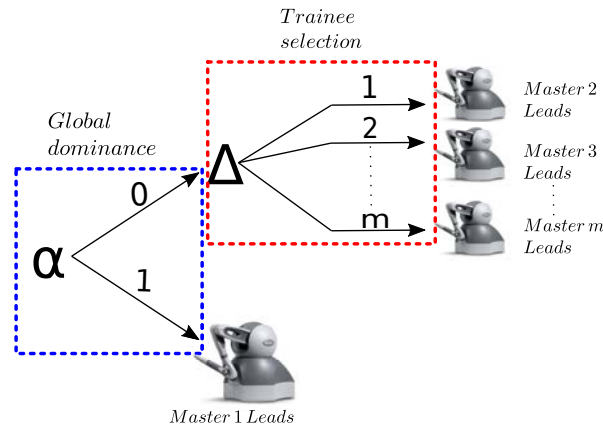


Figure 5-2 – Dominance hierarchy

α .

With D_s set, new D_{m2} to D_{mn} nodes must be created in parallel. Equation 5.3 shows the skew-symmetric matrices established for the D_{mi} nodes.

$$\begin{bmatrix} \dot{\theta}_{ri} \\ T_{si} \\ T_{sfi} \end{bmatrix} = \begin{bmatrix} 0 & f_{s1} & f_{s2} \\ -f_{s1} & 0 & 0 \\ -f_{s2} & 0 & 0 \end{bmatrix} \begin{bmatrix} T_{ri} \\ \dot{\theta}_{si} \\ \dot{\theta}_{sfi} \end{bmatrix} \quad (5.3)$$

where θ_{sfi} is the angular velocity injected by the MFS ² i into the trainee IPC i , T_{sfi} is the torque provided by the trainee IPC i to the MFS i , and θ_{ri} is the angular velocity transmitted to the trainee IPC i , affected by the dominance computation on θ_{si} and θ_{sfi} . f_{s1} and f_{s2} are used to simplify the matrix notation, and their values are set by equations 5.4 and 5.5

$$f_{s1} = (1 - \alpha)\delta_n \quad (5.4)$$

$$f_{s2} = 1 + \delta_i(\alpha - 1) \quad (5.5)$$

Each input/output of torque and angular velocity is applied consecutively in the expanded matrix D_s of equation 5.1.

The passivity (and so its stability) of the whole system is derived from the original architecture where a Time-Domain Passivity Observer monitored the energy generated by all the MFS in order

²Modulated Flow Sources. Introduced here in order to provide, on the one hand, force feedback and guidance to users for which $\alpha > 0$, and on the other hand, to guarantee the passivity of the system, see Liu et al. [2015]

to avoid any extra energy injection which may compromise the passivity and destabilize the system. Equation 5.6 shows how the energy generated by each MSF E_{pi} is computed.

$$E_{pi}(t) = \int_0^t \alpha \delta_i T_{ri} \dot{\theta}_{rs} dt \quad (5.6)$$

The passivity observer uses E_{pi} in equation 5.7 to determine the real-time value of $\dot{\theta}_{sfi}$, to ensure the preservation of passivity and the haptic feedback as long as the passivity is not compromised.

$$\dot{\theta}_{sfi} = \begin{cases} \theta_{rs}(t) & \text{if } E_{pi}(t) \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad (5.7)$$

5.3 Experimental results

5.3.1 New setup

To validate the extension of eESC into a multi-trainee approach, experiments were performed. For this, it was necessary to add more haptic devices into the core test bench, but as we only have access to three real devices, one virtual haptic device based on the model shown in the appendix A.1.4 was created, and it is used as the Master 1 device.

Indeed, the user who is manipulating (for whom $\delta_i = 1$) operates the Slave which follows her/his gestures. The Slave feeds back the interaction force between its tip and its environment. The other user devices reproduce the slave motion to guide these users. When a slave-environment interaction takes place, the follower users are asked to move their device to the same position as the leader device in order to feel the same force feedback and learn how to dose this force in a particular training context. Note that in all the plots, the notation (v) next to the device name represents the use of the virtual device.

5.3.2 Free motion

This experiment was performed to validate the position tracking between all devices, in free motion, and whatever value of the dominance factor α and the trainee selector Δ . All the follower devices must follow the leader's path, which is the one who has major dominance. For this experiment, we put the system in evaluation mode ($\alpha = 0$). During the task, the value of Δ is changed from 1 to 2 to

verify that the change of dominance between trainee devices does not compromise the tracking task. The chosen trajectory was a helical path performed by the trainee as consistent as possible, which allows us to demonstrate the position tracking in all the degrees of freedom. Figure 5-3 shows the path performed in the Cartesian space.

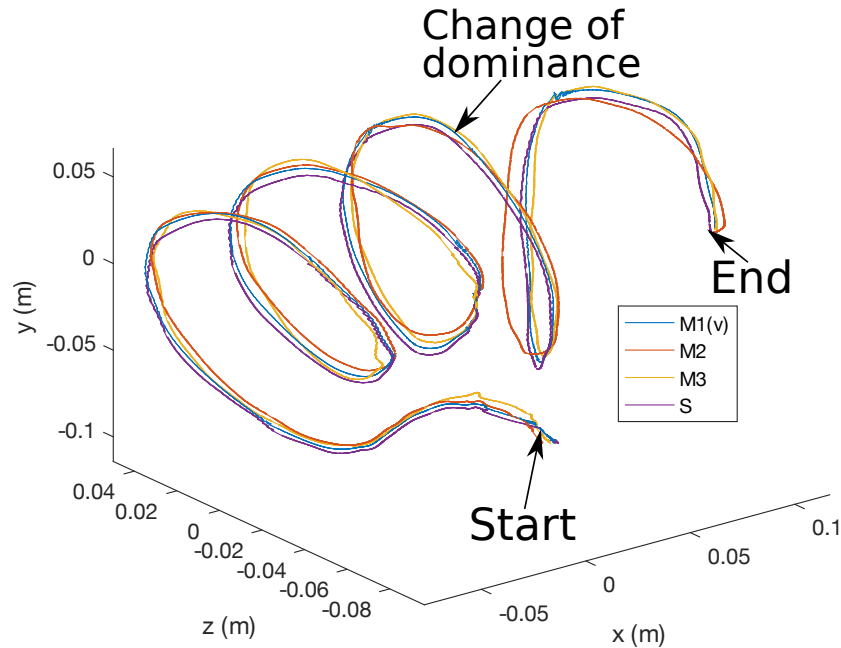


Figure 5-3 – Expansion m-trainee: Trajectory in free motion

In the beginning, $\Delta = 1$ so Master 2 (M2) is the leader device. At $t = 5s$ there is change of dominance: $\Delta = 2$ and Master 3 (M3) becomes leader. Figure 5-4 shows the path tracking for each Cartesian axis. It is important to highlight that the path tracking is not affected by the change of dominance: at $t = 5s$, M3 becomes the leader and continues the motion in the same way that M2. All the remaining devices continue the position tracking with this new reference.

To evaluate the performance of the position tracking, the RMS tracking errors between the Cartesian positions of the devices were computed. Table 5.1 shows these values when M2 and M3 are the respective leaders. This table clearly shows that the RMS errors between the leader device and the other devices are smaller than between other devices. For instance, the error between M2 and Slave (when M2 is the leader for $t \leq 5s$) is smaller than the one between M2 and the Slave (S) when it is a follower ($t > 5s$).

The position tracking errors between the slave and follower devices depend on the reaction of the users which may voluntarily resist the guidance of their device and artificially raise this error.

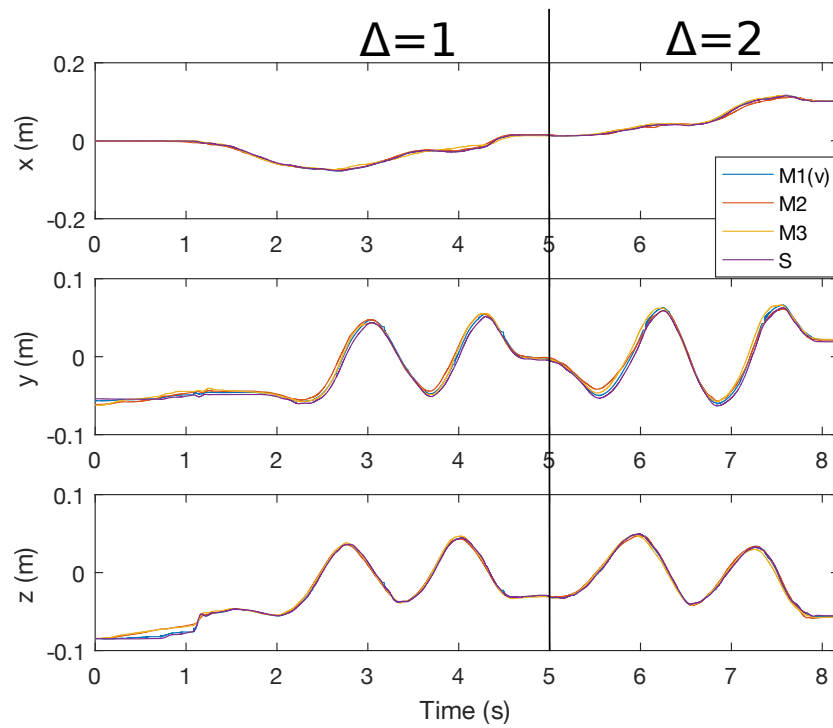
Figure 5-4 – Expansion m-trainee: Position with $\alpha = 0$ in free motion

Table 5.1 – RMS error in free motion

Error [mm]	M2 Leader ($t < 5$ s)	M3 Leader ($t \geq 5$ s)
M2-M1(v)	2.7	4.9
M2-M3	2.5	5.1
M2-S	4.2	5.1
M3-M1(v)	3.5	1.4
M3-S	4.5	3.5

The Cartesian RMS errors between each leader and the Slave are around 4 mm. With this kind of off-the-shelf low bracket haptic device, we estimate that this is a good result.

5.3.3 Force tracking

Once position tracking has been validated, we checked that the force tracking was accurate. This test was performed similarly to the one in section 3.5.3, with Δ changing from 1 to 2 during the task to prove the torque tracking of all devices. In the beginning, $\Delta = 1$ and Master 2 controls the Slave which collides with a surface located at $x = 9$ cm. At this moment, every follower device is guided up to the position of the wall. In order to feel the same force feedback as the leader user, follower users are asked to position their device in the same position as the leader master (displayed to each follower on

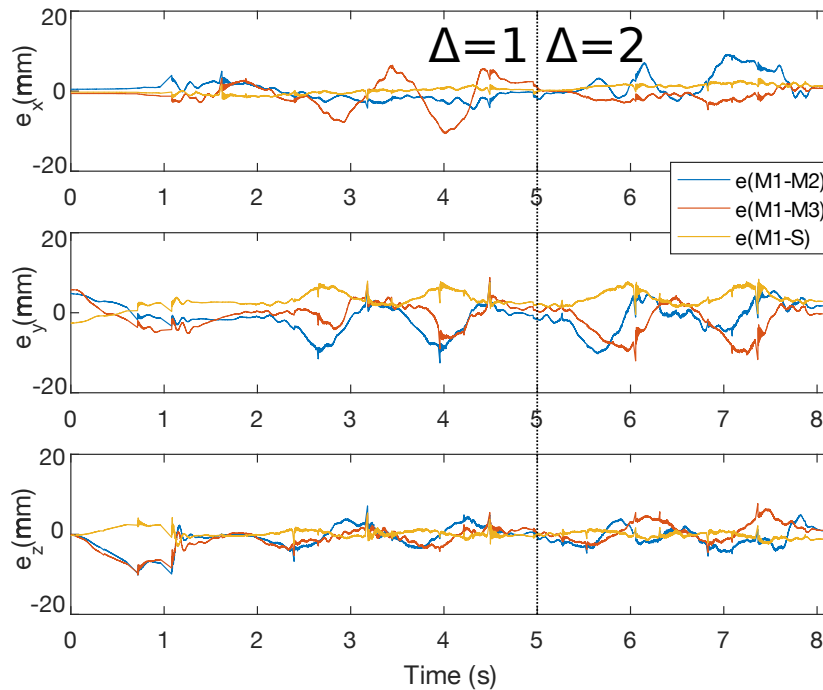


Figure 5-5 – Expansion m-trainee: Position errors with $\alpha = 0$ in free motion

their screen).

After a few seconds of getting the same haptic feedback, Master 2 goes back and leads the Slave out of contact with the environment. The dominance is then changed to $\Delta = 2$, which makes Master 3 become the leader. Master 3 reiterates the same procedure. Figures 5-6, 5-7 and 5-8 show the position and force tracking of the experiment. At $t = 25$ s the change of dominance was performed.

These figures highlight that, when the Slave collides a surface, the Master 2 user forces his device deeper into the wall to get haptic feedback. Once Master 2 stops, at $t = 7$ s Master 3 and Master 1 users, begin to exert effort to equal the position of the leader. As a response to this, both forces increase in Master 2 and Master 3. At $t = 11$ s, all the devices share almost the same position and force feedback. After 5 s, Master 2's user withdraws the Slave from the wall being followed by Master 1 and Master 2. At $t = 25$ s the change of leader is effectuated, and now Master 3 is the leader. The process is repeated to verify that all the devices can exert force tracking.

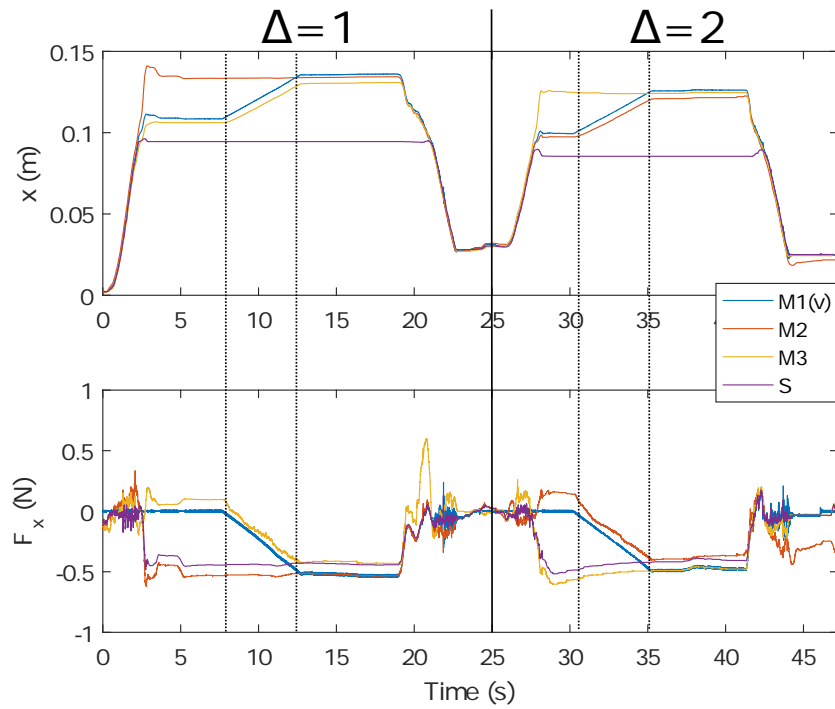


Figure 5-6 – Expansion m-trainee: Position and force in axis x in force tracking

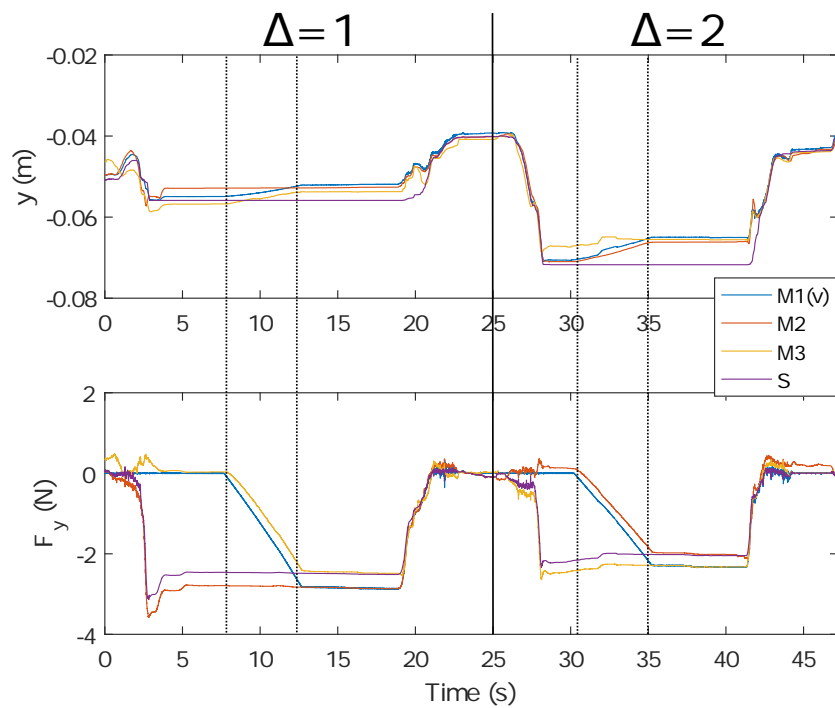


Figure 5-7 – Expansion m-trainee: Position and force in axis y in force tracking

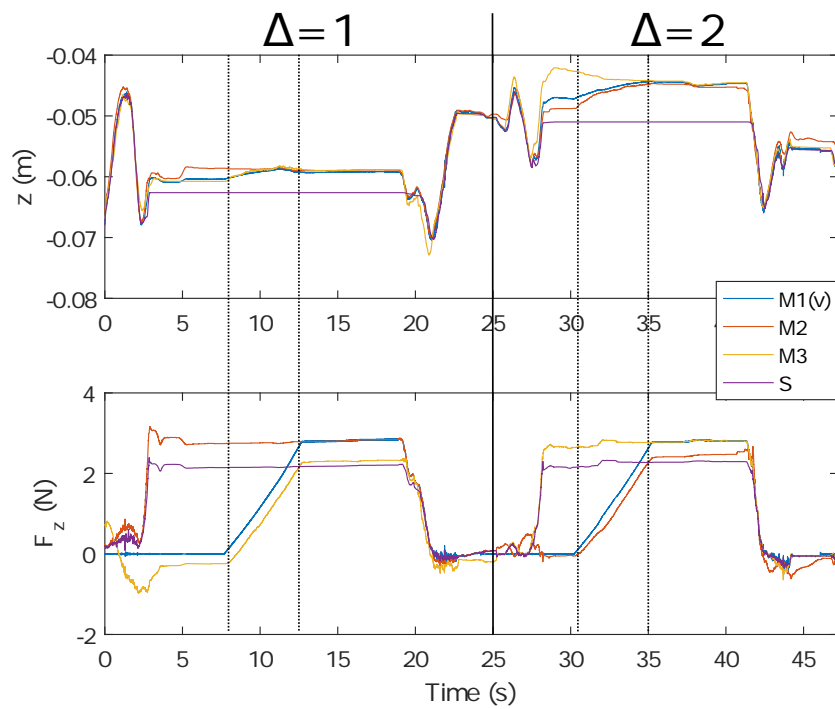


Figure 5-8 – Expansion m-trainee: Position and force in axis z in force tracking

5.4 Conclusion

The inherited configuration of the original ESC architecture, allows eESC to be expanded into a multi trainee approach without compromising the passivity of the system. With this architecture, it is possible to perform parallel training (in demonstration mode) between the medical students, as a trainer could teach m trainees, where the only limit is the number of available haptic devices. The trainer must determine the trainee that shares the dominance with him, while the rest only gets haptic feedback. This system can also be implemented in evaluation mode, where a student can perform a test to several medical evaluators and get qualitative feedback from them.

In the case of the guidance mode, the trainer can also establish with which trainee he/she wants to share the dominance. For example, during a session, the trainer can correct the trainee in real-time and give the other trainees a caution of what to avoid before committing the mistake. On the other hand, the trainees can train with their classmates, under the supervision of the trainer. Yet, it is not proved that this configuration is effective in terms of training efficiency. More tests in this matter could help determine this.

Acknowledgments

The use of computed based simulators for medical training is a recent paradigm for teaching complicated procedures. They present advantages over classical methods like mannequins and corpses due to its ability to repeat a process several times without getting short in supplies and the ability to get precise information about the performance of trainees for objective assessment purposes. Nowadays, there are several simulators on the market, but the vast majority do not provide a dual-user approach. With a dual-user approach, students can directly interact with the expert, or trainer, and with the environment. The trainer can lead the trainee and show her/him how to perform the task while the latter receives haptic feedback in addition to the trainer's commentaries and suggestions. The architectures exposed in section 2.4.5 do not satisfy the requirements, so we improved the ESC architecture introduced by Fei Liu during his Ph.D.

The first modification to the ESC architecture was the expansion into an n-DoF architecture. The expansion was performed by duplicating ESC for each degree of freedom. We proposed a new method to calculate the IPC controller based on experimental results.

The transparency analysis of Chapter 3 demonstrates that the Impedance-Admittance IPC control used in ESC is not fully transparent in low frequency. The control always exerts a sensation of inertia on the user action, however, it is possible to reduce this inertia by selecting a lower damping factor, but with the cost of stability. The experimental results prove that the architecture can perform position and force tracking, and the in Cartesian space verification shows that the system can overcome the limitation of using kinematically identical haptic devices.

The original ESC architecture utilizes a dominance factor α to determine the dominance of the slave according to the trainer and the trainee. This dominance allows the three possible scenarios of training: Demonstration, Guidance, and Evaluation. This dominance factor can also be used with an Adaptive Authority Adjusting method that provides a change of dominance in real-time according to the trainee's performance. Although ESC provides an AAA function, it uses four different parameters, which complicates the tuning process for the trainer. To solve this, we proposed a new method based on four profiles. The proposed profiles (Linear, Concave, Convex, and Bézier) allow a smooth change of the dominance according to the performance of the trainee. The profiles and its parameter e_{max} can be easily parameterized in real-time by the trainer. Unfortunately, our user feedback tests do not offer conclusive results to consider that the eAAA improves trainee learning. However, our test bench can be implemented for future work with long-term test subjects to validate this hypothesis.

The eESC architecture was expanded to work with multiple users at the same time. This allows m trainees to interact directly with one trainer. The dominance was upgraded by introducing a selection trainee (Δ) variable. The connection nodes of the architecture have been reworked to allow easy expansion to n master devices. Our test with two real trainees, one virtual trainee, and one trainer show that the system can perform position tracking and force tracking in all the devices. This new architecture opens the way to implementing parallel teaching, where a single trainer instructs m trainees in real-time.

The expanded ESC architecture can also be used for other training areas as control of manipulators used in dangerous environments like explosives, nuclear or biological environments, hazardous material manipulation, manual arts, etc.

Resuming from and retaking from the objectives and use requirements we developed an architecture that can:

- be operated by m trainees at the same time (**R1** and **O2**)
- provide several training modes which vary the dominance between trainer and trainees (**R2**, **R3** and **O3**). And also allows the trainer to select a trainee leader (**R4**)
- be used with different haptic devices, regardless of its kinematic configuration in n -DoF (**R5**, **R7**, **O1**, and **O5**)
- offer position and force tracking while ensuring the systems' stability (**R5**, **R6**, **R8**, **O4**, and **O6**)

For future work, we propose the following objectives:

- To enable teleoperation with the eESC architecture. By allowing the architecture to work in a remote training approach, the trainee can practice in any place regardless of the localization of the trainer. This also gives the possibility of training with costly medical equipment, like assisted-surgery robots, that can not be displaced from its location. With a remote training approach, students can use this equipment from their university. To achieve this several Time-Varying delay techniques like the Smith predictor (Uddin and Ryu [2016]), Adaptive-control under delay (Ching and Book [2006]) or scattering variables (Stramigioli et al. [2002]). A study comparative between these methods must be made to determine which one has better performances in training sessions.

- To validate the didactic interest of the dual user training system. For this, it is needed a more detailed study with long-term training sessions and more significant and real tasks, whose difficulty can be changed for evaluated trainee improvement. Also, this study can provide information about how immersive the simulator is, and how well the acquired skills are extrapolated into a real event. This also requires a change of the haptic devices, because of the limitations of the Geomagic Touch 3D, tasks that require the use of mechanical wrist can not be implemented. As it is necessary to prove that the architecture improves the trainee skills in complex tasks, the study must be a long-term one and continuously evaluate the students progress in comparison with students using classical approaches and other simulators.
- To reduce the felt inertia to the user. Based on the transparent study, to reduce the felt inertia added by the damping factor, it is required to propose a modification to the Impedance-Admittance IPC without compromising the passivity of the system. This can be done by recalculating eESC architecture and by finding a mathematical approach that allows to reduce the damping effect without compromising the passivity, or by adding in parallel a component that mitigates the inertia during low-frequency.

Part IV

Appendix

Appendix A

Mathematical models

A.1 Geomagic Touch 3D

The following mathematical models for the Geomagic Touch 3D (Figure A-1) are based on the ones published by Tavakoli et al. [2008].

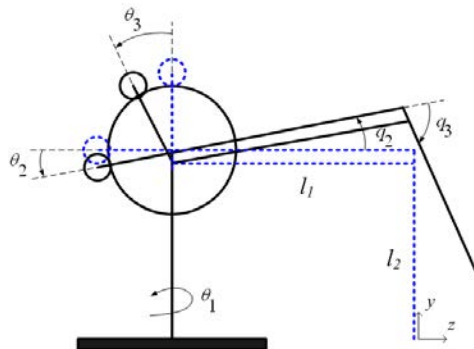


Figure A-1 – Geomagic Touch 3D kinematic analysis

A.1.1 Forward kinematics

The forward kinematic model of a manipulator is the relation between the end effector Cartesian position and the joint angles of the links. The Forward direct kinematics of the Geomagic 3D Touch is identical to the Phantom Omni (which is a previous version) presented by Jarillo-Silva et al. [2009]

and are set as:

$$\begin{aligned}x &= -\sin(\theta_1)(l_1 \cos(\theta_2) + l_2 \sin(\theta_3)) \\y &= l_2 - l_2 \cos(\theta_3) + l_1 \sin(\theta_2) \\z &= -l_1 + \cos(\theta_1)(l_1 \cos(\theta_2) + l_2 \sin(\theta_3))\end{aligned}\tag{A.1}$$

A.1.2 Inverse kinematics

The inverse kinematics is used to obtain the angular configuration of a serial-link robot ($\theta_1 \dots \theta_n$) in function of the cartesian variables (x, y and z). For the Geomagic Touch the inverse kinematics is set by equations A.2, A.3 and A.4

$$\theta_1 = \text{atan2}(x, z + l_1)\tag{A.2}$$

$$\theta_2 = \text{atan2}(y - l_2 a_1) + \cos^{-1}\left(\frac{l_1^2 - l_2^2 + a_2^2}{2l_1 a_2}\right)\tag{A.3}$$

$$\theta_3 = \theta_2 + \cos^{-1}\left(\frac{l_1^2 + l_2^2 - a_2^2}{2l_1 l_2}\right) - \frac{\pi}{2}\tag{A.4}$$

Where

$$a_1 = \sqrt{x^2 + (z + l_1)^2}\tag{A.5}$$

$$a_2 = \sqrt{x + (y - l_2)^2 + (z + l_1)^2}\tag{A.6}$$

A.1.3 Jacobian matrix

The coordinates Jacobian matrix of a manipulator robot is the function that helps relate the velocity the cartesian with the joint velocities (Spong et al. [2004]). It is used to commute the Cartesian velocity of the end-effector, to plan the trajectories and to transform the actuator torques into the force exerted by the end effector in Cartesian coordinates.

The Jacobian is as a $6 \times n$ matrix where n is the number of Dof that the manipulator has it is set as:

$$J = \begin{bmatrix} J_v \\ J_w \end{bmatrix} \quad (\text{A.7})$$

Where J_v and J_w are $3 \times n$ matrix, denoting the linear and angular velocity of the end effector respectively.

The Jacobian matrix for a Geomagic 3D touch is determined as:

$$J(\theta) = \begin{bmatrix} l_1 & -l_1 \sin(\theta_1) \sin(\theta_2) & \sin(\theta_2) \sin(l_2 + l_1 \sin(\theta_2)) \\ 0 & l_1 \cos(\theta_2) & l_1 (\cos(\theta_1) - \cos(\theta_2)) \\ 0 & -l_1 \cos(\theta_1) \sin(\theta_2) & \cos(\theta_1) (l_2 + l_1 \sin(\theta_2)) \\ 0 & 0 & -\cos(\theta_2) \\ 1 & 0 & 0 \\ 0 & 0 & \sin(\theta_1) \end{bmatrix} \quad (\text{A.8})$$

And the linear velocity of the system is given by:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = [J_v] \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad (\text{A.9})$$

To determine the desired torque applied to the actuators given desired force in the end-effector, we use:

$$\tau = J_v^T \times F \quad (\text{A.10})$$

where τ is the 3×1 vector torque applied to the manipulator actuators and F is the 3×1 vector force. By isolating F we could also get the force that is applied by the end effector :

$$F = (J_v^{-1})^T \times \tau \quad (\text{A.11})$$

This last expression determines the forces applied by the Geomagic Touch without the need of a torque sensor. However the use of this expression is limited by the capacity of the actuators. If the haptic device gets saturated, this force signal will further to increase, even if in reality such force is not applied it.

A.1.4 Lagrangian formulation

The Lagrangian formulation can be used to get the dynamic model of the Geomagic Touch 3d. The Lagrangian formulation is:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i \quad i = 1, 2, 3 \quad (\text{A.12})$$

So, using the Lagrangian formulation, the dynamic model of the system in matrix form are:

$$\begin{pmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & M_{23} \\ 0 & M_{32} & M_{33} \end{pmatrix} \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{pmatrix} + \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & 0 & C_{23} \\ C_{31} & C_{32} & 0 \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{pmatrix} + \begin{pmatrix} 0 \\ G_1 \\ G_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} \quad (\text{A.13})$$

where:

$$\begin{aligned} M_{11} &= \frac{1}{8}(4I_{ayy} + 4I_{azz} + 8I_{baseyy} + 4I_{beyy} + 4I_{bezz} + 4I_{cyy} + 4I_{czz} + 4I_{dfyy} + 4I_{dfzz} \\ &+ 4l_1^2 m_a + l_2^2 m_c + 4l_3^2 m_c) + \frac{1}{8}(4I_{beyy} - 4I_{bezz} + 4I_{cyy} - 4I_{czz} + l_1^2(4m_a + m_c)) \cos(2\theta_3) \\ &+ \frac{1}{8}(4I_{ayy} - 4I_{azz} + 4I_{dfyy} - 4I_{dfzz} - l_2^2 m_a - 4l_3^2 m_c) \cos(2\theta_3) + l_1(l_2 m_a \\ &+ l_3 m_c) \cos(\theta_2) \sin(\theta_3) \\ M_{22} &= \frac{1}{4}(4(I_{bexx} + I_{cxx} + l_1^2 m_a) + l_1^2 m_c) \\ M_{23} &= -\frac{1}{2} l_1 (l_2 m_a + l_3 m_c) \sin(\theta_2 - \theta_3) \\ M_{32} &= -\frac{1}{2} l_1 (l_2 m_a + l_3 m_c) \sin(\theta_2 - \theta_3) \\ M_{33} &= \frac{1}{4}(4I_{axx} + 4I_{dfxx} + l_2^2 m_a + 4l_3^2 m_c) \end{aligned} \quad (\text{A.14})$$

$$\begin{aligned}
C_{11} &= \frac{1}{8}(-2 \sin(\theta_2)((4I_{beyy} - 4I_{bezz} + 4I_{cyy} + 4l_1^2 m_a + l_1^2 m_c) \cos(\theta_2) \\
&\quad + 2l_1(l_2 m_a + l_3 m_c) \sin(\theta_3)) \dot{\theta}_2 + 2 \cos(\theta_3)(2l_1(l_2 m_a + l_3 m_c) \cos(\theta_2) + (-4I_{ayy} + 4I_{azz} \\
&\quad - 4I_{dfzz} + l_2^2 m_a + 4l_3^2 m_c) \sin(\theta_3)) \dot{\theta}_3) \\
C_{12} &= -\frac{1}{8}(-4l_1(l_2 m_a + l_3 m_c) \cos(\theta_2) \cos(\theta_3) - (-4I_{ayy} + 4I_{azz} - 4I_{dfyy} + 4I_{dfzz} \\
&\quad + l_2^2 m_a + 4l_3^2 m_c) \sin(2\theta_3)) \dot{\theta}_1 \\
C_{13} &= -\frac{1}{8}(-4l_1(l_2 m_a + l_3 m_c) \cos(\theta_2) \cos(\theta_3) - (-4I_{ayy} + 4I_{azz} - 4I_{dfyy} + 4I_{dfzz} + l_2^2 m_a \\
&\quad + 4l_3^2 m_c) \sin(2\theta_3)) \dot{\theta}_3 \\
C_{21} &= -C_{12} \\
C_{23} &= \frac{1}{2}l_1(l_2 m_a + l_3 m_c) \cos(\theta_2 - \theta_3) \dot{\theta}_3 \\
C_{31} &= -C_{13} \\
C_{32} &= \frac{1}{2}l_1(l_2 m_a + l_3 m_c) \cos(\theta_2 - \theta_3) \dot{\theta}_2 \\
G_1 &= \frac{1}{2}g(2l_1 m_a + 2l_5 m_{be} + l_1 m_c) \cos(\theta_2) \\
G_2 &= \frac{1}{2}g(l_2 m_a + 2l_3 m_c - 2l_6 m_{df}) \sin(\theta_3)
\end{aligned} \tag{A.15}$$

The numerical values of equation A.14 and A.15 are:

$$\begin{aligned}
l_1 &= 0.215; l_2 = 0.170; l_3 = 0.0325; l_5 = 0.0368; l_6 = 0.0527; \\
m_a &= 0.0202; m_c = 0.0249; m_{be} = 0.2359; m_{df} = 0.1906; \\
I_{axx} &= 0.4864 \cdot 10^{-4}; I_{ayy} = 0.0018 \cdot 10^{-4}; I_{azz} = 0.4864 \cdot 10^{-4}; \\
I_{cxx} &= 0.959 \cdot 10^{-4}; I_{cyy} = 0.959 \cdot 10^{-4}; I_{czz} = 0.0051 \cdot 10^{-4}; \\
I_{bexx} &= 11.09 \cdot 10^{-4}; I_{beyy} = 10.06 \cdot 10^{-4}; I_{bezz} = 0.591 \cdot 10^{-4}; \\
I_{dfxx} &= 7.11 \cdot 10^{-4}; I_{dfyy} = 0.629 \cdot 10^{-4}; I_{dfzz} = 6.246 \cdot 10^{-4}; \\
I_{baseyy} &= 11.87 \cdot 10^{-4}; g = 9.81;
\end{aligned}$$

A.2 Root Mean Square

As, in our experiments, the motion of the haptic devices is in 3-Dof it evaluating the performance requires objective metrics. By using the Root Mean Square function (RMS), it is possible to quantify

it. The RMS function is used to represent an enhanced the average of the error between two devices (which can be Master 1, Master 2, or the slave) for all the performed trajectories. This value helps determinate how close are both devices during a task.

To compute the RMS, first we need to get the difference of position of each sample during a test. The difference of position is given by:

$$\Delta P[k]^2 = (x_1[k] - x_2[k])^2 + (y_1[k] - y_2[k])^2 + (z_1[k] - z_2[k])^2 \quad (\text{A.16})$$

Where ΔP is the position error between the first device and the second device. $x_{1,2}, y_{1,2}, z_{1,2}$ are the position components for each device. These components are expressed in the Cartesian space but can be also expressed in the angular space $(\theta_1, \theta_2, \theta_3)$.

Once the position error is determined, we evaluate all the position samples for all the trajectory. By using eq. A.17 with the values of each iteration, we get the value of RMS.

$$\varepsilon = \sqrt{\frac{1}{k_t} \sum_{k=1}^k (\Delta P[k])^2} \quad (\text{A.17})$$

Where ε is the RMS error and k_t is the total number of samples of the trajectory. As ε represents a difference of position between both devices, it is always desirable that $\varepsilon \rightarrow 0$, as this means both systems are as close as possible, following the same trajectory.

Appendix B

Hybrid matrix of Impedance-Admittance IPC

Expanding the hybrid matrix of eq.(3.26) in section 3.2.2, we get

$$h_{11} = \frac{h_{11num}}{h_{11den}} \quad (\text{B.1})$$

Where

$$\begin{aligned} h_{11num} = & M_1 M_2 M_{mm} M_{ms} s^7 + (M_1 M_{mn} M_{ms} B_{mi2} + M_2 M_{mn} M_{ms} B_{mi1}) s^6 \\ & + (M_1 M_2 M_{mm} k_{mi2} + M_1 M_2 M_{ms} k_{mc} + M_1 M_{mn} M_{ms} k_{mi1} + M_2 M_{mn} M_{ms} k_{mc} \\ & + M_1 M_{mn} M_{ms} k_{mi2} + M_2 M_{mn} M_{ms} k_{mi1} + M_{mn} M_{ms} B_{mi1} B_{mi2}) s^5 + (M_1 M_{mn} B_{mi2} k_{mi2} \\ & + M_2 M_{mn} B_{mi1} k_{mi2} + M_1 M_{ms} B_{mi2} k_{mc} + M_2 M_{ms} B_{mi1} k_{mc} + M_{mn} M_{ms} B_{mi1} k_{mi1} + M_{mn} M_{ms} B_{mi2} k_{mc} \\ & + M_{mn} M_{ms} B_{mi1} k_{mi2} + M_{mn} M_{ms} B_{mi2} k_{mi1}) s^4 + (M_1 M_2 k_{mc} k_{mi2} + M_1 M_{mn} k_{mi1} k_{mi2} \\ & + M_2 M_{mn} k_{mc} k_{mi2} + M_2 M_{mn} k_{mi1} k_{mi2} + M_1 M_{ms} k_{mc} k_{mi1} + M_1 M_{ms} k_{mc} k_{mi2} + M_2 M_{ms} k_{mc} k_{mi1} \\ & + M_{mn} M_{ms} k_{mc} k_{mi1} + M_{mn} M_{ms} k_{mc} k_{mi2} + M_{mn} M_{ms} k_{mi1} k_{mi2} + M_{mn} B_{mi1} B_{mi2} k_{mi2} \\ & + M_{ms} B_{mi1} B_{mi2} k_{mc}) s^3 + (M_1 B_{mi2} k_{mc} k_{mi2} + M_2 B_{mi1} k_{mc} k_{mi2} + M_{mn} B_{mi1} k_{mi1} k_{mi2} + M_{mn} B_{mi2} k_{mc} k_{mi2} \\ & + M_{mn} B_{mi2} k_{mi1} k_{mi2} + M_{ms} B_{mi1} k_{mc} k_{mi1} + M_{ms} B_{mi1} k_{mc} k_{mi2} + M_{ms} B_{mi2} k_{mc} k_{mi1}) s^2 + (M_1 k_{mc} k_{mi1} k_{mi2} \\ & + M_2 k_{mc} k_{mi1} k_{mi2} + M_{mn} k_{mc} k_{mi1} k_{mi2} + M_{ms} k_{mc} k_{mi1} k_{mi2} + B_{mi1} B_{mi2} k_{mc} k_{mi2}) s \\ & + B_{mi1} k_{mc} k_{mi1} k_{mi2} + B_{mi2} k_{mc} k_{mi1} k_{mi2} \end{aligned}$$

$$\begin{aligned}
h_{11_{den}} = & M_1 M_2 M_{ms} s^6 + (M_1 M_{ms} B_{mi2} + M_2 M_{ms} B_{mi1}) s^5 + (M_1 M_2 k_{mi2} \\
& + M_1 M_{ms} k_{mi1} + M_2 M_{ms} k_{mc} + M_1 M_{ms} k_{mi2} + M_2 M_{ms} k_{mi1} \\
& + M_{ms} B_{mi1} B_{mi2}) s^4 + (M_1 B_{mi2} k_{mi2} + M_2 B_{mi1} k_{mi2} + M_{ms} B_{mi1} k_{mi1} \\
& + M_{ms} B_{mi2} k_{mc} + M_{ms} B_{mi1} k_{mi2} + M_{ms} B_{mi2} k_{mi1}) s^3 \\
& + (M_1 k_{mi1} k_{mi2} + M_2 k_{mc} k_{mi2} + M_2 k_{mi1} k_{mi2} + M_{ms} k_{mc} k_{mi1} \\
& + M_{ms} k_{mc} k_{mi2} + M_{ms} k_{mi1} k_{mi2} + B_{mi1} B_{mi2} k_{mi2}) s^2 \\
& + (B_{mi1} k_{mi1} k_{mi2} + B_{mi2} k_{mc} k_{mi2} + B_{mi2} k_{mi1} k_{mi2}) s \\
& + k_{mc} k_{mi1} k_{mi2}
\end{aligned}$$

$$h_{12} = \frac{k_{mc} k_{mi1} k_{mi2}}{h_{12_{den}}} \quad (\text{B.2})$$

Where

$$\begin{aligned}
h_{12_{den}} = & M_1 M_2 M_{ms} s^6 + (M_1 M_{ms} B_{mi2} + M_2 M_{ms} B_{mi1}) s^5 + (M_1 M_2 k_{mi2} + M_1 M_{ms} k_{mi1} \\
& + M_2 M_{ms} k_{mc} + M_1 M_{ms} k_{mi2} + M_2 M_{ms} k_{mi1} + M_{ms} B_{mi1} B_{mi2}) s^4 + (M_1 B_{mi2} k_{mi2} + M_2 B_{mi1} k_{mi2} \\
& + M_{ms} B_{mi1} k_{mi1} + M_{ms} B_{mi2} k_{mc} + M_{ms} B_{mi1} k_{mi2} + M_{ms} B_{mi2} k_{mi1}) s^3 + (M_1 k_{mi1} k_{mi2} + M_2 k_{mc} k_{mi2} \\
& + M_2 k_{mi1} k_{mi2} + M_{ms} k_{mc} k_{mi1} + M_{ms} k_{mc} k_{mi2} + M_{ms} k_{mi1} k_{mi2} + B_{mi1} B_{mi2} k_{mi2}) s^2 + (B_{mi1} k_{mi1} k_{mi2} \\
& + B_{mi2} k_{mc} k_{mi2} + B_{mi2} k_{mi1} k_{mi2}) s + k_{mc} k_{mi1} k_{mi2}
\end{aligned}$$

$$h_{21} = -h_{12} \quad (\text{B.3})$$

$$h_{22} = \frac{h_{22_{num}}}{h_{22_{den}}} \quad (\text{B.4})$$

Where

$$\begin{aligned}
h_{22_{num}} = & -M_1 M_2 s^5 + (-M_1 B_{mi2} - M_2 B_{mi1}) s^4 + (-M_1 k_{mi1} - M_2 k_{mc} - M_1 k_{mi2} - M_2 k_{mi1} - B_{mi1} B_{mi2}) s^3 \\
& + (-B_{mi1} k_{mi1} - B_{mi2} k_{mc} - B_{mi1} k_{mi2} - B_{mi2} k_{mi1}) s^2 + (-k_{mc} k_{mi1} - k_{mc} k_{mi2} - k_{mi1} k_{mi2}) s
\end{aligned}$$

$$\begin{aligned}
h_{22_{den}} = & M_1 M_2 M_{ms} s^6 + (M_1 M_{ms} B_{mi2} + M_2 M_{ms} B_{mi1}) s^5 + (M_1 M_2 k_{mi2} \\
& + M_1 M_{ms} k_{mi1} + M_2 M_{ms} k_{mc} + M_1 M_{ms} k_{mi2} + M_2 M_{ms} k_{mi1} \\
& + M_{ms} B_{mi1} B_{mi2}) s^4 + (M_1 B_{mi2} k_{mi2} + M_2 B_{mi1} k_{mi2} + M_{ms} B_{mi1} k_{mi1} \\
& + M_{ms} B_{mi2} k_{mc} + M_{ms} B_{mi1} k_{mi2} + M_{ms} B_{mi2} k_{mi1}) s^3 + (M_1 k_{mi1} k_{mi2} \\
& + M_2 k_{mc} k_{mi2} + M_2 k_{mi1} k_{mi2} + M_{ms} k_{mc} k_{mi1} + M_{ms} k_{mc} k_{mi2} \\
& + M_{ms} k_{mi1} k_{mi2} + B_{mi1} B_{mi2} k_{mi2}) s^2 + (B_{mi1} k_{mi1} k_{mi2} \\
& + B_{mi2} k_{mc} k_{mi2} + B_{mi2} k_{mi1} k_{mi2}) s + k_{mc} k_{mi1} k_{mi2}
\end{aligned}$$

Appendix C

Time Domain Passivity Control (TDPC)

A standard method used to offer stability in a bilateral teleoperation system is the use of a Time Domain Passivity Control (TDPC). These methods are based on the passivity approach, which offers stability to the system as long as the conditions are met.

C.1 Passivity

The passivity approach focuses on the quantity of energy exchanged between a system and its environment. The advantages of the passivity, listed by Hannaford and Ryu [2002], are:

- The system is passive if the energy that flows in the inputs always exceeds the one flowing in the outputs . That means a system must dissipate more energy than it receives.
- It allows subdividing the system into individual system blocks, or subsystems. If all the blocks are passive, and their interconnections are passive, the global system is passive as well.
- It is valid for linear and non-linear systems
- There is evidences in the bibliography that prove the human intervention is passive as well.

To define passivity, it is necessary to consider a system H , as shown in Figure C-1 with input $u(t) \in \mathbb{R}^P$, output $y(t) \in \mathbb{R}^P$ and $t \in \mathbb{R}_+$.

Then the passivity is defined by Van Der Schaft [2000] as:

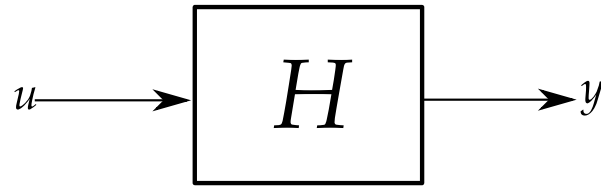


Figure C-1 – Input and Output from a H system

Definition 1: Global Passivity of a system

The system H with states x is passive from input $u \in U$ and output $y \in Y$ if there exists a semidefinite function $S : \mathbb{R}^p \rightarrow \mathbb{R}_+$ called storage function such that

$$S(x(\tau)) - S(x_0) \leq \int_0^\tau y^T(t)u(t)dt$$

For all input signals $u : [0, \tau] \rightarrow \mathbb{R}^p$ with initial states $x_0 \in \mathbb{R}^p$ and $\tau \in \mathbb{R}_+$. It is said also that:

The input is strictly passive if there exists $\delta_u > 0$ such that

$$S(x(\tau)) - S(x_0) \leq \int_0^\tau (y^T(t)u(t) - \delta_u \|u(t)\|^2)dt$$

Moreover, the output is also strictly passive if there exists a scalar $\delta_y > 0$ such that

$$S(x(\tau)) - S(x_0) \leq \int_0^\tau (y^T(t)u(t) - \delta_y \|y(t)\|^2)dt$$

For all the input signals $u : [0, \tau] \rightarrow \mathbb{R}^p$

Where S is the total energy of the system, the term $y^T(t)u(t)$ is the power supplied to the system at time t . The integral term $\int_0^\tau y^T(t)u(t)dt$ is the energy supplied to the system during the time interval $[0, \tau]$.

C.1.1 Passivity for an M-port network

To prove passivity in an M-port network (a system with multiple inputs and outputs) first, we establish the system, as shown in Figure C-2, with the convention of forces and velocities. We declare storage energy as E and initially is equal to 0 at $t = 0$.

The definition of passivity for the M-port network given is:

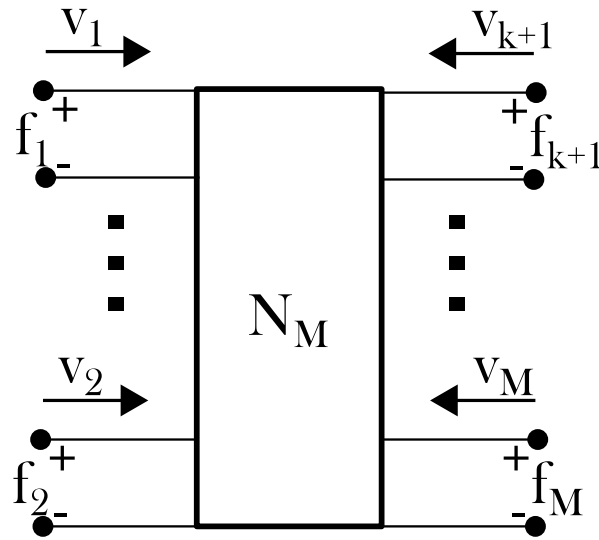


Figure C-2 – M-port network

Definition 2: Passivity for a M-port network

The M-port network N_M with initial storage energy $E(0)$ is passive if and only if

$$\int_0^{\tau} (f_1(\tau)v_1(\tau) + \dots + f_M(\tau)v_M(\tau))d\tau + E(0), \forall t \geq 0$$

for all the admissible forces (f_1, \dots, f_M) and velocities (v_1, \dots, v_M)

C.1.2 Passivity observer

A passivity observer is a function which helps determine if a system remains passive or not during its operation. The passivity observer can be instrumented by reading the forces and velocities of the system, which can be approximated or sensed. Equation C.1 defines the passivity observer.

$$E_{obsv}(n) = \sum_{k=1}^n [f_1(k)v_1(k) + \dots + f_M(k)v_M(k)] \quad (C.1)$$

So, in the passivity observer if $E_{obsv} \geq 0$ for every time n , the system is dissipating energy, thus the system is passive. On the other hand, if $E_{obsv} < 0$ this means that the system is active, and therefore generating energy.

Appendix D

Survey AAA validation

The following questions were asked to the participants of section 4.4 to evaluate their qualitative response to the haptic system.

1. What group are you in?

- Group 1
- Group 2

2. Can your data and information be used for evaluation and research purposes?

- Yes
- No

3. What is your current employment status?

- Teacher
- Student
- Other

4. Have you ever used a force feedback joystick or haptic interface like the one used here?

- Yes
- No

5. How did the guidance feel?

- Natural
 - Heavy
 - Twitchy
 - Soft
 - Abrupt
 - Other
6. How much did the trainer (in guidance mode) help you in your manipulation from 1 (a few) to 5 (a lot)?
- 1
 - 2
 - 3
 - 4
 - 5
 - No answer
7. How much did the video stream help you in your manipulations from 1 (a few) to 5 (a lot)?
- 1
 - 2
 - 3
 - 4
 - 5
 - No answer
8. Did you succeed?
- Yes
 - No
9. Final comments

In the case of question 4 to 7, the Group 2 participants were instructed to no answered it as this mode was not implicated in their test.

Table D.1 – Participants' answers to the survey

Participant	Language	Group	Q1	Q2	Q3	Q4	Q5	Q6	Q7
1	French	1	Yes	Student	No	1	5	5	Yes
2	French	2	Yes	Student	No				
3	English	1	Yes	Student	No	4	3	5	Yes
4	English	2	Yes	Student	No				
5	English	1	Yes	Student	No	2	4	5	Yes
6	English	2	Yes	Student	No				
7	English	1	Yes	Student	No	1	3	4	Yes
8	French	2	Yes	Student	No				
9	English	1	Yes	Student	No	1	4	5	Yes
10	French	2	Yes	Student	No				
11	English	1	Yes	Student	No	3	3	4	Yes
12	French	2	Yes	Student	No				
13	English	1	Yes	Student	No	3	4	4	Yes
14	French	2	Yes	Student	Yes				
15	English	1	Yes	Student	No	4	2	3	Yes
16	French	2	Yes	Student	No				
17	English	1	Yes	Student	No	1	4	5	Yes
18	French	2	Yes	Student	No				

Bibliography

- 3DSYSTEMS (2019). <https://www.3dsystems.com/haptics-devices/touch/specifications>. Accessed:2019-10-15.
- Alamilla Daniel, M. d. I. a., Moreau, R., and Redarce, T. (2018). A new method to render virtual walls for haptic systems: “Tracking wall”. Application to needle insertion simulation. In *ICMCE*, page 6 pages, Amsterdam, Netherlands.
- Aldana, C. I., Nuño, E., Basañez, L., and Romero, E. (2014). Operational space consensus of multiple heterogeneous robots without velocity measurements. *Journal of the Franklin Institute*, 351(3):1517–1539.
- Aliaga, I., Rubio, Á., and Sánchez, E. (2004). Experimental Quantitative Comparison of Different Control Architectures for Master-Slave Teleoperation. *IEEE Transactions on Control Systems Technology*, 12(1):2–11.
- Arcara, P. and Melchiorri, C. (2002). Control schemes for teleoperation with time delay: A comparative study. *Robotics and Autonomous Systems*, 38(1):49–64.
- Arcara, P., Melchiorri, C., and Stramigioli, S. (2001). Intrinsically passive control in bilateral teleoperation mimo systems. In *2001 European Control Conference (ECC)*, pages 1180–1185, Porto, Portugal.
- Beal, M. D., Kinnear, J., Anderson, C. R., Martin, T. D., Wamboldt, R., and Hooper, L. (2017). The effectiveness of medical simulation in teaching medical students critical care medicine. *Simulation in Healthcare*, 12(2):104–116.
- Beltran-Carbajal, F. (2012). Advances in Vibration Engineering and Structural Dynamics. *InTech*, page 378.
- Berger, J. T., Rosner, F., and Cassell, E. J. (2002). Ethics of practicing medical procedures on newly dead and nearly dead patients. *Journal of General Internal Medicine*, 17(10):774–778.
- Boost (2019-09-02). Boost C++ Libraries. <http://www.boost.org/>.
- Ching, H. and Book, W. J. (2006). Internet-Based Bilateral Teleoperation Based on Wave Variable With Adaptive Predictor and Direct Drift Control. *Journal of Dynamic Systems, Measurement, and Control*, 128(1):86.
- Conti, F., Barbagli, F., Balaniuk, R., Halg, M., Lu, C., Morris, D., Sentis, L., Warren, J., Khatib, O., and Salisbury, K. (2003). The CHAI libraries. In *Proceedings of Eurohaptics 2003*, pages 496–500, Dublin, Ireland.

- Cooper, J. B. and Taqueti, V. R. (2008). A brief history of the development of mannequin simulators for clinical education and training (Reprinted from *Quality & Safety in Health Care*, vol 13, pg 111-118, 2004). *Postgraduate Medical Journal*, 84(997):563–570.
- Cristian, L., Banerjee, P., Florea, L., and Dawe, G. (2005). Design of the ImmersiveTouch™: a High-Performance Haptic Augmented Virtual Reality System. In *11th international conference on human-computer interaction*, Las Vegas, NV.
- Cui, J., Tosunoglu, S., Roberts, R., Moore, C., and Repperger, D. W. (2003). A review of teleoperation system control. In *Proceedings of the Florida Conference on Recent Advances in Robotics (FCRAR)*, pages 1–12, Florida, USA.
- Culbertson, H., Schorr, S. B., and Okamura, A. M. (2018). Haptics: The Present and Future of Artificial Touch Sensation. *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1):385–409.
- Delorme, S., Laroche, D., Diraddo, R., and F. Del Maestro, R. (2012). NeuroTouch: A physics-based virtual simulator for cranial microneurosurgery training. *Neurosurgery*, 71(SUPPL.1):32–42.
- Deutsch, E. S. (2008). High-fidelity patient simulation mannequins to facilitate aerodigestive endoscopy training. *Archives of Otolaryngology - Head and Neck Surgery*, 134(6):625–629.
- Escobar-Castillejos, D., Noguez, J., Neri, L., Magana, A., and Benes, B. (2016). A Review of Simulators with Haptic Devices for Medical Training. *Journal of Medical Systems*, 40(4):1–22.
- Feygin, D., Keehner, M., and Tendick, R. (March 2002). Haptic Guidance: Experimental Evaluation of a Haptic Training Method for a Perceptual Motor Skill. *10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. (HAPTICS 2002)*, pages 40–47.
- Franken, M., Stramigioli, S., Misra, S., Secchi, C., and Macchelli, A. (2011). Bilateral Telemanipulation With Time Delays: A Two-Layer Approach Combining Passivity and Transparency. *Robotics, IEEE Transactions on*, 27(4):741–756.
- Fritz, P. Z., Gray, T., and Flanagan, B. (2008). Review of mannequin-based high-fidelity simulation in emergency medicine. *EMA - Emergency Medicine Australasia*, 20(1):1–9.
- Galvao, A. and Kemp, A. (1999). Kinaesthesia and Instrumental Music Instruction: Some Implications. *Psychology of Music*, 27(2):129–137.
- Hannaford, B. (1989). A Design Framework for Teleoperators with Kinesthetic Feedback. *IEEE Transactions on Robotics and Automation*, 5(4):426–434.
- Hannaford, B. and Ryu, J. H. (2002). Time-domain passivity control of haptic interfaces. *IEEE Transactions on Robotics and Automation*, 18(1):1–10.
- Hashtrudi-Zaad, K. and Salcudean, S. E. (2001). Analysis of Control Architectures for Teleoperation Systems with Impedance/Admittance Master and Slave Manipulators. *The International Journal of Robotics Research*, 20(6):419–445.
- Huang, P. and Lu, Z. (2015). Auxiliary asymmetric dual-user shared control method for teleoperation. In *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence, (URAI 2015)*, pages 267–272.

- Jarillo-Silva, A., Domínguez-Ramírez, O. A., Parra-Vega, V., and Ordaz-Oliver, J. P. (2009). PHAN-ToM OMNI haptic device: Kinematic and manipulability. In *Electronics Robotics and Automotive Mechanics Conference (CERMA 2009)*, pages 193–198.
- Jones, F., Passos-Neto, C. E., and Braghiroli, O. F. M. (2015). Telemedicine Systems. *Principles and Practice of Clinical Research*, 2(1):56–63.
- Katsura, S., Suzuyama, T., and Ohishi, K. (2007). A realization of multilateral force feedback control for cooperative motion. *IEEE Transactions on Industrial Electronics*, 54(6):3298–3306.
- Khademian, B. and Hashtrudi-Zaad, K. (2007). A four-channel multilateral shared control architecture for dual-user teleoperation systems. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2660–2666.
- Khademian, B. and Hashtrudi-Zaad, K. (2013). A Framework for Unconditional Stability Analysis of Multimaster/Multislave Teleoperation Systems. *Robotics, IEEE Transactions on*, 29(3):684–694.
- Kunkler, K. (2008). The role of medical simulation: an overview. In *The International Journal Of Medical Robotics And Computer Assisted Surgery*, number September 2006, pages 203–210.
- Li, Z., Ding, L., Gao, H., Duan, G., and Su, C.-y. (2013). Trilateral Teleoperation of Adaptive Fuzzy Force / Motion Control for Nonlinear Teleoperators. *IEEE Transactions on Fuzzy Systems*, 21(4):610–624.
- Licona Rodriguez, A. R., Lelevé, A., Pham, M. T., and Eberard, D. (2019). A multi-trainee architecture for haptic hands-on training. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Macau, China.
- Liu, F. (2016). *Dual-user Haptic Training System*. PhD thesis, Institut National des Sciences Appliquées de Lyon (INSA Lyon).
- Liu, F., Lelevé, A., Eberard, D., and Redarce, T. (2015). A dual-user teleoperation system with online authority adjustment for haptic training. In *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1168–1171.
- Liu, F., Lelevé, A., Eberard, D., and Redarce, T. (2015). A Dual-user Teleoperation System with Adaptive Authority Adjustment for Haptic Training. In *MESROB 2015*, Nantes, France.
- Liu, F., Licona Rodriguez, A. R., Lelevé, A., Eberard, D., Pham, M. T., and Redarce, T. (2019). An Energy-Based Approach for n-dof Passive Dual-user Haptic Training Systems. In *Robotica*. Cambridge University Press.
- Moghimi, S., Sirouspour, S., and Malysz, P. (2008). Haptic-enabled Collaborative Training with Generalized Force and Position Mappings. In *Haptic interfaces for virtual environment and teleoperator systems*, pages 287–294.
- Neumann, P. F., Sadler, L. L., and Gieser, J. (1998). Virtual reality vitrectomy simulator. In *Wells W.M., Colchester A., Delp S. (eds) Medical Image Computing and Computer-Assisted Intervention — MICCAI'98. MICCAI 1998. Lecture Notes in Computer Science*, volume 1496, pages 910–917, Springer, Berlin.

- Niemeyer, G. and Slotine, J. J. E. (1991). Stable Adaptive Teleoperation. *IEEE Journal of Oceanic Engineering*, 16(1):152–162.
- Ogata, K. (2003). *System Dynamics*. Pearson Prentice Hall.
- Owen, H. (2012). Early use of simulation in medical education. *Simulation in Healthcare*, 7(2):102–116.
- Panait, L., Akkary, E., Bell, R. L., Roberts, K. E., Dudrick, S. J., and Duffy, A. J. (2009). The Role of Haptic Feedback in Laparoscopic Simulation Training. *Journal of Surgical Research*, 156(2):312–316.
- Panzirsch, M., Balachandran, R., Artigas, J., Riecke, C., Ferre, M., and Albu-Schaeffer, A. (2017). Haptic intention augmentation for cooperative teleoperation. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5335–5341.
- Perez, A. G., Lobo, D., Chinello, F., Cirio, G., Malvezzi, M., Martín, J. S., Prattichizzo, D., and Otaduy, M. A. (2015). Soft finger tactile rendering for wearable haptics. In *2015 IEEE World Haptics Conference (WHC)*, pages 327–332.
- Raju, G. J., Verghese, G. C., and Sheridan, T. B. (1989). Design issues in 2-port network models of bilateral remote manipulation. In *Proceedings, 1989 International Conference on Robotics and Automation*, pages 1316–1321 vol.3.
- Rosker, J. and Sarabon, N. (2010). Kinaesthesia and Methods for its Assessment: Literature Review. *Sport Science Review*, 19(5-6):165–208.
- Sarakoglou, I., Brygo, A., Mazzanti, D., Hernandez, N. G., Caldwell, D. G., and Tsagarakis, N. G. (2016). Hexotrac: A highly under-actuated hand exoskeleton for finger tracking and force feedback. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1033–1040.
- Schout, B. M. A., Hendriks, A. J. M., Scheele, F., Bemelmans, B. L. H., and Scherpbier, A. J. J. A. (2010). Validation and implementation of surgical simulators: A critical review of present, past, and future. *Surgical Endoscopy and Other Interventional Techniques*, 24(3):536–546.
- Secchi, C., Fantuzzi, S., and Stramigioli Cesare (2007). *Control of Interactive Robotic Interfaces A Port-Hamiltonian Approach*. Springer.
- Shahbazi, M., Atashzar, S. F., and Patel, R. V. (2013). A dual-user teleoperated system with virtual fixtures for robotic surgical training. In *2013 IEEE International Conference on Robotics and Automation*, pages 3639–3644.
- Shahbazi, M., Atashzar, S. F., Talebi, H. A., and Patel, R. V. (2014). An expertise-oriented training framework for robotics-assisted surgery. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5902–5907.
- Shahbazi, M., Atashzar, S. F., Talebi, H. A., and Patel, R. V. (2015). Novel Cooperative Teleoperation Framework: Multi-Master/Single-Slave System. *IEEE/ASME Transactions on Mechatronics*, 20(4):1668–1679.

- Shahbazi, M., Talebi, H. A., Atashzar, S. F., Towhidkhan, F., Patel, R. V., and Shojaei, S. (2011). A novel shared structure for dual user systems with unknown time-delay utilizing adaptive impedance control. In *2011 IEEE International Conference on Robotics and Automation*, pages 2124–2129.
- Shamaei, K., Kim, L. H., and Okamura, A. M. (2015). Design and evaluation of a trilateral shared-control architecture for teleoperated training robots. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4887–4893.
- Spong, M. W., Hutchinson, S., and Vidyasagar, M. (2004). *Robot Dynamics and Control*. Wiley, second edition.
- Sreelakshmi, M. and Subash, T. (2017). Haptic Technology: A comprehensive review on its applications and future prospects. *Materials Today: Proceedings*, 4(2):4182–4187.
- Stramigioli, S. (1996). Creating Artificial Damping By Means of Damping Injection. In *Proceedings of the ASME Dynamic Systems and Control Division*, pages 601–606.
- Stramigioli, S., Van Der Schaft, A., Maschke, B., and Melchiorri, C. (2002). Geometric scattering in robotic telemanipulation. *IEEE Transactions on Robotics and Automation*, 18(4):588–596.
- Sutherland, C., Hashtrudi-Zaad, K., Sellens, R., Abolmaesumi, P., and Mousavi, P. (2013). An augmented reality haptic training simulator for spinal needle procedures. In *IEEE Transactions on Biomedical Engineering*, volume 60, pages 3009–3018.
- Tavakoli, M., Aziminejad, A., Patel, R. V., and Moallem, M. (2007). High-fidelity bilateral teleoperation systems and the effect of multimodal haptics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(6):1512–1528.
- Tavakoli, M., Patel, R. V., Moallem, M., and Aziminejad, A. (2008). *Haptics For Teleoperated Surgical Robotic Systems*. World Scientific.
- Tsai, M. D., Hsieh, M. S., and Jou, S. B. (2001). Virtual reality orthopedic surgery simulator. *Computers in Biology and Medicine*, 31(5):333–351.
- Tse, B., Harwin, W., Barrow, A., Quinn, B., San Diego, J., and Cox, M. (2010). Design and development of a haptic dental training system - HapTEL. In *Lecture Notes in Computer Science*, number 2, pages 101–108.
- Uddin, R. and Ryu, J. (2016). Annual Reviews in Control Predictive control approaches for bilateral teleoperation R. *Annual Reviews in Control*, 42:82–99.
- Ullrich, S. and Kuhlen, T. (2012). Haptic palpation for medical simulation in virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):617–625.
- Van Der Schaft, A. (2000). *L2 - Gain and Passivity Techniques in Nonlinear Control*. Springer.
- Vaughan, N., Dubey, V. N., Wainwright, T. W., and Middleton, R. G. (2016). A review of virtual reality based training simulators for orthopaedic surgery. *Medical Engineering & Physics*, 38(2):59–71.
- Woodrum, D. T., Andreatta, P. B., Yellamanchilli, R. K., Feryus, L., Gauger, P. G., and Minter, R. M. (2006). Construct validity of the LapSim laparoscopic surgical simulator. *American Journal of Surgery*, 191(1):28–32.

- Xu, X. and Ke, F. (2014). From psychomotor to motorpsycho: learning through gestures with body sensory technologies. *Educational Technology Research and Development*, 62(6):711–741.
- Zakerimanesh, A., Hashemzadeh, F., and Ghiasi, A. R. (2017). Dual-user nonlinear teleoperation subjected to varying time delay and bounded inputs. *ISA Transactions*, 68:33–47.
- Zevin, B., Aggarwal, R., and Grantcharov, T. P. (2014). Surgical simulation in 2013: Why is it still not the standard in surgical training? *Journal of the American College of Surgeons*, 218(2):294–301.



FOLIO ADMINISTRATIF

THESE DE L'UNIVERSITE DE LYON OPEREE AU SEIN DE L'INSA LYON

NOM : LICONA RODRIGUEZ

DATE de SOUTENANCE : 12/03/2020

Prénoms : Angel Ricardo

TITRE : Collaborative Hands-on Training on Haptic Simulators

NATURE : Doctorat

Numéro d'ordre : 2020LYSEI018

Ecole doctorale : Electronique, Electrotechnique, Automatique

Spécialité : Automatique

RESUME :

Depuis les débuts de la médecine, les étudiants en médecine s'entraînent sur des simulateurs (animaux, cadavres, mannequins) afin d'acquérir puis enrichir leurs compétences médicales, dont notamment celles nécessitant des gestes précis. C'est un moyen de faire des erreurs sans blesser de patient. Pourtant, ces simulateurs manquent souvent de réalisme, s'usent et finissent par coûter cher sur le long terme. Depuis plusieurs décennies, les technologies de simulation informatique ont enrichi le panel des simulateurs médicaux, et aujourd'hui les technologies haptiques offrent désormais un réalisme encore accru. Cependant, ces simulateurs haptiques n'intègrent pas le formateur dans la simulation, ce qui empêche de former les apprenant sur des gestes impliquant à la fois des trajectoires et des interactions outil/environnement où les efforts doivent être subtilement dosés et donc maîtrisés. Dans les travaux de recherche récents, des architectures dual-user (à deux utilisateurs) de simulateurs dédiés à la formation au geste ont été proposés ces dernières années, mais l'étude bibliographique proposée dans ce document démontre qu'aucun d'entre eux n'a été conçu dans ce sens : un retour haptique est proposé à l'apprenant mais le formateur n'a pas la possibilité de montrer simultanément la position des outils et les efforts à appliquer dessus.

Récemment M. Fei Liu a conçu une solution à ce besoin (l'architecture dual-user ESC - Energy Shared Control), fondée sur une approche énergétique garantissant la passivité et donc la stabilité du, utilisable avec un dispositif esclave réel ou virtuel, et intégrant un mécanisme d'autorité adaptative (Adaptive Authority Adjustment - AAA) permettant au formateur de reprendre la main instantanément à partir du moment où l'apprenant dévie de la trajectoire désirée par le formateur. Cependant, cette solution se limitait à un seul degré de liberté et ce mécanisme d'autorité adaptative nécessitait de régler laborieusement trois paramètres peu intuitifs. Les solutions proposées dans ce mémoire poursuivent donc les travaux de M. Liu en améliorant l'architecture ESC afin de l'étendre à des dispositifs à plusieurs degrés de liberté, ayant potentiellement des cinématiques différentes (surtout le dispositif esclave par rapport aux dispositifs maîtres) et à plusieurs apprenants simultanés. Une amélioration de la fonction AAA est proposée ainsi que des profils d'utilisation dépendant du niveau de l'apprenant. Une première série de retours d'usage a été réalisée afin de déterminer l'intérêt pédagogique de cette fonction. Les résultats ne permettant pas de conclure, des préconisations sont proposées pour une deuxième série.

MOTS-CLÉS : Haptique, Simulateur, Apprentissage supervisé, Dual-user, Passivité, Étude comparative

Laboratoire (s) de recherche : AMPERE - Laboratoire AMPERE (UMR5005)

Directeur de thèse : Minh tu PHAM

Co-directeur de thèse : Arnaud LELEVÉ

Président de jury :

Composition du jury :

Pierre Veyres (Rapporteur), Emmanuel Promayon (Rapporteur), Véronique Perdereau, Minh Tu Pham, Damien Ebérard, Arnaud Lelevé