



Contribution à la planification de mouvements en robotique

Michel Taïx

► To cite this version:

Michel Taïx. Contribution à la planification de mouvements en robotique. Robotique [cs.RO]. Université Paul Sabatier - Toulouse III, 2011. <tel-00644801>

HAL Id: tel-00644801

<https://theses.hal.science/tel-00644801v1>

Submitted on 25 Nov 2011

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

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



HAL Authorization

Habilitation à Diriger des Recherches

Préparée au

Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS

Présentée à

l'Université Paul Sabatier, Université de Toulouse

par

Michel TAÏX

Maître de Conférences à l'Université Paul Sabatier de Toulouse

CONTRIBUTION A LA PLANIFICATION DE MOUVEMENTS EN ROBOTIQUE

Soutenue le mardi 22 mars 2011 devant le jury composé de :

Michel COURDESSES	Professeur, Université Toulouse III, LAAS-CNRS	Président
Seth HUTCHINSON	Professeur, University of Illinois, Urbana-Champaign	Rapporteur
Christian LAUGIER	Directeur de Recherche, INRIA Grenoble	Rapporteur
Véronique PERDEREAU	Professeur, Université Paris VI, ISIR	Rapporteur
Philippe FRAISSE	Professeur, Université Montpellier II, LIRMM	Examineur
Jean-Paul LAUMOND	Directeur de Recherche, LAAS-CNRS Toulouse	Garant

Remerciements

Je remercie les directeurs successifs du LAAS-CNRS, laboratoire dans lequel mes travaux ont été réalisés, pour faire vivre cette structure et m'y avoir accueilli depuis ma thèse. Merci également à tous les personnels des services du laboratoire car ils nous donnent les moyens de travailler dans de bonnes conditions.

J'ai débuté au LAAS-CNRS par ma thèse, dans le groupe RIA. Depuis, il a évolué en plusieurs groupes de recherche. Que tous les responsables de ces groupes soient remerciés pour avoir su dynamiser les recherches en robotique.

Je tiens à exprimer ma gratitude aux rapporteurs de mon mémoire d'habilitation à diriger les recherches pour la caution qu'ils ont bien voulu apporter à mes travaux :

Seth HUTCHINSON, Professeur de l'University of Illinois,

Christian LAUGIER, Directeur de Recherche à l'INRIA de Grenoble,

Véronique PERDEREAU, Professeur d'Université à Paris VI- ISIR

Leurs conseils et leurs questions ont été source d'enrichissement.

Mes remerciements vont également à Philippe FRAISSE, Professeur d'Université à Montpellier II-LIRMM qui m'a fait l'honneur et l'amitié de participer à ce jury.

Une HDR est le reflet d'un travail personnel qui est le fruit de longues discussions et séances de travail avec de nombreux collègues. Il ne m'est pas possible de faire une liste exhaustive de tous ceux qui m'ont permis de réaliser ces travaux. Cependant, je ne peux passer sous silence quelques personnes qui ont joué un rôle déterminant dans ma carrière.

Jean-Paul LAUMOND m'a accueilli lorsque je suis arrivé au LAAS. J'ai été ton premier doctorant. Tu m'as initié à la problématique de la planification de trajectoires non-holonyme et tu as su me donner les bons conseils au bon moment. Pour tout cela, je te suis reconnaissant.

Avec toi, Michel COURDESSES, parce que tu m'as permis de faire des vacances, j'ai découvert les multiples facettes du métier d'enseignant-chercheur. Je te remercie pour le temps et les conseils que tu m'as donnés tout au long de ces nombreuses années et pour m'avoir donné l'envie de faire ce métier si exigeant mais tellement riche.

Si cette HDR a vu le jour, c'est aussi grâce à mon collègue de bureau et ami, Philippe. J'ai toujours plaisir à échanger avec toi, que ce soit lors de nos séances de travail au tableau ou pendant nos discussions sur tout ce qui fait le sel de la vie. Merci pour ces moments et pour m'avoir encouragé à mener à bien ce travail.

Bien entendu, je n'oublie pas mes collègues roboticiens, présents ou anciens, Gepettistes ou non, ni mes collègues de l'UPS et de l'AIP, et plus particulièrement l'équipe "Systèmes Intelligents". Il m'est toujours aussi agréable de travailler avec vous, que vous soyez du LAAS ou de l'IRIT. Tous les étudiants de Master ou de Doctorat que j'ai encadrés se reconnaîtront dans ce mémoire. Je les remercie et je les félicite pour le travail accompli ensemble.

Un grand merci à ma famille et à ma femme qui a su supporter mes moments d'absence et de doute.

Table des matières

Préambule	1
I Informations générales	3
1 Curriculum Vitae	5
1.1 Etat Civil	5
1.2 Titres et Diplômes	5
1.3 Fonctions exercées	6
1.4 Résumé de mon parcours à l'université Paul Sabatier	7
2 Activités d'encadrement	9
2.1 Encadrement de thèses d'Université	9
2.2 Encadrement de Master 2 Recherche	10
2.3 Encadrement de Master 2 Professionnel	11
3 Participation à des jurys	13
4 Projets de recherche et valorisation	15
4.1 Projets de recherche	15
4.2 Valorisation industrielle	17
4.3 Animation scientifique	17
5 Activités d'enseignement	19
6 Responsabilités pédagogiques et administratives	27

II	Activités de recherche	33
1	Introduction	35
1.1	Le problème de la planification de trajectoires	36
1.2	Objectifs de mes travaux	37
1.2.1	Planification de trajectoires pour les robots à roues	38
1.2.2	De la planification au mouvement humain	40
2	Planification de trajectoires en robotique mobile	43
2.1	Planification de trajectoires pour robots non-holonomes	43
2.1.1	Robot non-holonyme	43
2.1.2	Planificateur exact itératif	44
2.1.3	Conclusion	46
2.2	Planification en terrain accidenté	47
2.2.1	Planificateur pour des robots articulés	48
2.2.2	Planification robuste de trajectoires	49
2.2.2.1	Prise en compte des incertitudes	50
2.2.2.2	Prise en compte des amers	50
2.2.3	Résultats et conclusion	51
2.3	Planification référencée capteurs	53
2.3.1	Définition d'une planification référencée amers	54
2.3.2	Application	56
2.3.3	Résultats et conclusion	58
2.4	Recouvrement de surface	60
2.4.1	Formalisation du problème	61
2.4.2	Cas des champs convexes	62
2.4.3	Prise en compte des obstacles	63
2.4.4	Cas des champs non convexes	64
2.4.5	Conclusion	66
2.5	Bilan	66
3	De la planification au mouvement humain	69
3.1	Planification probabiliste	71
3.1.1	Arbres locaux et notion de visibilité	72
3.1.2	Principes de l'algorithme VIS-LT	74
3.1.3	Résultats et conclusion	74
3.2	Planification interactive	77

3.2.1	Planification et Interaction pour l'aide au mouvement	78
3.2.2	Planification interactive	79
3.2.2.1	En environnement immersif	79
3.2.2.2	Dans un contexte de bureau d'étude	80
3.2.3	Conclusion	82
3.3	Mouvement naturel et robotique	84
3.3.1	Enchaînement de tâches et primitives sensorimotrices	84
3.3.2	Commande de mouvement biologiquement inspirée	85
3.3.3	Mouvement d'atteinte visuellement guidé	87
3.3.4	Intérêt des primitives motrices	89
3.3.5	Conclusion	91
3.4	Bilan	92
4	Prospectives	95
4.1	Planification d'objets souples	96
4.2	Prise en compte de l'humain	101
4.2.1	Planification de mouvements réalistes	101
4.2.2	Planification et contrôle moteur humain	103
	Références	105
5	Publications	117
III	Les cinq publications jugées essentielles	121

Préambule

Ce document, destiné aux rapporteurs et aux membres du jury, présente ma contribution à la planification de trajectoire et apporte une vue globale sur ma carrière d'enseignant-chercheur à l'université Paul Sabatier.

Il est organisé en trois parties :

- Une présentation générale qui inclut un CV, mes implications en tant que chercheur et la description de mes activités d'enseignant-chercheur.
- Une présentation de l'ensemble de mes activités de recherche que j'ai choisi de décrire en mettant en avant la continuité de mon travail et son évolution dans le temps.
- La liste des cinq publications jugées essentielles qui permet d'approfondir certains points de la deuxième partie.

La première partie, axée sur des informations générales, présente l'ensemble de mes activités depuis le début de ma carrière. Elle ne concerne pas directement le partage des connaissances au sujet de ma contribution à la problématique scientifique de la planification de trajectoire. Cependant, il me semble indispensable de l'intégrer à ce document car mes activités d'enseignement et de responsabilités pédagogiques ou administratives sont implicitement liées à mes activités de recherche. Passer sous silence cette part importante de mon métier d'enseignant-chercheur ne permettrait pas de donner une vue juste et globale sur ma carrière.

La deuxième partie, qui ne traite que des activités de recherche est *self content*, elle constitue l'intégralité de la contribution scientifique.

I

Informations générales

Partie 1

Curriculum Vitae

1.1 Etat Civil

Nom : **TAÏX**

Prénoms : **Michel**, *Jean, Alain*

Date et lieu de naissance : *27 janvier 1961 à Lourdes (65)*

Nationalité : *Française*

Etat civil : *Marié, 3 enfants*

Adresse personnelle : *3, rue du Soleil Levant, 31320 PECHABOU*

Adresse professionnelle : *Laboratoire d'Analyse et d'Architecture des Systèmes*

LAAS-CNRS

7, avenue du Colonel Roche

31077 TOULOUSE CEDEX

Tél. : 05 61 33 78 31

Courriel : taix@laas.fr

1.2 Titres et Diplômes

- **Baccalauréat** Série C, 1979.
- **Maîtrise d'Electronique-Electrotechnique-Automatique** de l'Université des Sciences et Techniques du Languedoc, Montpellier, juin 1983.
- **Diplôme d'Ingénieur** de l'Institut National des Sciences Appliquées (INSA), Option Génie Electrique, Toulouse, juin 1985.

- **Diplôme d’Etudes Approfondies** en Automatique (mention AB), Université Paul Sabatier, Toulouse, juin 1985.
- **Doctorat de l’Université Paul Sabatier**, spécialité robotique, “*Planification de mouvement pour robot mobile non-holonyme* ” Toulouse, le 23 janvier 1991 (mention très honorable).

Jury :

Jean-Daniel Boissonnat	rapporteur	DR INRIA	Sophia-Antipolis
Marc Courvoisier	examineur	PU 61	UPS, Toulouse
Georges Giralt	président	DR CNRS	LAAS, Toulouse
Jean-Claude Latombe	rapporteur	DR	Stanford University
Jean-Louis Lacombe	examineur	Matra-MS	Toulouse
Jean-Paul Laumond	dir. de thèse	DR CNRS	LAAS, Toulouse
Alain Liegeois	rapporteur	PU 61	LIRMM, Montpellier

1.3 Fonctions exercées

- **Chercheur doctorant** allocataire de recherche, Groupe Robotique et Intelligence Artificielle (RIA), Laboratoire d’Automatique et d’Analyse des Systèmes (LAAS-CNRS), Toulouse : sept. 1987 – sept. 1990.
- **Attaché Temporaire d’Enseignement et de Recherche**, Département EEA, UFR Physique Chimie Automatique, Université Paul Sabatier (UPS), Toulouse : oct. 1990 – sept. 1991.
- **Maître de Conférences** au titre de la 61^{ème} section, UFR Physique-Chimie-Automatique, Département EEA et IUP, UPS, Toulouse, et membre du LAAS-CNRS : depuis oct. 1991.
- **Enseignant-chercheur** au sein du groupe RIA, LAAS-CNRS, Toulouse : oct. 1991 – nov. 2007.
- **Enseignant-chercheur** au sein du groupe Gepetto, LAAS-CNRS, Toulouse : depuis nov. 2007.
- **Co-directeur** du Diplôme de Recherche Technologique ISIA *Ingénierie des Systèmes Informatiques et Automatiques*, UPS, Toulouse : sept. 2000 – sept 2006.
- **Directeur** de l’IUP Systèmes Intelligents, UPS, Toulouse : depuis sept. 2000.
- **Porteur** du projet de création du département *Systèmes et Microsystèmes Intelligents* dans le cadre de la création de l’école d’ingénieurs UPSSITECH : depuis sept. 2009.

1.4 Résumé de mon parcours à l'université Paul Sabatier

A l'issue de mon DEA et après l'obtention de mon diplôme d'ingénieur, je suis parti deux années au Gabon dans le cadre du VSNA. J'ai eu l'opportunité d'enseigner dans deux établissements d'enseignement supérieur. Bien que les conditions de travail ne soient pas comparables à la France – il n'existait pas de laboratoire de recherche – cette expérience m'a fait découvrir l'enseignement supérieur et m'a donné l'envie d'expérimenter le métier de la recherche.

En cours de dernière année, j'ai candidaté et obtenu une bourse de docteur ingénieur qui m'a permis de débiter mon doctorat dès mon retour.

Durant la préparation du doctorat, j'ai effectué des enseignements (vacataire puis moniteur) dans le but de poursuivre dans une carrière d'enseignant-chercheur. A l'issue de mon poste d'ATER, j'ai candidaté sur des postes de maître de conférences et j'ai obtenu un poste à l'UPS en 61^{ème} section (classé aussi 1^{er} à l'USTL de Montpellier sur un poste en 61^{ème} section).

Mon activité de recherche a débuté au sein du groupe RIA du LAAS-CNRS durant la préparation de mon doctorat. Mes travaux s'inscrivaient dans la problématique des projets du groupe RIA (*Robotique en environnement humain* et *Robotique d'exploration*) autour des différents aspects de la planification de trajectoires dont la question clé est : *comment calculer une trajectoire géométrique sans collision entre deux situations pour effectuer une tâche spécifique ?* Ce fil conducteur m'a amené à étudier ensuite différentes facettes du problème de la planification, allant des aspects géométriques à la prise en compte des contraintes capteurs, pour me poser des questions plus fondamentales sur la problématique du mouvement sur des systèmes antro-pomorphes.

J'enseigne depuis près de vingt ans dans le cadre de la 61^{ème} section à l'université Paul Sabatier depuis mon recrutement en tant que Maître de conférences. J'ai successivement participé à des enseignements d'Automatique, sur les systèmes linéaires, échantillonnés et non-linéaires, de Traitement du Signal, d'Informatique Industrielle, de Modélisation de Robots manipulateurs, de Robotique Mobile, de Théorie des Graphes, de Programmation Linéaire et de Planification de Mouvement.

En plus de mes activités de recherche et d'enseignement, je me suis rapidement investi au sein de l'UPS. En 1994, j'ai monté et mis en place l'année de maîtrise de l'IUP SI lors de sa création, et, en 2000, j'ai pris la direction des trois années de ce parcours de formation.

Partie 2

Activités d'encadrement

- Depuis 1994 : 5 co-directions de thèse de doctorat dont une en cours,
- Depuis 1993 : 10 directions ou co-directions de Master^a 2 Recherche ou DEA,
- Depuis 1992 : 5 directions ou co-directions de Master 2 Professionnel ou stages de fin d'études.

^aLes stages de Master 2 ont une durée moyenne de 5 mois

2.1 Encadrement de thèses d'Université

◇ 1994-1998 : [TH1] Alain Haït, *Algorithmes pour la planification de trajectoires robustes d'un robot mobile autonome sur un terrain accidenté*,

Encadrement à 50% en co-direction avec Thierry Siméon.

Soutenue le 8 juillet 1998. Mention très honorable.

Publications associées : [3, 24, 25]¹.

Actuellement, Alain Haït est maître de conférences.

◇ 2002-2005 : [TH2] Abeb Malti, *Navigation sensorielle en robotique mobile*,

Encadrement à 50% en co-direction avec Florent Lamiraux.

Soutenue le 21 décembre 2005. Mention très honorable.

Publications associées : [2, 18, 20, 21].

Actuellement, Abeb Malti occupe un poste de post-doctorant.

◇ 2006-2009 : [TH3] Minh Tran Tuan, *Approche neuro-robotique pour le contrôle des systèmes anthropomorphes*,

¹Numéro de publication correspondant de la section 5

Encadrement à 50% en co-direction avec Philippe Souères.

Soutenue le 26 novembre 2009.

Publications associées : [8, 10, 13, 15, 17].

Actuellement, Minh Tran Tuan occupe un poste d'ingénieur en entreprise.

◇ 2007-2010 : [TH4] David Flavigné, *Planification de mouvement Interactive : Coopération Humain-Machine pour la recherche de trajectoires et l'animation*,

Encadrement à 90% en co-direction avec Jean-Paul Laumond.

Soutenue le 22 septembre 2010.

Publications associées : [7, 9, 11, 12].

Actuellement, David Flavigné occupe un poste de post-doctorant.

◇ 2010-2013 : [TH5] Antonio El Khoury, *Planification de mouvements pour robots humanoïdes*,

Encadrement à 50% en co-direction avec Florent Lamiriaux.

Début de thèse le 11 octobre 2010.

En 1992, j'ai participé à l'encadrement de la thèse de Philippe Juhel alors doctorant dans le groupe RIA du LAAS-CNRS. Mon implication s'est concrétisée par les publications [26, 6].

2.2 Encadrement de Master 2 Recherche

◇ 1993 : [MR1] Didier Gazen, UPS, *Etude d'un algorithme incrémental pour la planification de trajectoires*.

Encadrement à 100%.

◇ 1994 : [MR2] Alain Haït, UPS, *Stratégie de planification pour un robot mobile*.

Encadrement à 100%.

◇ 2001 : [MR3] Sébastien Fabre, ENSAE, *Planification de trajectoires pour engins agricoles*.

Co-encadrement à 50% avec P. Souères.

◇ 2002 :

– [MR4] Thomas Godart, ENSAM, *Planification sensorielle en robotique mobile*.

Co-encadrement à 50% avec F. Lamiriaux.

– [MR5] Hélène Frayssinet, INSAT, *Planification de trajectoires pour engins agricoles : cas pratiques*.

Co-encadrement à 50% avec P. Souères.

◇ 2004 : [MR6] Minh Tran Tuan, IFI, *Planification de trajectoires pour engins agricoles : cas des champs non convexes*.

Co-encadrement à 50% avec P. Souères.

◇ 2006 : [MR7] Brice Burger, ENSI Bourges, *Planification de mouvements référencés capteurs*.

Encadrement à 100%.

◇ 2007 :

- [MR8] Luis Delgado, ENSAM, *Planification de mouvements pour tâches coopératives entre plusieurs mannequins virtuels*.
Co-encadrement à 75% avec J.P. Laumond.
- [MR9] David Flavigné, UPS, *Planification de mouvements probabiliste et interactive*.
Encadrement à 100%.

◇ 2010 : [MR10] Antonio ElKhoury, EMP, *Interactive motion planning for digital actors*.
Encadrement à 100%.

2.3 Encadrement de Master 2 Professionnel

◇ 1997 : [MP1] Guilhem Roques (ESIEE)
Commande d'un bras manipulateur par vision, AIP, Toulouse.
Co-encadrement à 50% avec J.Y. Fourquet.

◇ 2001 : [MP2] Franck Plisson (I3L)
Planification de trajectoires avec contraintes perceptuelles, LAAS-CNRS, Toulouse.
Encadrement à 100%.

◇ 2003 : [MP3] Stéphane Mas (IUP SI)
Navigation sur amers visuels pour des robots mobiles, LAAS-CNRS, Toulouse.
Co-encadrement à 50% avec F. Lerasle.

◇ 2004 : [MP4] Luc PRUNET (DESS CAMSI)
Intégration de fonctions de navigation visuelle en robotique mobile, LAAS-CNRS, Toulouse,
Co-encadrement à 50% avec F. Lerasle

◇ 2005 : [MP5] Francisco Menini (IUP SI)
Planification et contrôle de mouvements pour des robots manipulateurs industriels, LAAS-CNRS, Toulouse.
Encadrement à 100%.

Tutorat de stages de fin d'études

Depuis 1995, chaque année je suis tuteur de trois ou quatre stagiaires pour des stages de niveau Master 1 ou 2 de l'IUP SI et du Master 2 IRR.

Partie 3

Participation à des jurys

Jurys de Doctorat

◇ David Flavigné :

Planification de mouvement interactive : coopération Humain-Machine pour la recherche de trajectoire et l'animation.

Doctorat de l'université Paul Sabatier, Toulouse III, 22 septembre 2010.

Qualité : **co-directeur de thèse.**

◇ Nicolas Ladeveze :

Apports des méthodes de planification automatique dans des simulations interactives d'industrialisation et de maintenance en réalité virtuelle.

Doctorat de l'INP Toulouse - ENIT, 19 avril 2010.

Qualité : **examineur.**

◇ Minh Tran TUAN :

Approche neuro-robotique pour le contrôle des systèmes anthropomorphes.

Doctorat de l'Université Paul Sabatier, LAAS-CNRS, 26 novembre 2009.

Qualité : **co-directeur de thèse.**

◇ Abed MALTI :

Navigation sensorielle en robotique mobile.

Doctorat de l'Université Paul Sabatier, LAAS-CNRS, 21 décembre 2005.

Qualité : **co-directeur de thèse.**

◇ Alain HAÏT :

Algorithmes pour la planification de trajectoires robustes d'un robot mobile autonome sur un terrain accidenté.

Doctorat de l'Université Paul Sabatier, LAAS-CNRS, 8 juillet 1998.

Qualité : **co-directeur de thèse.**

Jurys de Diplôme de Recherche Technologique

◇ Sylvain PEYRUQUEOU :

Approche et applications de l'ingénierie des systèmes multi-agents adaptatifs.

DRT ISIA de l'Université Paul Sabatier, IRIT, septembre 2002.

Qualité : **examineur**.

◇ David DAURENJOU :

Détection d'obstacles par un système de vision monoculaire en environnement autoroutier.

DRT ISIA de l'Université Paul Sabatier, LAAS-CNRS, septembre 2002.

Qualité : **examineur**.

◇ Thomas SONTHEIMER :

STAFF : un modèle adaptatif de prévision de crues.

DRT ISIA de l'Université Paul Sabatier, IRIT, janvier 2002.

Qualité : **membre invité**.

◇ Florent FIGUET :

Simulation du rayonnement électromagnétique dans des scènes virtuelles 3D par lancer de rayons.

DRT ISIA de l'Université Paul Sabatier, IRIT, décembre 2003 .

Qualité : **examineur**.

◇ Lilian MORENO :

Modélisation corporelle pour le suivi de lésions dermatologiques : étude des évolutions technologiques.

DRT ISIA de l'Université Paul Sabatier, IRIT, décembre 2003.

Qualité : **examineur**.

Partie 4

Projets de recherche et valorisation

4.1 Projets de recherche

◇ **Responsable LAAS du projet AMSI (2007-2009) :** *Algorithmique du Mouvement et Simulation Interactive*, projet ANR-RNTL sous la coordination de J. Perret de la société Haption. Partenaires : Haption, KineoCam, CEA-LIST et LAAS-CNRS.

Objectif : proposer une nouvelle solution de recherche de trajectoires pour les problèmes d’assemblage mécanique en faisant coopérer une approche haptique avec une approche *motion planning*.

Implication personnelle : notre contribution concernait la partie la plus innovante du projet, axée sur le développement d’une solution interactive entre un utilisateur et un algorithme de type RRT. Alors que les travaux existants proposent de décomposer le problème en deux étapes, notre méthode permet de générer une boucle générale entre le comportement de l’utilisateur et le développement de l’arbre de recherche de solutions automatique. En plus de l’encadrement scientifique, j’ai participé aux diverses réunions avec nos partenaires et effectué la rédaction des rapports LAAS.

Encadrements liés : thèse [TH4]¹.

◇ **Co-responsable LAAS du sous-projet SP1 du projet PERF-RV 2 (2005-2009) :** sous la responsabilité de J.P. Laumond pour la contribution du LAAS-CNRS. Partenaires : dix partenaires académiques et onze partenaires industriels.

Objectif : démontrer que l’intégration de l’humain virtuel dans l’usine numérique permettra d’améliorer, dès la conception, l’efficacité et l’ergonomie des postes de travail.

Implication personnelle : je me suis impliqué à partir de 2007 dans l’amélioration des techniques de planification probabiliste dans un but d’accroître la convergence des algorithmes de recherche de solutions dans des espaces des configurations fortement contraints par de nombreux passages étroits et d’optimiser les opérations élémentaires de planification de mouvements pour des chaînes mécaniques fermées. En plus de l’encadrement scientifique, j’ai effectué la rédaction des rapports LAAS.

Encadrements liés : stages de Master [MR8] et [MR9].

¹référence de la section 2

◇ **Participation au projet ZEUXIS (2005-2008)** : projet soutenu par la Fondation EADS. Partenaire : LAAS.

Objectif : étudier les principes computationnels du mouvement humain pour proposer des solutions permettant de réduire la complexité combinatoire liée au grand nombre de degrés de liberté du corps humain dans le but de reproduire des mouvements anthropomorphes sur des mannequins numériques ou de commander des robots humanoïdes.

Implication personnelle : mes différents travaux s'incrivent pleinement dans ce projet. Présentation de mes activités lors du workshop de 2007.

Encadrements liés : stage de Master [MR9] et thèse [TH3].

◇ **Participation au pré-projet puis au projet EGOCENTRE (2001-2005)** : *Enchaînement de tâches référencées multi-capteurs pour l'exécution de mouvements complexes en robotique mobile*, projet RobEA sous la coordination de P. Souères du LAAS-CNRS. Partenaires : IRISA, LGP-ENIT, CERCO et LAAS-CNRS.

Objectif : Etude de la modélisation et l'exécution de mouvements complexes à partir de tâches référencées multi-capteurs en nous inspirant des mécanismes neuronaux.

Implication personnelle : j'ai participé à l'élaboration du pré-projet et du projet lors de sa définition. Mon implication scientifique concernait le lien entre les méthodes globales de planification et les tâches locales de régulation de type sensori-motrice.

Encadrements liés : stages de Master [MR4], [MR7] et en partie la thèse [TH2].

◇ **Participation à l'ACI De la perception à la commande : une histoire de repères (2001-2004)** : programme interdisciplinaire "Cognition et traitement de l'information" sous la coordination de Y. Trotter du CERCO. Partenaires : CERCO et LAAS-CNRS.

Objectif : comprendre le lien entre perception et commande au travers des fonctions sensorimotrices chez le vivant et de la commande référencée capteurs en robotique.

Implication personnelle : travail avec les chercheurs du centre de recherche du CERCO (Cerveau et Cognition) afin de pouvoir développer des solutions novatrices en commande de robots.

◇ **Participation au projet PROMotion (1992-1995)** : *Planning Robot Motion* sous la coordination de J.P. Laumond du LAAS/CNRS. Partenaires : Universitat Politecnica de Catalunya (Barcelona), Ecole Normale Supérieure (Paris), Universita Degli Studi di Roma, La Sapienza (Roma), INRIA (Sophia-Antipolis), Utrecht University (Utrecht) et LAAS/CNRS.

Objectif : création d'une synergie entre la robotique, l'automatique, la géométrie algorithmique et algébrique.

Implication personnelle : ce projet m'a permis d'accroître et de diversifier les échanges avec d'autres laboratoires de recherche travaillant sur ce projet lors des différents workshop et d'étendre mon champ de connaissances hors de la planification de trajectoires.

◇ **Programme Européen COMETT (1993)** : *Diffusion des Technologies mises en oeuvre dans l'environnement CIM*. Partenaires : Bruno Gorla, Jean-Yves Fourquet et Michel Taïx.

Objectif : rédiger un document intitulé *Robotics*, présentant les aspects de robotique mobile et de manipulation dans un environnement de CIM.

Implication personnelle : rédaction d'un tiers du document de 95 pages. Avec mon collègue Jean-Yves Fourquet, nous avons dispensé l'enseignement correspondant aux étudiants de l'*Eskola politeknioa* de Mondragon au Pays Basque espagnol durant la même année.

4.2 Valorisation industrielle

En octobre 2000, les responsables du centre de Recherche et Développement de Renault Agriculture nous ont contactés pour un projet de commande automatique de tracteurs agricoles à partir de données GPS. Notre contribution dans ce projet a été de développer un planificateur de trajectoires pour réaliser la couverture complète d'un champ tout en minimisant le recouvrement entre les passages successifs. Cette problématique était motivante à plusieurs titres : elle répondait à une vraie problématique scientifique, elle n'était pas abordée dans mon groupe de recherche RIA, et elle présentait de multiples applications.

Dans le cadre de ce travail, j'ai co-encadré avec Philippe Souères trois stages de recherche [MR3, MR5, MR6] qui ont donné lieu à plusieurs publications [19, 22, 23]. Les solutions et les algorithmes proposés à l'issue des deux premières années ont motivé Renault Agriculture à déposer deux plis d'huissiers afin de protéger nos résultats.

4.3 Animation scientifique

• Séminaires majeurs dans des manifestations scientifiques

- Au niveau **international** : présentation à *ETFA'01*, *FSR'03*, *IROS'04*, *EUROS'08*, *RFIA'10*.
- Au niveau **national** : membre du GDR-Robotique, participation aux Journées Nationales de la Recherche en Robotique (2002, 2005, 2007), aux Journées Nationales de la Robotique Humanoïde (2005, 2007, 2008, 2010) et aux Journées du Programme Interdisciplinaires RobEA, (2002, 2004).
- Au niveau **local** : présentation au workshop annuel du groupe en 2001, 2003, 2007.

• Participation à des comités de lecture

IEEE Int. Conf on Robotics and Automation (2001, 2002, 2003, 2004, 2006, 2007, 2009, 2011), IEEE Int. Conf. on Intelligent Robots and Systems (2001, 2002, 2003, 2007, 2010), Int. Fed of Automatic Control (2002).

Mathematical, Physical and Engineering Sciences, (2002), IEEE Trans. on Robotics and Automation (2002), IEEE Trans. on Robotics (2008, 2009), Journal of Intelligent & Robotic Systems (2002).

Partie 5

Activités d'enseignement

Mon expérience d'enseignement au Gabon, pendant mon service national actif à l'Ecole d'Ingénieurs de Libreville et à l'université de Masoukou, m'a motivé pour assurer des vacances d'enseignement dès 1988, pendant la préparation de mon doctorat, alors que le monitorat n'existait pas encore. Dès la création du monitorat, l'année suivante, j'ai obtenu un poste de moniteur en 61^{ème} section à l'Université Paul Sabatier (UPS), prolongé par un poste d'ATER en 1990.

J'enseigne depuis près de vingt ans dans le cadre de la 61^{ème} section à l'université Paul Sabatier depuis mon recrutement en tant que Maître de conférences. J'ai successivement participé à des enseignements d'Automatique linéaire, échantillonnée et non-linéaire, de Traitement du Signal, d'Informatique Industrielle, de Modélisation de Robots manipulateurs, de Robotique Mobile, de Théorie des Graphes, de Programmation Linéaire et de Planification de Mouvements.

Ces enseignements se sont appuyés sur une activité de recherche et d'encadrement dans le domaine de la Robotique au LAAS-CNRS. Cette expérience intègre un travail pédagogique très différent selon le type d'intervention : cours, travaux dirigés (TD), travaux pratiques (TP), bureaux d'études (BE), projet, travail en équipe, création de cours et de manipulations concrètes et aussi en fonction des différents degrés de maturité scientifique des publics concernés : Licence 1, Licence 3, Master 1 ou Master 2, formation du domaine Automatique (EEA) ou Informatique, formation universitaire classique, professionnalisée ou technologique, école d'ingénieurs.

Ces activités d'enseignement sont exposées dans les pages suivantes, décomposées en trois périodes.

1988 - 1991

Fonction : vacataire, moniteur puis ATER en automatique-robotique.

◇ Maîtrise EEA : TP du module *Automatique 2* : analyse et correction dans l'espace d'état des systèmes linéaires continus et échantillonnés et des systèmes non linéaires.

◇ DESS IRR : BE de robotique : mise en oeuvre des cours théoriques de modélisation de robots manipulateurs et TP sur des robots et des systèmes industriels (SCEMI, Eric, GT6A, Visiomat).

1992 - 1998

Fonction : maître de conférences à l'Université Paul Sabatier.

A l'issue de mon recrutement en qualité de maître de conférences à l'Université Paul Sabatier en octobre 1991, j'ai enseigné essentiellement en DESS IRR et en maîtrise EEA durant les premières années. A partir de 1994, mes interventions ont été étendues à l'IUP SI, formation au sein de laquelle j'avais la responsabilité de la troisième année (maîtrise). J'ai également été amené à intervenir pour des volumes moindres en première année du DEUG Technologie Industrielle et en dernière année à l'ENSEEIH. Ces enseignements peuvent être regroupés sous les mots-clés *Informatique Industrielle, Automatique, Traitement du Signal et Robotique*.

Enseignement en Informatique Industrielle

◇ Maîtrise EEA :

- TD du module *Informatique Industrielle* : analyse et conception, méthode SADT et SA-RT (1992 et 1993)
- TP du module *Informatique Industrielle* : cet enseignement a été mis en place en 1992, l'équipe enseignante a dû monter les nouvelles manipulations (1992-1994). J'ai plus particulièrement participé à une manipulation illustrant les TD portant sur les techniques d'acquisition par scrutation et par interruption.

Enseignement d'Automatique et de Traitement du Signal

◇ Maîtrise EEA :

- TD du module *Automatique 2* : représentation et commande dans l'espace d'état (1992-1995)
- TP des modules *Automatique 2* et *Traitement Numérique du Signal* et *UM3* et *UM6* en fonction des différentes habilitations de diplôme (1991-1996) : systèmes linéaires échantillonnés, systèmes non-linéaires, commande modale, synthèse et réalisation de filtres numériques, identification par SBPA et moindres carrés. J'ai plus particulièrement monté une manipulation sur la commande numérique de bacs d'eau réels et une autre sur l'identification de système par moindres carrés. J'ai aussi participé à la mise en place d'une unité de formation en Traitement du Signal pour la formation continue de l'UPS (1995).

◇ DEUG première année Technologies Industrielles :

- COURS d'introduction à l'Automatique du module *Découverte* : j'ai construit un cours de quatre heures sur la problématique des systèmes automatisés (1993 et 1994) afin de faire découvrir aux étudiants une des facettes des métiers de l'ingénierie.

◇ IUP SI deuxième année (Licence) :

- COURS et TD de *modélisation dans l'espace d'état* du module *Automatique et Traitement du Signal* (1993-1998) : cet enseignement visait principalement à modéliser et à analyser un système continu sous forme d'équations d'état.
- TP du module *Automatique et Traitement du Signal* (1993) : avec mon collègue Jean-Yves Fourquet, nous avons mis en place des séances destinées à illustrer aussi bien l'approche fréquentielle que celle basée sur le formalisme d'état à l'aide d'outils logiciels dédiés.

◇ IUP SI troisième année (Maîtrise) :

- COURS de *commande dans l'espace d'état* du module *Automatique* (1994-1998) : cet enseignement illustre la commande modale et la mise en oeuvre à l'aide d'observateurs.
- TP de *commande dans l'espace d'état* du module *Automatique* (1994) : j'ai développé une manipulation en simulation de type bacs d'eau et mis en oeuvre une manipulation réelle sur une commande par retour d'état avec observateur sur un moteur à courant continu.

Enseignement en Robotique

◇ DESS I.R.R. :

- BE de robotique : l'objectif était de mettre en pratique l'ensemble des enseignements de robotique de la formation dans des manipulations ou des projets sur du matériel industriel à l'Atelier Interuniversitaire de Productique de Toulouse, AIP. Ayant la responsabilité de ces BE qui comportaient un volume horaire important de cent heures, j'ai décidé de les décomposer en trois parties :
 - Le développement d'un projet visant à mettre en oeuvre le cours de modélisation du bras manipulateur sur un robot réel, le GT6A avec l'implémentation en C des modèles et la mise en oeuvre réelle.
 - Une série de TP sur les différents systèmes de l'AIP : manipulation d'objets avec robot SCEMI, programmation robotique sous AML, commande avec robot GT6A, apprentissage et reconnaissance d'objets avec le Visiomat, saisie d'objet par couplage Visiomat et robot Eric. En tant que responsable, j'ai adapté et fait évoluer les différentes manipulations en fonction de l'évolution matérielle de l'AIP (achat et renouvellement) ainsi que du choix des thèmes de manipulation que je voulais mettre en valeur, ce qui impliquait une modification constante du support de T.P. correspondant.
 - La dernière partie avait pour objectif de faire le lien entre la robotique et des disciplines plus fondamentales. J'ai créé et mis en place une manipulation illustrant la problématique de localisation de robot mobile à partir d'informations bruitées en lien avec le traitement du signal. Avec Bruno Gorla de l'IUT A, nous avons monté un mini-projet de douze heures sous Matlab pour illustrer à la fois la génération de mouvements et la commande d'un bras manipulateur.

◇ IUP SI troisième année :

- COURS de robotique mobile : à partir de 1996, j'ai eu la charge de mettre en place seize heures de cours en robotique mobile pour illustrer la problématique de la modélisation, de la commande, de la localisation et de la planification de trajectoires.

◇ Année Spéciale Automatique - IUT GEII :

- COURS et TD d'introduction à la robotique de manipulation : sensibiliser les étudiants aux problèmes posés par l'utilisation de bras manipulateur, faire le lien avec leur formation en automatique et introduire la modélisation géométrique.

◇ ENSEEIH :

- TP de systèmes robotiques en dernière année filière *automatique* : les manipulations résultaient d'une adaptation des manipulations que j'avais montées en DESS IRR, étant donné le faible volume horaire d'enseignement en robotique de cette formation.

Depuis 1999

Fonction : maître de conférences à l'Université Paul Sabatier.

A la rentrée 1999, Maurice Briot, directeur de l'IUP *Systèmes Intelligents* (SI), a annoncé à l'équipe pédagogique son souhait de laisser la direction en raison de son départ à la retraite imminent. Il m'a proposé de m'investir davantage dans cette formation afin d'assurer la continuité. Après l'appel de candidature en septembre 2000, j'ai été élu directeur de cet IUP. J'ai donc recentré mes enseignements principalement à l'IUP SI, formation au sein de laquelle je m'étais déjà fortement investi tout en intervenant en DESS IRR. J'ai également été amené à intervenir dans une moindre mesure en Licence 1 lors de la mise en place de la nouvelle habilitation en septembre 2003, ainsi qu'aux formations de l'ISIM à Toulouse.

Ces enseignements peuvent être regroupés sous les mots-clés *Automatique, Traitement du Signal et Robotique*.

Enseignement d'Automatique et Traitement du Signal

◇ Licence 1 :

- TP d'introduction à l'Automatique de l'unité de valeur d'orientation au premier semestre : l'objectif de cette manipulation était de faire comprendre aux étudiants la notion de boucle ouverte et boucle fermée avec mise en oeuvre sur un robot Peeke. Après mon investissement pour participer à l'élaboration de cette manipulation pendant l'année de mise en place de la nouvelle habilitation dans le schéma LMD, mes différentes charges m'ont amené à laisser cet enseignement (2003).

◇ Maîtrise puis Master 1 IUP SI :

- COURS de *commande dans l'espace d'état* du module *Automatique* : cet enseignement, qui illustrait jusqu'alors la commande modale et la mise en oeuvre à l'aide d'observateurs, a été modifié avec celui de l'IUP SI deuxième année. La commande modale a été introduite l'année précédente et j'ai redéfini ce cours pour y introduire la commande modale numérique ainsi que les aspects de mise en oeuvre sur ordinateur. En tant que responsable de cet enseignement, j'ai participé à la définition des TP associés.
- TP du cours d'estimation du module *Traitement du signal* : j'ai monté un TP pour illustrer le cours d'estimation basée sur la localisation de robot sur des balises artificielles à l'aide de moindres carrés.
- TD et TP de programmation linéaire dans l'unité de valeur *Systèmes robotiques* suite à la mise en place de la nouvelle maquette de formation en 2004. Cet enseignement n'existant pas dans l'ancienne habilitation, j'ai monté un ensemble de TD et TP pour illustrer le cours centré sur l'algorithme du simplexe.

Enseignement en lien avec la Robotique

◇ DESS puis Master 2 I.R.R. :

- BE de robotique : lors de la nouvelle maquette d’habilitation de 1999, nous avons décidé avec Michel Courdresses d’introduire des BE de robotique mobile avec une mise en oeuvre sur trois robots mobiles Scout de la société *Nomadic Inc.* achetés en 1999 par l’AIP. Etant le seul enseignant à monter ce projet, j’ai décidé de laisser les parties des BE portant sur les bras manipulateurs à ma collègue Viviane Cadenat et de prendre seulement en charge la partie robotique mobile.

Le but de ce bureau d’étude est de permettre à un robot mobile de réaliser une tâche de déplacement de manière autonome. Il s’agit de se déplacer dans un environnement connu après une localisation relative sur des amers de l’environnement. Pendant le déplacement du robot, des obstacles imprévus peuvent se trouver sur sa trajectoire.

Deux problèmes principaux sont abordés :

- Le problème de la localisation en position d’un robot mobile. Ce problème, essentiel dans le domaine de la robotique mobile, se pose dès lors que l’on envisage de faire naviguer un véhicule dans un environnement donné. Nous nous intéressons plus particulièrement à une technique basée sur la méthode des moindres carrés.
- Le problème de l’évitement d’obstacles imprévus. Lors de l’exécution de la trajectoire dans un environnement domestique, certains objets sont détectés par les capteurs de proximité et le robot doit les éviter pour poursuivre sa mission.

De plus, la mise en oeuvre réelle sur un robot permet aux étudiants de se confronter à la difficulté expérimentale et de bien percevoir la différence entre simulation et implémentation réelle. Au cours de l’année 2006, nous avons porté ce BE sur une nouvelle gamme de robots mobiles de la société Festo.

- COURS de robotique mobile : j’ai mis en place ce cours en m’appuyant sur celui que j’ai monté dans le module *Robotique et Productique* en IUP SI troisième année mais en l’adaptant au public du DESS qui provient d’horizons différents (maîtrise informatique, automatique, mécanique, ...)
- COURS de graphes : avec Claudette Cayrol, nous avons monté un cours de douze heures s’adressant à des non-informaticiens. Je suis en charge de la partie recherche de chemins pour un volume de six heures depuis 2007.

◇ Licence 3 IUP SI :

- COURS et TD en *modélisation et commande des robots manipulateurs* dans le module *Robotique* : cet enseignement était assuré par mon collègue Jean-Yves Fourquet, avant son départ de l’UPS, nous en avons redéfini le contenu ensemble. Il se décomposait en 23 heures de COURS, 23 heures de TD et 42 heures de BE. J’assurais la majorité des cours et TD avec ma collègue Viviane Cadenat. Outre les heures en présence des étudiants, la mise en place de cet enseignement m’a amené chaque année à redéfinir le projet que les étudiants avaient à réaliser, choisir les thèmes des manipulations qui se déroulaient en alternance avec le projet, et modifier le polycopié correspondant. Cet enseignement a complètement évolué avec la mise en place de la nouvelle habilitation LMD en 2003 puisque le volume horaire du thème *modélisation des robots manipulateurs* n’est plus que de soixante heures.

- COURS , TD et TP en *génération de mouvement pour des bras manipulateurs* : mon intervention illustre la génération de trajectoires dans l'espace opérationnel à vitesse imposée d'un robot RP.
- ◇ Maîtrise puis Master 1 IUP SI :
 - COURS de robotique mobile dans le module *Robotique et Productique* : avec l'évolution de la robotique mobile et la mise en place de la nouvelle habilitation, j'ai complètement refondu mon ancien cours pour illustrer l'ensemble de la thématique : modélisation de plateforme mobile, commande de robot mobile par technique linéarisante, localisation, planification de trajectoires et architectures informatiques générales.
 - Projet de T.E.R en robotique : j'ai proposé divers sujets à des groupes de quatre à six étudiants dans le but d'appréhender le travail en équipe et de se focaliser sur un point de recherche lié à leur cours.
- ◇ D.E.S.S Systèmes Intelligents
 - COURS de robotique avancée : ce cours dont la thématique change chaque année ("Planification avec incertitudes, mouvements référencés capteurs en robotique ou télérobotique") de quatre à six heures a pour objectif de montrer le lien entre les techniques avancées en robotique et les systèmes interactifs et coopératifs (1999-2007).
- ◇ Master 2 Ingénierie des Systèmes Intelligents
 - COURS de planification de trajectoires : ce cours vise à montrer d'une part les liens entre les algorithmes de planification et la robotique, et d'autre part les possibilités d'utilisation dans d'autres contextes (depuis 2007).
 - Projet de Grande Envergure (PGE) : suivi des étudiants sur les cinq mois du projet allant de la définition du cahier des charges de l'industriel jusqu'à la livraison (depuis 2006). Le PGE est un dispositif pédagogique unique car tous les étudiants travaillent sur le même projet, en même temps, pour faire face aux exigences d'un cahier des charges proposé par un véritable client industriel. D'octobre à janvier, les étudiants doivent, en parallèle de leur cours, avec les conseils et critiques des encadrants du PGE (industriels et enseignants-chercheurs), produire une conception détaillée de leur système. En février, les étudiants consacrent la totalité de leur temps sur leur PGE pour effectuer la livraison au client avant leur départ en stage.
- ◇ Master 2 Engineering of Intelligent Systems : avec mes collègues Patrick Danès et Viviane Cadenat nous avons monté un enseignement intégré de quatre-vingt-deux heures en langue anglaise dans le cadre d'un Master international franco-tchèque qui s'adresse à un public très diversifié (depuis 2009). J'assume une partie des cours de robotique pour un volume de vingt heures.
- ◇ Institut Supérieur d'Ingénieurs de Montpellier : l'ISIM envoie à l'AIP ses étudiants de dernière année de formation en stage professionnel pendant une semaine à temps complet. Nous avons donc proposé un projet professionnel de formation qui a évolué au fil des ans.
 - Projet de robotique et vision : ce projet avait pour objectif de montrer l'intérêt de la vision industrielle dans la réalisation d'une tâche robotique industrielle (1999 et 2000).
 - Projet de navigation de robotique mobile : suite à l'évolution de la formation ISIM, j'ai monté un projet de robotique mobile autour du transport de pièces pour illustrer les deux principales

thématiques que sont la planification de trajectoires et la localisation.

◇ Ecole des Mines d'Albi :

- COURS d'introduction à la robotique mobile en deuxième année de la filière *Génie Industriel* : avec mon collègue Philippe Souères, nous avons monté et partagé un COURS de quinze heures adapté à cette formation à finalité non robotique.

Partie 6

Responsabilités pédagogiques et administratives

- **Directeur** du parcours IUP *Systèmes Intelligents* (SI), L3, M1 et M2 (depuis janvier 2000).
- Responsable de l'année de maîtrise IUP SI (1993-1998).
- Responsable des modules Automatique et Robotique en Licence IUP SI (1993-1998).
- Co-directeur du Diplôme de Recherche Technologique ISIA *Ingénierie des Systèmes Informatiques et Automatiques*, responsable de l'option *Systèmes Intelligents* (2000-2007).
- Responsable en Master 2 SI des UE *Formation générale aux SI* (depuis 1999).
- Responsable des bureaux d'études du DESS IRR (1991-1998).
- Membre du Conseil de Perfectionnement de l'IUP SI (depuis 1993).
- Membre du jury de recrutement de l'IUP SI (depuis 1993) et président de jury (depuis 2000).
- Responsable du recrutement en ligne au département Inter-UFR d'Ingénierie (depuis 2004).
- Co-créateur et membre du réseau de DESS IUP STIC (de 2001 à 2003).
- Membre de l'Association des Directeurs d'IUP (depuis 2000).
- Membre du CA du Groupement d'Innovation pour l'Industrie, GIPI (depuis 2006).
- Membre des Collèges Scientifiques : élu du collège scientifique EEA de l'UPS (2009-2012), membre nommé du comité de sélection de l'Ecole Nationale d'Ingénieurs de Tarbes, ENIT, (2009-2012).
- Membre des Commissions de spécialistes : 63^{ème} section (1995-1998) et 61^{ème} section (1995-2004) de l'Université Paul Sabatier. 61^{ème} section (1995-1998 et 2001-2007) de l'Université de Perpignan. Commission mixte de l'ENIT (2001-2004).
- Président du Groupe d'Avancement et de Promotion des sections 60-61-62-63 à l'UPS (2009-2012).
- Membre nommé du Conseil du pôle RIA du LAAS-CNRS (2007-2011).
- Membre nommé de la Commission Enseignement-Recherche du laboratoire (2006-2009).

A l'IUP SI

A partir de 1993

Pendant toute cette période, l'essentiel de mes responsabilités administratives a été lié à l'IUP SI. L'IUP SI, créé par Maurice Briot et dirigé par lui jusqu'à la rentrée 2000, est situé à la charnière entre les parcours classiques d'Informatique et EEA. Suite à mon implication initiale en DESS IRR, formation courte à double compétence, Maurice Briot m'a associé à la réflexion de cette création. Le point clef d'un tel cursus est de proposer une formation professionnalisante longue sur trois ans, bien ancrée avec la réalité du monde industriel. Un des éléments permettant d'atteindre cet objectif est son Conseil de Perfectionnement. Composé à parité d'universitaires et d'industriels, il se réunit régulièrement afin de définir la politique générale et est présidé par un industriel (actuellement Patrice Berranger, Directeur de la société *Magellium*) Cette instance participe au recrutement des étudiants et délivrait jusqu'en 2004 le diplôme d'Ingénieur Maître, sur proposition d'un jury d'enseignants.

L'IUP SI a accueilli ses premiers étudiants à la rentrée 1992-93, et c'est au début de 1993 que Maurice Briot m'a demandé de prendre la responsabilité de la troisième année (niveau maîtrise) et d'en assurer la mise en place. A partir de cette date, j'ai participé à la définition précise des contenus et des équilibres entre la deuxième et la troisième année, plus particulièrement en ce qui concerne les enseignements d'automatique et de robotique. A la rentrée 1994-95, la première promotion est entrée en troisième année. J'ai assumé cette responsabilité jusqu'en juillet 1998. Cette formation proposait des enseignements communs avec la maîtrise EEA et la maîtrise de l'IUP Ingénierie des Systèmes Informatiques, ainsi que des enseignements spécifiques dans les disciplines des métiers préparés par l'IUP SI. L'imbrication de ces trois formations induisait des contraintes fortes et a nécessité un effort de communication important entre les différents intervenants : secrétariat, UFR, enseignants et étudiants.

Mon action portait sur de nombreux aspects :

- définition des contenus,
- organisation des enseignements et des emplois du temps.
- organisation du contrôle des connaissances,
- participation au conseil de perfectionnement,
- communication entre les différents intervenants.

En outre, je participais et participe toujours chaque année aux deux phases¹ du recrutement des étudiants en première année ou en entrée parallèle en deuxième année.

Lors des différentes campagnes d'habilitation, j'ai assuré, en collaboration avec mes collègues Jean-Yves Fourquet et Cyril Briand, à la refonte des programmes d'Automatique, Robotique et Productique en deuxième et troisième année.

Durant l'année 1997-1998, le conseil de perfectionnement a décidé de créer un DESS SI spécifique pour amener nos étudiants au niveau Bac+5. Nous avons élaboré avec Maurice Briot et Cyril

¹Les étudiants sont recrutés à l'issue d'un premier tri sur dossier, puis d'une audition devant un jury *mixte* enseignant-industriel.

Briant le programme de cette formation qui a été mise en place en 1999 après habilitation par le ministère.

Outre la responsabilité de la troisième année, j'étais également responsable des modules et principalement du module 6 (Robotique et Productique) de troisième année (1994-1998).

Depuis 2000

Le conseil de perfectionnement de l'IUP SI m'a élu directeur de cet IUP en septembre 2000. Une de mes premières démarches a été de développer les liens avec le tissu industriel, avec nos partenaires du Conseil de Perfectionnement, avec le réseau d'entreprises du GIPI et les sociétés accueillant nos stagiaires.

En 2002, nous avons décidé au sein du département d'Ingénierie, département qui coordonne le rôle des douze IUP de l'UPS, de proposer une maquette d'habilitation dans le cadre du schéma LMD. Il a donc fallu repenser intégralement la maquette pour amener nos étudiants non plus à Bac+4 mais à Bac+5. Les principales modifications concernent le niveau Master. Nous avons élaboré une maquette permettant aux étudiants de pouvoir s'orienter vers un autre Master 2, différent de celui de l'IUP SI par un choix d'unités de valeur et nous avons refondu le programme de notre Master 2 (ex DESS) pour y intégrer une forte composante en gestion de projets. L'élaboration de cette maquette a été un gros travail, mené avec les collègues des nombreuses thématiques de l'IUP SI, mais aussi avec les différents partenaires industriels. La mise en place du niveau Licence en septembre 2003 et du niveau Master en septembre 2004 n'aurait pas pu avoir lieu sans un investissement énorme de la part de tous les responsables des années de l'IUP SI.

Un des rôles importants du directeur concerne l'étape de recrutement. Il faut organiser et planifier les jurys de recrutement tant au niveau étude de dossiers qu'audition, et coordonner ce recrutement avec les autres IUP du département d'Ingénierie de l'UPS. De plus, je participe chaque année à plusieurs journées de communication auprès des IUT pour faire connaître nos formations.

Un autre rôle consiste à animer l'équipe pédagogique sur l'ensemble de la formation qui concerne plus d'une centaine d'intervenants (enseignants-chercheurs, chercheurs, industriels) avec des enseignements communs à plusieurs formations. Ensuite, il est nécessaire de gérer les interactions avec les structures de l'UPS (UFR PCA, UFR MIG, département d'EEA, département d'Informatique, IUT A) impliquées dans notre formation.

Dès ma nomination en tant que directeur, j'ai adhéré à l'Association des Directeurs d'IUP, ADIUP. Cette association, qui regroupe la majorité des IUP au plan national, permet de réfléchir à l'évolution du dispositif IUP mais aussi de proposer une coordination nationale des formations. Deux réunions annuelles sont programmées. Dès 1999, j'ai eu l'occasion de participer à ces réunions et ainsi de faire connaissance avec plusieurs directeurs d'IUP du domaine Génie Mathématique et Informatique, GMI.

L'UPS a démarré en 2008 un projet de création d'une école d'ingénieurs universitaire. J'ai été chargé de monter et de porter le dossier d'une des cinq spécialités de cette future école, la spécialité *Systèmes et Micros-systèmes Intelligents*.

Diplôme de Recherche Technologique ISIA

J'ai été, de septembre 2000 à septembre 2007, co-directeur avec P. Bazex du Diplôme de Recherche Technologique ISIA, *Ingénierie des Systèmes Informatiques et Automatiques*.

Le DRT sanctionne une période d'initiation à la recherche et des travaux de recherche technologique sur une période de deux ans visant la résolution d'un problème en relation étroite avec une entreprise. Ces travaux sont effectués dans un laboratoire ou un service de recherche ou d'études, public ou privé, sous la responsabilité conjointe de deux directeurs de travaux, l'un enseignant-chercheur ou chercheur habilité à diriger des recherches, l'autre exerçant une activité principale de recherche ou d'études dans le laboratoire ou le service concerné. Une convention de partenariat sur le projet de recherche est établie entre l'université et l'entreprise.

Les thèmes des sujets de recherche doivent se situer dans l'un des grands champs scientifiques définis par la spécialité du DRT *Ingénierie des Systèmes Informatiques et Automatiques* ou à l'intersection de plusieurs de ces secteurs pour des actions pluridisciplinaires (informatique, automatique-informatique industrielle, traitement du signal, intelligence artificielle, interaction Homme-Machine (image, oral, écrit), robotique, ...

Les étudiants qui s'inscrivent en DRT ISIA sont donc en contrat (CDD ou CDI) avec un industriel dans le but d'effectuer le transfert technologique du laboratoire de recherche associé au contrat. En fonction du sujet de recherche appliqué, les étudiants choisissent une option de parcours de formation, soit *Systèmes Intelligents*, soit *Ingénierie des Systèmes Informatiques* associés tous deux aux DESS SI et ISI. Nous avons convenu avec Pierre Bazex, directeur de l'IUP ISI, de nous occuper chacun de manière indépendante des étudiants inscrits dans nos formations.

C'est ainsi que j'ai eu à gérer cinq étudiants (trois en 2002, un en 2003 et un en 2004). Ce type de formation par la recherche technologique nécessite un encadrement prolongé car il est de la responsabilité de l'enseignant d'amener l'étudiant jusqu'à sa soutenance de diplôme. Pour cela, des soutenances intermédiaires en présence de son employeur et de son responsable universitaire ont été mises en place dès 2001.

Afin de bien synchroniser les étudiants des deux options, une collaboration fructueuse s'est créée entre l'IUP SI et l'IUP ISI.

En DESS et master 2 I.R.R de l'UPS

Au moment de ma nomination en 1991, Michel Courdesses m'a demandé de prendre la responsabilité des Bureaux d'Etudes de robotique ; ils représentaient une composante importante de la formation au vu de son volume horaire (cent heures). L'objectif était d'assurer l'évolution de ces BE ainsi que la coordination entre les différents cours s'y référant. Cette tâche impliquait aussi bien l'organisation temporelle des ressources matérielles nécessaires à l'AIP, tâche non négligeable étant donné le nombre de formations accédant en même temps aux ressources de l'AIP, mais aussi l'étude, le choix et le test des nouveaux matériels industriels. C'est ainsi que j'ai régulièrement participé aux rencontres avec les sociétés vendant du matériel dans les domaines de la robotique et de la vision.

J'ai naturellement participé aux jurys de recrutement. Le public du DESS IRR venant d'horizons fort différents, la phase de recrutement est cruciale car l'effet d'émulation n'est pas facile à

obtenir en une période si courte et il faut préserver l'hétérogénéité des admis.

Au département d'Ingénierie de l'UPS

Réseau de DESS IUP STIC

Devant le nombre croissant d'étudiants décidant de poursuivre en DESS à finalité informatique dans différents IUP, nous avons décidé de créer un réseau national de DESS IUP du domaine GMI dans la thématique des STIC. Nous avons mis rapidement en place des réunions nationales fréquentes de directeurs d'IUP du domaine GMI pour que ce réseau puisse démarrer dès la rentrée 2001. Il visait à favoriser la mobilité des étudiants entre IUP, mais aussi à tenir compte de leurs trois années de professionnalisation lors du DESS IUP, ce qui n'était pas le cas dans les DESS classiques qui accueillent un public n'ayant pas le même type de formation initiale. Ce réseau a fonctionné jusqu'en 2003, date à laquelle la réforme LMD a nécessité de profonds changements.

Recrutement du département Inter-UFR d'Ingénierie

En 2002, le département Inter-UFR d'Ingénierie des IUP de l'UPS a décidé de simplifier la procédure de recrutement en mettant en ligne un service *web* permettant aux étudiants de candidater. J'ai été chargé par le département Inter-UFR d'Ingénierie de coordonner et de mettre en place cette procédure en tant que superviseur, les aspects techniques et logiciels étant confiés au Centre Informatique et de Gestion de l'UPS.

J'ai donc organisé les réunions, le suivi et l'avancement du dossier. Ce service web a été mis en place en mars 2003 et j'en suis depuis cette date le responsable au niveau du département.

Groupement d'Innovation pour l'Industrie

Le Groupement d'Innovation pour l'Industrie, GIPI, réunit des dirigeants de P.M.I., des chercheurs et des responsables de l'activité économique régionale autour de quelques préoccupations communes. L'association propose des activités génératrices d'informations, conférences et études en groupes de travaux. Enfin, elle organise des activités commerciales ou d'intérêt économique plus générales. Les adhérents de l'association sont essentiellement des responsables d'entreprises, petites et moyennes. Chacun y apporte son savoir-faire dans les hautes technologies, électronique, informatique, mécanique, mais aussi dans les domaines plus traditionnels.

En raison de mes relations avec le GIPI dans le cadre de l'IUP SI, son président, Gérard Libéros, m'a demandé de devenir membre du comité d'animation du GIPI en 2003 et j'y suis aujourd'hui le représentant du département Inter-UFR d'Ingénierie. Ce comité, constitué d'une douzaine de personnes dont une large proportion d'industriels (quatre universitaires ou école d'ingénieurs), réfléchit et propose les actions organisées par le GIPI : visites d'entreprises, soirées à thèmes... Ces manifestations rassemblent de trente à quatre-vingt industriels dans un but d'information et d'échange et permettent d'avoir une vision plus précise du contexte industriel.

Commission de spécialistes et comité de sélection

J'ai participé et participe encore aux commissions suivantes :

- Membre titulaire de la commission de spécialistes 63^{ème} section de l'Université Paul Sabatier (1995-1998).
- Membre titulaire de la commission de spécialistes 61^{ème} section de l'Université Paul Sabatier (1995-2001 et 2004).
- Membre suppléant de la commission de spécialistes 61^{ème} section de l'Université Paul Sabatier (2001-2003).
- Membre titulaire de la commission de spécialistes de l'ENIT (2001-2004).
- Membre titulaire (1995-1998 et 2004-2007) et suppléant (2001-2004) de la commission de spécialistes 61^{ème} section de l'université de Perpignan.
- Membre élu du collège scientifique EEA de l'UPS (2009-2012).
- Membre nommé du comité de sélection de l'ENIT (2009-2012).

Intérêt général

- Président du Groupe d'Avancement et de Promotion des sections 60-61-62-63 à l'UPS (2009-2012).
- Membre nommé du Conseil du pôle RIA du LAAS -CNRS (2007-2011).
- Membre du Conseil scientifique du groupe Gepetto du LAAS -CNRS.
- Membre nommé de la Commission Enseignement-Recherche du laboratoire (2006-2009).

II

Activités de recherche

Partie 1

Introduction

Mon activité de recherche a débuté au sein du groupe *Robotique et Intelligence Artificielle* (RIA) du Laboratoire d'Automatique et d'Analyse des Systèmes (LAAS) du CNRS. J'y ai préparé mon doctorat [1]¹ de l'université Paul Sabatier sous la direction de Jean-Paul Laumond, directeur de recherche au CNRS, sur le thème de la planification de trajectoires.

La question centrale de ce groupe de recherche était la possibilité de rendre un système robotique autonome dans le but d'accomplir des tâches de haut niveau.

Afin d'atteindre cet objectif, notre équipe de recherche était organisée autour de deux projets fédérateurs principalement dédiés à l'élaboration et l'intégration d'algorithmes pour la navigation autonome des robots :

- Le projet de *Robotique en environnement humain*, dont l'objet était l'étude du problème de navigation en milieu structuré et dont les supports expérimentaux étaient les plateformes mobiles Hilare et Diligent.
- Le projet de *Robotique d'exploration* dédié à l'étude du problème de navigation en milieu extérieur dont les supports expérimentaux étaient principalement les robots tout-terrains (Lama, Eve et Dala).

Quel que soit l'environnement considéré, le robot doit posséder certaines capacités pour acquérir cette autonomie. Nous pouvons les classer en trois grandes catégories :

- Perception : le robot doit percevoir son environnement afin d'en établir une ou plusieurs modélisations. Ces modèles doivent être adaptés à la tâche robotique afin de limiter la quantité d'informations à traiter. Pour pouvoir construire ces modèles, les robots sont équipés de plusieurs capteurs proprioceptifs (odomètre, centrale inertielle, ...) et extéroceptifs (caméra, télémètre laser, ...). Cette hétérogénéité des données doit être prise en compte afin d'en établir les modèles qui peuvent être de type métrique ou topologique.
- Planification : afin de réaliser une tâche, il est nécessaire de définir les actions à exécuter ainsi que leur enchaînement (prendre une image, se déplacer, ...). Cette planification, souvent associée à une supervision, est très dépendante du contexte qui est en évolution permanente. Il est donc difficile de définir des tâches robotiques à un haut niveau surtout lorsque l'humain

¹Les références sous forme numérique correspondent aux publications du chapitre 5.

est dans la boucle de décision à travers l'interaction Homme-Robot.

- Exécution : une fois les actions planifiées et ordonnancées, il faut les exécuter. Leur exécution ne peut se faire qu'après vérification de pré-conditions. Elle dépend très souvent d'un contrôle qui s'appuie sur des données sensorielles acquises par la perception.

Parmi toutes les fonctions possibles permettant de définir l'autonomie du robot, la gestion des déplacements est un des points critiques. Il faut donc pouvoir planifier le mouvement du robot. Cette planification est plus ou moins complexe en fonction du système robotique et du contexte, et nous verrons qu'elle peut aussi être fonction des données sensorielles ou de l'interaction avec l'homme. Le problème peut se résumer à calculer une trajectoire entre un placement initial du robot et un placement final correspondant à une description simplifiée de la tâche.

Avant de présenter nos contributions sur la planification de trajectoires au travers de diverses problématiques, nous allons définir de manière plus précise le problème que nous cherchons à résoudre.

1.1 Le problème de la planification de trajectoires

La question de savoir s'il est possible de déplacer un objet dans un environnement, et comment le faire, s'est formalisée avec les travaux de [Lozano-Pérez 83] suite à la définition de l'espace des configurations. Cette notion a donné lieu par la suite à de multiples travaux.

Il a fallu attendre les années 90 pour que le premier ouvrage définissant la problématique du *Motion Planning* apparaisse [Latombe 91]. Depuis, les ouvrages de [Choset 05, LaValle 06] sont venus enrichir la littérature et permettent d'appréhender cette problématique plus facilement. Je conseillerais au lecteur voulant approfondir cette thématique de se plonger dans ces ouvrages bien écrits et complets. En conséquence, je ne donnerai pas ici les définitions formelles liées au *Motion Planning* mais seulement les idées et les définitions utilisées par la suite.

Dans un premier temps, nous pouvons nous poser cette question légitime : ce problème est-il réellement un problème difficile ?

Par exemple, si nous voulons planifier une trajectoire dans un plan pour un point parmi des objets, la méthode *BUG* [Lumelsky 87] (aller en ligne droite jusqu'à) apparaît tellement simple qu'il est difficile de percevoir la complexité du problème générique. Il suffit pourtant de considérer le mouvement d'un polyèdre quelconque dans l'espace nous entourant pour constater que le problème devient beaucoup plus ardu.

De manière plus générale, nous voulons considérer des systèmes robotiques plus complexes qu'un simple polyèdre et calculer une trajectoire solution non pas pour un seul corps mais pour un ensemble de corps, articulés entre eux, soumis à de multiples contraintes (butées articulaires, stabilité, fermeture de chaînes cinématiques,...) ce qui complique encore plus le problème.

Il est donc nécessaire de définir des concepts génériques permettant de traiter avec le même formalisme tous les systèmes robotiques.

Que s'est-il passé entre les deux exemples précédents ?

Premièrement, en passant du mouvement d'un objet, de type point, sur un plan en dimension deux (2D) à un polyèdre en dimension trois (3D) nous passons d'un objet ayant deux degrés de liberté (d.d.l.) à un objet ayant six d.d.l. Nous avons donc augmenté la dimension de l'espace de recherche de trajectoire solution qui passe d'un espace de dimension 2 (x, y) à un espace de dimension 6 (trois paramètres de position et trois d'orientation).

Deuxièmement, la différence provient aussi de la structure géométrique que l'on veut déplacer dans un espace. Nous constatons que la forme géométrique du polyèdre va avoir une grande importance sur la difficulté à trouver une solution (à la base du principe de certains "casse-tête").

Si nous transformons le déplacement du polyèdre dans un espace 3D en un déplacement d'un point dans un certain espace, les deux problèmes sont similaires. Il suffit que ce point corresponde à un placement (position et orientation) possible du polyèdre dans l'espace de travail de départ, noté W . La seule différence sera la dimension de l'espace de recherche.

C'est l'idée fondamentale de la notion d'espace des configurations, noté CS : nous allons représenter le système robotique par une configuration $\mathbf{q}$ qui correspond à un point dans cet espace.

Le problème de planification de trajectoires sans collision pour le robot se ramène à calculer une trajectoire τ libre de collision entre une configuration initiale $\mathbf{q}_{init}$ et une configuration finale $\mathbf{q}_{fin}$. Il faudra donc explorer un sous-espace de CS pour déterminer si les deux configurations appartiennent à une même composante connexe de ce sous-espace. Suivant les problématiques, ce sous-espace de recherche peut se ramener à l'espace des configurations libres de collision C_{Libre} ou bien à l'espace des configurations admissibles (libre ou contact) $C_{Admissible}$.

Une structuration exacte de l'espace de recherche permet d'envisager des méthodes complètes (garantie de réponse) mais ces méthodes sont limitées à des espaces de dimensions réduites puisque la complexité algorithmique du problème [Canny 88] est exponentielle en la dimension de CS. Afin de contourner le problème de complexité de construction effective de la frontière de C_{Libre} et suite au développement technologique, les techniques de *Motion Planning* ont abandonné l'idée de complétude exacte pour développer des approches complètes en résolution (discrétisation de l'espace) ou bien complètes en probabilité (échantillonnage aléatoire de l'espace).

Le développement des approches probabilistes, complètes en probabilité, de type *PRM* (*Probabilistic Roadmap Method*) [Kavraki 96] ou *RRT* (*Rapidly Random Tree*) [LaValle 00] permet maintenant d'envisager l'exploration d'espaces de grande dimension. L'idée est de construire un graphe, appelé *roadmap*, en explorant aléatoirement CS sans connaître, ni construire, la frontière de C_{Libre} . Les tests de collision s'effectuant dans l'espace W , il est inutile de connaître explicitement CS. Nous verrons par la suite certains développements autour de ces approches.

1.2 Objectifs de mes travaux

Au départ, mes travaux s'inscrivaient dans les projets du groupe RIA (*Robotique en environnement humain et Robotique d'exploration*) autour des différents aspects de la planification de

trajectoires pour un robot mobile dont la question clé est : *comment calculer une trajectoire géométrique sans collision entre deux situations pour effectuer une tâche spécifique ?* Depuis 2007 et mon changement d'équipe de recherche², ils se sont élargis à des problèmes hors du champ de la robotique mobile.

Autour du fil conducteur de la planification de trajectoires, mes travaux tournent autour des questions suivantes :

- Comment prendre en compte la contrainte de roulement sans glissement dans les algorithmes de planification pour des robots mobiles à roues en environnement d'intérieur ?
- Comment garantir la stabilité des trajectoires et la convergence des algorithmes pour des robots non-holonomes se déplaçant sur des terrains accidentés ?
- Comment s'affranchir de la notion de trajectoire solution globale pour avoir des algorithmes robustes aux imprécisions du modèle et pouvant être utilisés dans une approche commande référencée capteurs ?
- Comment garantir le balayage de surface tout en minimisant le recouvrement en prenant en compte les caractéristiques du type de travail pour des engins agricoles ?
- Comment faire coopérer un utilisateur et un algorithme de planification automatique lors de la recherche de solution ?
- Comment faire réaliser à un robot humanoïde des trajectoires réalistes ?

1.2.1 Planification de trajectoires pour les robots à roues

Les travaux de mon doctorat concernaient la problématique de la planification de trajectoires sans collision pour un robot mobile non-holonyme en environnement d'intérieur de type bâtiment. Une propriété topologique locale du système nous a permis de prouver qu'il existe une solution au système non-holonyme s'il existe une solution dans l'espace libre pour le système holonyme équivalent. Nous avons alors pu développer le premier planificateur itératif non-holonyme exact (algorithme complet) en nous appuyant sur des techniques issues de la géométrie algorithmique en considérant un robot et un environnement polygonal ainsi que sur des chemins locaux respectant la contrainte non-holonomes et optimaux en distance. A cette époque, la problématique du *Motion Planning* était souvent abordée sur des problèmes de dimension réduite pour lesquels les méthodes visant à construire des algorithmes complets étaient privilégiées. Dans cette optique, j'ai ensuite participé à une extension de mes travaux (thèse de Philippe Juhel³) pour le calcul exact des placements admissibles d'un polyèdre évoluant dans un plan. Il est intéressant de noter que ces travaux de *Motion Planning* pour des robots non-holonomes ont donné lieu à de nombreuses extensions en robotique, mais aussi auprès de la communauté de l'automatique.

Si notre approche peut se généraliser à tout système non-holonyme, elle n'a été mise en oeuvre que dans le cas d'un environnement d'intérieur polygonal pour lequel la notion d'obstacle est

²RIA, devenu pôle du LAAS-CNRS est composé maintenant de trois groupes de recherche.

³Cet encadrement n'est pas comptabilisé dans le dossier.

simple à définir. Si nous voulons placer le robot en extérieur et envisager une problématique d'exploration, il faut s'affranchir de la notion d'algorithme complet pour envisager des approches plus réalistes. Pour un robot se déplaçant dans un environnement naturel en extérieur et fortement accidenté, la difficulté de la planification de trajectoires est liée à la notion d'obstacle ; celle-ci est fonction des contraintes liées au terrain et au robot qui ne peut plus être approximé par un simple polygone.

L'objectif de notre travail autour de la thèse d'Alain Haït (1998) a été de développer une algorithmique pour calculer, si elle existe, une trajectoire pour un robot articulé en terrain accidenté garantissant des *contraintes de validité* (stabilité, limites mécaniques, non collision). Les contraintes d'efficacité algorithmique nous ont amenés à utiliser une approche discrète (complète en résolution). De plus, des algorithmes géométriques spécifiques efficaces ont été développés pour prendre en compte les particularités du type de châssis. Afin de pouvoir envisager une application de ces techniques, il est apparu nécessaire d'améliorer la robustesse des trajectoires, en considérant des incertitudes sur le Modèle Numérique de Terrain ainsi que sur la localisation du robot, directement au niveau de l'algorithmique lors de la recherche de la trajectoire solution.

Ce travail nous a permis de mettre en évidence l'importance du lien entre planification, localisation et exécution. Bien souvent, une trajectoire planifiée à partir d'un modèle d'environnement n'est pas exécutable, principalement à cause des erreurs de modèle et des erreurs de localisation lors de l'exécution. Si nous considérons que notre système sensoriel peut percevoir des amers attachés à l'environnement, nous pouvons utiliser l'erreur entre la perception attendue des amers du modèle et celle réalisée dans le monde réel lors de l'exécution, pour relocaliser le robot dans l'environnement. L'étape suivante a été de définir automatiquement les amers pertinents à sélectionner le long de la trajectoire et d'étudier les conditions de leur enchaînement. On transforme ainsi une planification de trajectoire géométrique en une suite de couples amer-capteur à partir desquels les tâches référencées capteurs successives peuvent être définies. Pour valider notre approche et montrer sa généralité, nous avons intégré nos travaux sur un robot mobile. Cette approche de l'enchaînement de primitives amers-capteurs était un des thèmes du projet *Egocentre* et constituait le cœur de la thèse de A. Malti (2005).

Dans les travaux précédents, la tâche robotique était définie par le calcul d'une trajectoire sans collision entre deux configurations sous certaines contraintes. Mais il existe des problèmes pour lesquels c'est la tâche robotique qui amène à définir implicitement une trajectoire. Par exemple si nous voulons tondre une pelouse avec un robot-tondeuse, c'est le problème du recouvrement de surface qui va induire la trajectoire en fonction de différents paramètres (taille de la lame, manoeuvre de l'outil, ...).

Une collaboration avec Renault Agriculture sur la problématique de commande automatique d'un tracteur à partir de données GPS nous a amenés à aborder de manière différente le thème du recouvrement de surface largement développé dans la communauté robotique. Pour pouvoir réaliser la tâche robotique de recouvrement, il est nécessaire de calculer une trajectoire qui tient compte de la contrainte de travail liée à l'outil, et des contraintes liées aux manoeuvres du tracteur, ce qui implique la création de zones fourrières pour lesquelles il faut aussi planifier

une trajectoire de recouvrement. Ce problème de création et de gestion des zones fourrières n'était pas pris en compte à l'époque de ces travaux. Plus précisément, nous avons développé un planificateur géométrique permettant de réaliser la couverture complète d'un champ polygonal en minimisant les recouvrements entre les passages successifs et gérant de manière automatique les manoeuvres en fonction du type de travail et d'une direction principale de travail. Ici aussi, le robot était de type robot mobile non-holonyme et le type de manoeuvres considérées pour respecter la contrainte non-holonyme tenait compte du travail effectué. Nos derniers travaux présentent une extension permettant de réaliser la couverture complète d'un champ polygonal quelconque ou avec obstacles qui implique la création de zones fourrières autour des obstacles ou entre régions ayant des directions de travail différentes.

1.2.2 De la planification au mouvement humain

En 2006, j'ai participé à la réorganisation de la robotique au sein du laboratoire. L'année suivante, mes réflexions m'ont amené à intégrer le nouveau groupe GEPETTO afin de pouvoir étendre ma problématique scientifique à la compréhension et à la modélisation du mouvement pour des systèmes anthropomorphes et de ne plus me limiter à la robotique mobile.

Dès que nous considérons des systèmes ne se réduisant pas à un objet en mouvement sur un plan, nous sommes confrontés au problème de la dimension de CS. Récemment, les techniques développées en planification de trajectoires, autour des algorithmes de recherche probabiliste, permettent de traiter des problèmes de grande dimension en perdant la propriété de complétude (complet en probabilité). L'intérêt de ces approches est de construire une représentation implicite de C_{Libre} au moyen d'un graphe construit aléatoirement. Comme nous ne connaissons pas explicitement la frontière de C_{Libre} , la difficulté est de construire un graphe qui représente correctement la couverture de C_{Libre} et sa connexité. Si nous pouvons biaiser facilement ces approches pour les faire converger rapidement vers le but que nous connaissons, il est beaucoup plus difficile de les rendre efficaces en présence de passages étroits dans l'espace de recherche. Dans un premier temps, nous avons étudié différentes approches pour en combiner leurs avantages afin de limiter leurs inconvénients. En généralisant les arbres locaux d'une approche *Rapidly-exploring Random Trees (RRT)*, nous diminuons la difficulté des passages étroits mais au détriment du nombre d'arbres. Cette approche ne peut être rendue efficace qu'à partir d'un contrôle de la croissance des noeuds et des connexions entre les arbres. Pour cela, nous utilisons la notion de visibilité développée dans une extension des approches *PRM*, *Probabilistic Roadmap Method*.

Ces approches probabilistes permettent de considérer des applications dans le domaine du *PLM* (*Product Lifecycle Management*) où il est possible, à partir d'une maquette numérique, de valider la conception d'une pièce mécanique avant même une première fabrication. Le groupe GEPETTO possède une solide expérience en planification automatique de trajectoires, mais ici un algorithme automatique ne suffit plus, il faut introduire l'utilisateur dans la boucle de recherche de solution. Nous avons donc développé une nouvelle thématique autour de la planification interactive de trajectoires qui s'est concrétisée par le travail de thèse de David Flavigné (2007-

2010). Les travaux existants utilisent une décomposition en deux étapes successives qui limite l'interaction entre l'utilisateur et le processus. L'approche proposée a pour but d'améliorer le guidage d'un opérateur lors d'une tâche d'assemblage dans un environnement virtuel à l'aide d'un algorithme de planification de chemin automatique dans des opérations de *PLM*. Nous proposons une variante des méthodes *RRT* pour construire un *RRT* Interactif ; ce travail s'est intégré dans le cadre du projet *AMSI*.

Les méthodes de planification probabiliste de trajectoires fournissent des trajectoires sous forme d'une suite de chemins locaux. Ces chemins locaux respectent des contraintes d'optimisation sur le système mais sont très éloignés d'une trajectoire que réaliserait un opérateur humain. Si nous voulons que la trajectoire solution pour l'objet manipulé soit semblable à celle [Canny 88] que l'objet effectuerait s'il était manipulé par un opérateur, il est nécessaire de s'intéresser au problème de la cognition spatiale chez l'homme. J'ai donc participé à une ACI avec une équipe de neurobiologistes du CERCO (2001-2004) impliquée dans l'étude des processus de représentation de l'espace et du mouvement. Cette collaboration s'est poursuivie par le projet *Egocentre* (2002-2005) et s'est concrétisée par le co-encadrement de la thèse de M. Tran Tuan (2006-2009) qui porte sur une application des principes moteurs neurobiologiques pour le contrôle du geste d'atteinte des robots humanoïdes. Ici, le modèle est principalement basé sur les hypothèses que l'énergie des signaux des motoneurones est en continu minimisée le long de la trajectoire et que les efforts dynamiques et statiques sont traités séparément. Nous utilisons ce paradigme pour contrôler le robot humanoïde HRP2 en considérant un modèle dynamique du bras qui modélise le système neuromusculaire. Nous montrons que cette approche permet de générer des mouvements réalistes qui respectent les caractéristiques du mouvement humain et qu'il est ensuite possible de synthétiser les mouvements d'atteinte par une combinaison de primitives motrices.

Nous savons maintenant planifier des trajectoires pour des systèmes complexes mais il est encore difficile de planifier des trajectoires qui respectent les contraintes d'un opérateur humain. Si nos premiers travaux permettent de synthétiser un mouvement d'atteinte, ce n'est qu'au prix d'un temps de calcul important totalement inexploitable dans un algorithme de planification probabiliste. Or, dans le domaine en pleine expansion de la conception ou de l'apprentissage de geste à partir de maquette numérique, un enjeu réside dans l'introduction de mannequins numériques et de leur simulation lors de la manipulation de l'objet. Aussi, nous souhaitons étendre nos travaux à la *Planification de mouvements réalistes* en cherchant à caractériser les critères permettant d'améliorer les méthodes de planification de mouvement existantes pour en améliorer la réactivité et les rendre réalistes par rapport au mouvement humain. Les méthodes de planification et de commande de systèmes anthropomorphes développés dans le groupe Gpetto reposent essentiellement sur la géométrie et la cinématique. Il faut les étendre, de manière efficace, pour tenir compte de la dynamique du système au niveau du calcul des chemins locaux permettant de relier les noeuds de la *roadmap*.

A travers le projet *AMSI*, il est apparu que de nombreux problèmes pratiques d'assemblage (ou désassemblage) ne peuvent pas être résolus en considérant seulement des objets rigides (câbles, insertion par déformation,...). Aussi, il serait intéressant d'étendre les approches probabilistes à

la *Planification d'objets déformables*. Le problème ici n'est pas de développer une planification qui se substitue à la simulation physique de corps déformables car nous savons déjà que ces approches sont coûteuses en temps de calcul, mais de proposer une modélisation permettant une intégration efficace du calcul des déformations dans une algorithmique de type probabiliste.

Les approches probabilistes de type *PRM* ou *RRT* apparaissent à ce jour les seules alternatives à la planification de trajectoires pour des systèmes de dimension élevée malgré les inconvénients inhérents à leur principe. Ne pas utiliser au mieux l'information géométrique de la représentation physique du monde 3D qui nous entoure pour travailler seulement dans CS n'est certainement pas la meilleure solution pour exploiter de manière optimale le principe de ces approches probabilistes et il serait intéressant de mieux en étudier le lien.

Partie 2

Planification de trajectoires en robotique mobile

2.1 Planification de trajectoires pour robots non-holonomes

Mes travaux de thèse [1] ont été consacrés à la détermination d'une nouvelle méthode de planification de trajectoires pour des robots à roues (type voiture ou *differential drive*) se déplaçant sur un sol plan encombré d'obstacles. Sur ce type de robot, l'hypothèse de roulement sans glissement des roues le long de la trajectoire impose la prise en compte d'une contrainte qui est appelée *contrainte non-holonyme*.

Avec un robot non-holonyme, nous n'avons plus la garantie qu'une trajectoire solution sans collision soit réalisable. Cette question posée au LAAS par G. Giralt a rapidement donné lieu à des premiers résultats partiels [Laumond 86, Barraquand 89] et à de nombreux travaux dans les années 90. Rappelons brièvement nos travaux dans ce domaine ; ils ont abouti à la construction de la première méthode de planification exacte et complète pour des robots non-holonomes et ont donné lieu à de multiples extensions.

2.1.1 Robot non-holonyme

En général, les planificateurs géométriques considèrent que le robot peut se mouvoir dans toutes les directions en tout point de sa trajectoire. Pour un robot mobile équipé de roues, qui doivent rouler sans glisser, cette propriété n'est plus valide car la contrainte de roulement sans glissement des roues implique au minimum une contrainte non-holonyme du type :

$$F(\mathbf{q}, \dot{\mathbf{q}}) = 0 \quad (2.1)$$

Si nous nous intéressons seulement à la problématique de l'évitement d'obstacle, la configuration d'un robot mobile peut être décrite par trois paramètres (x, y, θ) , x et y décrivent la position de l'origine du repère attaché au robot et θ définit l'orientation du robot (en général, la direction).

Le modèle cinématique du robot peut alors s'écrire comme un système différentiel comprenant deux paramètres de commande v et ω qui représentent respectivement les valeurs des vitesses linéaire et angulaire :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega \quad (2.2)$$

Dans ce cas, on obtient la contrainte non-holonome suivante :

$$\dot{y} \cos \theta - \dot{x} \sin \theta = 0 \quad (2.3)$$

qui traduit la contrainte de roulement sans glissement.

Cette contrainte non-holonome ne réduit pas la dimension de CS mais impose des contraintes sur les directions instantanées de mouvement. Ainsi la voiture qui roule sans glisser ne peut se déplacer que dans une direction tangente à la direction du véhicule (angle θ).

La question que le roboticien doit résoudre devient alors plus complexe puisqu'il faut que la trajectoire solution soit sans collision et respecte les contraintes non-holonomes. Par conséquent toute trajectoire géométrique solution dans $C_{Admissible}$ ne correspond pas nécessairement à une trajectoire réalisable par le robot.

Nous appelons trajectoire faisable, une trajectoire qui va respecter les contraintes de non-collision et les contraintes de non-holonomie.

Le problème est alors de développer un planificateur complet qui nous permette de répondre à l'existence de solution faisable et de construire cette solution de manière efficace, si elle existe.

2.1.2 Planificateur exact itératif

Nous appelons *méthode locale*, $ML(q_1, q_2)$ pour le système 2.2 une méthode qui calcule une trajectoire entre deux configurations en l'absence d'obstacle en respectant la contrainte cinématique non-holonome associée.

Si nous considérons que notre robot peut se déplacer dans un voisinage proche d'une configuration $\mathbf{q}$ sans trop s'en éloigner, comme illustré sur la figure 2.1, alors nous disons que la méthode locale vérifie la propriété topologique locale, suivante :

$$\forall B(\mathbf{q}, \rho) \subset CS_{libre}, \exists \varepsilon \mid \forall \mathbf{q}^* \in B(\mathbf{q}, \varepsilon) \exists ML(\mathbf{q}, \mathbf{q}^*) \subset B(\mathbf{q}, \rho) \quad (2.4)$$

avec $B(\mathbf{q}, \rho)$ la boule centrée sur $\mathbf{q}$ de rayon ρ dans CS.

Une méthode locale vérifiant la propriété 2.4 permet de prouver qu'il existe une solution au système non-holonome s'il existe une solution dans l'espace libre pour le système équivalent sans la contrainte non-holonome. En effet, il suffit de paver de *boules libres* toute trajectoire solution (voir figure 2.2) dans C_{Libre} pour la transformer en une trajectoire faisable en utilisant une méthode locale qui vérifie la propriété 2.4.

Ce résultat nous a permis de développer un planificateur itératif non-holonome général [4] fonctionnant en trois étapes :

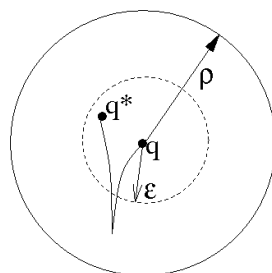


FIG. 2.1 – Méthode locale respectant la propriété topologique

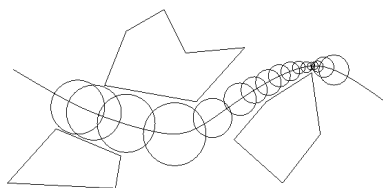


FIG. 2.2 – Pavage d'une trajectoire libre quelconque

- Calcul d'une solution holonome pour le robot en se basant sur les travaux de [Avnaim 89]. L'intérêt de leurs travaux est de pouvoir calculer une solution exacte (si elle existe) dans C_{Libre} pour un polygone quelconque évoluant dans un plan contenant des obstacles polygonaux. Nous répondons ainsi à la question de l'existence et à la construction effective de cette solution en des temps polynomiaux.
- Transformation de cette trajectoire par une concaténation de trajectoires respectant la contrainte non-holonyme. Nous avons utilisé les résultats de [Reeds 90] pour définir la méthode locale à appliquer entre deux configurations voisines. Les courbes de Reeds et Shepp, RS , permettent de calculer la courbe optimale en distance entre deux configurations en espace libre et nous avons montré que ces courbes respectent bien la propriété 2.4.
- Optimisation de la trajectoire solution pour réduire le nombre de manoeuvres qui est la conséquence de la concaténation de courbes de type RS .

Les différentes étapes de l'algorithme sont illustrées dans le cas typique du créneau sur la figure 2.3.

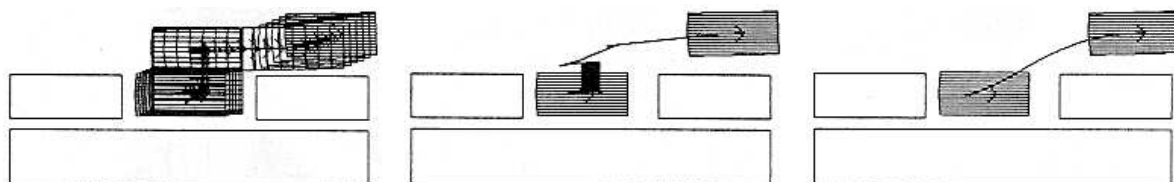


FIG. 2.3 – Exemple des trois étapes de l'algorithme dans le cas d'un parking automatique : trajectoire holonome, non holonome par approximation et après optimisation (de gauche à droite)

2.1.3 Conclusion

Ce planificateur [30] a permis de construire la première solution exacte et complète pour un système non-holonyme. L'intérêt de ce travail est qu'il a pu être généralisé [4] et appliqué à tous les planificateurs géométriques donnant une solution holonome dans C_{Libre} pour un système robotique vérifiant la propriété 2.4. De plus, le travail sur le planificateur holonome nous a permis de développer une extension de l'algorithme initial [Avnaim 89] au calcul exact des placements admissibles d'un polyèdre sur un plan [6]. Nous pouvons ainsi planifier des trajectoires pour un robot mobile non-holonyme en tenant compte de sa hauteur et de la hauteur des obstacles, par exemple, le robot peut ainsi passer sous une table mais pas sous une chaise.

Ce résultat, fort intéressant, n'a abordé la planification de robots mobiles que dans le cadre idéal d'environnements polygonaux plans, ce qui restreint son application à des environnements d'intérieur pour lesquels l'hypothèse de roulement sans glissement est parfaitement justifiée. Or, dans les années 90, de nombreux travaux s'intéressaient au problème de la planification de trajectoires pour des déplacements dans des environnements d'extérieur. C'est tout naturellement que nous nous sommes intéressés à cette problématique en y intégrant la contrainte de type non-holonyme pour respecter la direction de déplacement du véhicule. La section suivante en présente la synthèse.

2.2 Planification en terrain accidenté

Les projets d'exploration, terrestre ou spatiale, se sont développés au cours des années 90 au travers de nombreux projets (VAP/Risp, EUREKA/Iares,...). Pour pouvoir mener à bien sa tâche d'exploration, le robot doit effectuer des tâches de navigation autonome en prenant en compte les contraintes spécifiques à ce type de mission (autonomie, communication limitée, consommation d'énergie,...). Parmi toutes les tâches que la navigation et le superviseur de mission doivent gérer, il y a la problématique de la planification de trajectoires.

Nous retrouvons donc ici une problématique de type *Motion Planning* mais qui doit prendre en compte les caractéristiques liées à l'exploration planétaire.

Les particularités du problème sont ici :

- Le modèle de l'environnement *naturel*, non plan et généralement non structuré. Les modèles les plus utilisés sont de type modèles numériques (MNT), modèles polygonaux (facettes polygonales) ou modèles continus (spline).
- Le modèle mécanique du robot, conçu en fonction de l'environnement. Les robots sont généralement articulés afin d'augmenter leur capacité de franchissement mais cela complexifie leurs modèles car il faut prendre en compte les différentes limitations articulaires.
- L'interaction entre le robot et le terrain qui engendre de nouvelles contraintes. Nous considérons ici principalement les contraintes statiques qui doivent assurer la stabilité et la non collision du robot avec le terrain tout en prenant en compte les limites mécaniques du robot. Une contrainte sur le vecteur vitesse de type non-holonomie permet de tenir compte de la direction de déplacement du robot, même si, ici, il est impossible de parler de roulement sans glissement.

Dans ce contexte, la planification de trajectoires ne peut plus se résumer à calculer une trajectoire sans collision. Il nous faut prendre en compte les contraintes liées à la structure mécanique du robot ainsi que les contraintes de stabilité (au minimum statique) liées à l'interaction sol/robot. Les algorithmes doivent exploiter au mieux les capacités de locomotion du robot pour planifier une trajectoire en fonction de ces différentes contraintes. Dans l'idéal, il faudrait utiliser une modélisation fine de l'interaction sol/robot qui nécessiterait l'utilisation d'un modèle dynamique de déplacement.

Il faut noter que, dans le cas d'un environnement naturel fortement accidenté, la notion d'obstacle est plus complexe à définir que dans le paragraphe précédent car une portion de terrain peut constituer ou non un obstacle en fonction de la structure mécanique du robot (ses capacités de franchissement) mais aussi en fonction de la trajectoire effectuée sur cette portion (comment elle est traversée).

Les travaux menés au LAAS avaient pour objectif de résoudre le problème en envisageant l'intégration des algorithmes au sein du système embarqué. Il fallait donc des solutions réalistes en terme de temps de calcul, ce qui explique que seul l'aspect géométrique soit pris en compte ici. D'autres approches se sont développées dans cette même période à l'aide d'une modélisation plus fine de l'interaction sol/robot en intégrant un modèle dynamique du robot [Ben Amar 93, Ben Amar 95, Cherif 94, Cherif 95, Milési-Bellier 93]. Ces approches permettent une simula-

tion physique réaliste du comportement du robot mais avec des temps de traitement beaucoup plus importants.

A la différence des travaux existant à cette période ([Gaw 86, Liegeois 93, Shiller 90]), les travaux de [Siméon 91, Siméon 93] ont proposé des algorithmes efficaces permettant de planifier des trajectoires pour des modèles réalistes de robots rigides munis de suspensions en formalisant le problème dans l'espace des configurations. Les expérimentations dans le cadre du projet EDEN (Expérimentation de Déplacement en Environnement Naturel) ont permis de démontrer l'intérêt d'une telle approche.

2.2.1 Planificateur pour des robots articulés

L'évolution du projet nous a amenés à considérer des structures mécaniques plus complexes constituées de plusieurs segments articulés. Le modèle de robot utilisé dans nos travaux correspond à un robot articulé possédant trois essieux articulés et des suspensions passives (voir figure 2.4) se déplaçant sur un Modèle Numérique de Terrain.

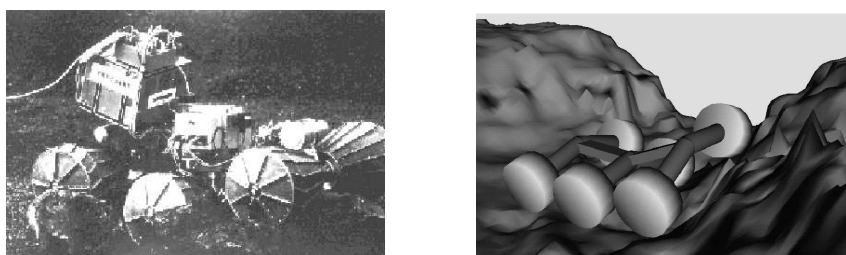


FIG. 2.4 – Le robot LAMA et son modèle de placement

Le placement de ce type de robot consiste à définir six paramètres de situation et n paramètres décrivant l'état du robot (suspensions, angles entre articulations). Si nous considérons que toutes les roues du robot doivent être en contact avec le sol pour définir son équilibre statique, nous pouvons définir un espace des configurations simplifié car les $(n + 6)$ paramètres ne sont plus indépendants. On peut alors définir un nombre réduit de paramètres de configuration. Pour notre robot, une configuration se définit par le triplet (x, y, θ) avec (x, y) position du repère central attaché au robot et θ orientation du robot dans le plan. Pour chaque $\mathbf{q}$ on peut déterminer les six paramètres définissant le placement complet du robot.

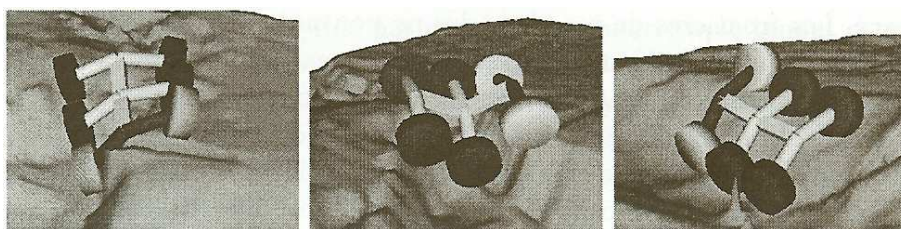


FIG. 2.5 – Contraintes de placement : collision, articulaires et stabilité (de gauche à droite)

Nous pourrions traiter le problème de manière classique en travaillant dans l' EC_{libre} correspondant, mais le calcul des paramètres de placement est très couteux car il nécessite la vérification de toutes les contraintes de placement (voir figure 2.5). Aussi, pour des raisons d'efficacité, nous avons choisi d'utiliser une approche se basant sur une discrétisation de cet espace [Haït 96]. Sur cet espace discrétisé nous pré-calculons le placement d'un essieu en chaque point du MNT, ce qui se ramène à calculer des surfaces associées aux paramètres de placement. Ce pré-calcul permet alors de placer le robot de façon quasi-instantanée en une configuration (x, y, θ) donnée. Le calcul de placement étant résolu, il suffit que les contraintes soient vérifiées pour que la configuration correspondant à ce placement soit valide. L'utilisation d'un modèle hiérarchique de terrain pour la détection de collision permet de valider une configuration de manière efficace. La recherche de trajectoire solution se ramène au problème classique de développer un graphe à partir de la configuration initiale [Barraquand 92]. Pour tenir compte de la difficulté du terrain, une carte de coût est associée à chaque point du MNT et intégrée dans le calcul du potentiel de guidage. Cette carte permet de guider efficacement le développement du graphe suivant un compromis entre distance parcourue et difficulté du terrain. Un exemple du travail de la thèse de Alain Hait est illustré sur la figure 2.6.

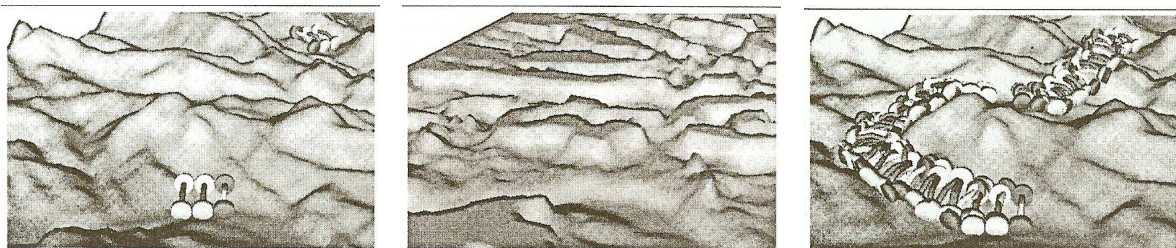


FIG. 2.6 – Exemple de planification sur terrain accidenté avec sa carte de coût associée et la trajectoire solution

2.2.2 Planification robuste de trajectoires

Dans le cadre d'une mission d'exploration, il est évident que les modèles vont être acquis par les capteurs sensoriels à bord du robot afin de construire en ligne un modèle de terrain (MNT dans notre cas) exploitable par le système. Les données capteurs étant entachées d'une erreur, la fonction de perception va construire un modèle qui sera plus ou moins différent de l'environnement réel (incertitudes sur l'élévation de terrain).

De plus, il ne faut pas oublier que nous calculons une trajectoire de référence sur laquelle la boucle de commande va s'asservir. Cette boucle de commande nécessite alors la connaissance de l'état du robot qui contient au minimum l'information de localisation du robot. Or, cette fonction de localisation se base, elle aussi, sur des informations capteurs (odométrie, télémètre laser, image,...) qui seront entachées d'erreurs (incertitudes sur la position du robot).

Dans le contexte du terrain accidenté, nous ne pouvons pas négliger ces deux sources d'erreurs si nous ne voulons pas générer une trajectoire planifiée *non sûre* au niveau de son exécution.

La difficulté de notre travail était de pouvoir prendre en compte ces erreurs pour générer des trajectoires plus *robustes* vis-à-vis des incertitudes tout en limitant la complexité des algorithmes mis en oeuvre. Dans le cadre du co-encadrement de la thèse d'Alain Haït [Haït 98], effectué avec Nicolas Siméon, nous avons résolu ce problème en prenant en compte soit directement les incertitudes au niveau des calculs des placements [3], soit la possibilité d'effectuer des recalages globaux sur des amers connus [25].

2.2.2.1 Prise en compte des incertitudes

◇ Incertitudes sur l'élévation de terrain

Nous avons utilisé un modèle ensembliste pour intégrer ces incertitudes car nous considérons que tous les points du MNT sont acquis avec un intervalle d'incertitude sur l'élévation z . Nous pouvons donc construire deux MNT, MNT_{min} et MNT_{max} , qui vont représenter les erreurs extrêmes sur l'acquisition du modèle. En considérant les placements extrêmes entre ces deux MNT , nous pouvons garantir la validité du placement en une configuration pour un terrain compris entre ces deux MNT .

◇ Incertitudes sur la position du robot

Afin de simplifier le problème, nous avons considéré une incertitude constante autour d'une configuration $\mathbf{q}$, ce qui revient à borner l'incertitude. A partir de ce modèle, une configuration sera sûre si elle reste valide sur tout son voisinage. Etant donné les coûts de calcul des placements pour la structure de notre robot, il est irréaliste de vouloir vérifier cette validité en répétant les tests sur l'ensemble du voisinage. L'idée est de ramener cette incertitude sur une incertitude de placement d'un essieu. On revient ainsi au cas précédent où les placements extrêmes garantissent la validité du couloir de trajectoire autour de la trajectoire planifiée.

Vouloir borner l'erreur de localisation n'est pas réaliste car nous savons qu'elle est fonction de la distance parcourue et des éléments perceptibles dans l'environnement. Aussi, pour pouvoir planifier des trajectoires sur de longues distances, il est nécessaire de prendre en compte ces éléments de l'environnement, les amers, qui permettent une relocalisation du robot comme nous allons le voir dans le paragraphe suivant.

2.2.2.2 Prise en compte des amers

En général, la localisation extéroceptive sur des amers permet de réduire fortement l'incertitude de localisation du robot, mais la perception de ces amers est limitée par les champs de vue des capteurs et par les occultations du terrain.

Une des approches possibles pour accroître la robustesse des trajectoires est donc de prendre en compte les régions de perception des amers dès la phase de planification. L'incertitude va dépendre à la fois du chemin suivi et des éventuels recalage sur les amers.

Nous aurions pu choisir d'explorer l' EC en prenant en compte ces contraintes mais nous avons comme objectif de développer une approche réaliste en temps de calcul et embarquable sur le robot.

Aussi, nous décomposons le problème en deux étapes :

- Nous calculons un chemin (x, y) qui prend en compte l'évolution de l'incertitude, la carte de coût du terrain et la visibilité des amers. Le résultat est un chemin composé d'une alternance de portions de trajectoire où aucun amer n'est visible avec des portions de trajectoire passant par des régions de visibilité de certains amers.
- Ce chemin est transformé en une trajectoire exécutable par le robot (en respectant les contraintes de validité des placements, de non-holonomie et d'incertitude sur la position du robot) en se limitant à une zone autour du chemin solution afin de restreindre les calculs.

2.2.3 Résultats et conclusion

Dans les deux cas d'incertitudes, que ce soit au niveau modèle du terrain ou position du robot, l'incertitude se trouve prise en compte lors de la phase de pré-calcul ce qui n'engendre pas une forte augmentation du temps de calcul de la recherche de trajectoire.

De plus, afin de rendre cette recherche efficace, nous intégrons cette incertitude au niveau de l'heuristique de guidage afin de pénaliser les zones de terrain avec une forte incertitude.

La figure 2.7 montre une recherche de trajectoire dans une vallée étroite. Dans le cas sans incertitude (figures de gauche) le robot est guidé directement à travers la vallée en prenant en compte les inégalités du terrain. Par contre, si on inclut une incertitude sur la position du robot (figures de droite) le problème est plus contraint et le robot se trouve guidé à l'extérieur de la vallée la plus proche. Les figures du haut montrent la modification des cartes de coût associées.

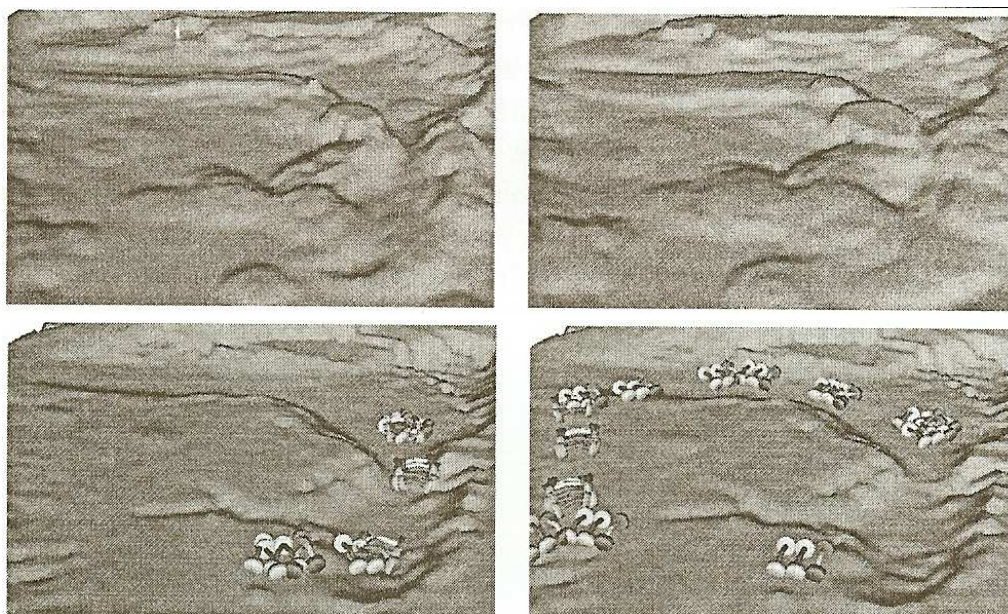


FIG. 2.7 – Cartes de coût et trajectoires solutions sans (gauche) et avec incertitude (droite)

Nous pouvons modifier le calcul des enveloppes supérieure et inférieure du terrain pour y intégrer aussi l'erreur de localisation en position. L'intégration de ces deux incertitudes aug-

mente alors les temps de pré-calcul, mais son principal inconvénient reste la surcontrainte que l'on impose au problème en se mettant dans les cas les plus défavorables.

La planification sur de longues distances avec un recalage est illustrée sur la figure 2.8 en utilisant deux amers.

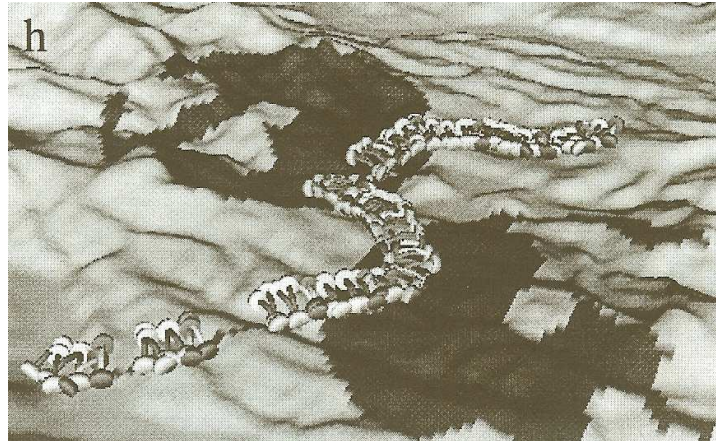


FIG. 2.8 – Trajectoire du robot avec relocalisation sur deux amers

La décomposition du chemin en portions permet d'identifier la(les) portion(s) en échec lorsque le planificateur ne trouve pas de solution. Une amélioration consiste à construire un graphe de régions traduisant l'atteignabilité entre régions de visibilité par propagation depuis les frontières des différentes régions de visibilité [24].

En conclusion, nous pouvons dire que nous avons apporté une réponse à notre problème initial en fonction des capacités technologiques de cette époque. Depuis, le traitement des incertitudes s'est développé dans le cadre de la planification de trajectoires et nous le retrouvons actuellement dans des travaux autour des méthodes probabilistes.

Il nous semble que la prise en compte de la perception au sein des algorithmes de planification est un point important, abordé dans ce travail. C'est cette réflexion qui nous a amenés à développer le travail présenté à la section suivante.

2.3 Planification référencée capteurs

Nous savons que si nous voulons développer des technologies logicielles permettant de doter un robot d'autonomie dans le but de réaliser une tâche précise, il est indispensable de prendre en compte la boucle de perception, décision, action.

Dans la majorité des travaux que nous avons effectués jusqu'ici, nous n'avons considéré que le problème de la planification de trajectoires indépendamment des informations sensorielles et du processus d'exécution de la trajectoire planifiée. Cette approche purement géométrique se base sur un principe de résolution en deux phases :

- une première étape de *planification* calcule une trajectoire solution sans collision entre une configuration initiale et finale. Cette étape fait généralement l'hypothèse que l'environnement est parfaitement connu.
- la deuxième étape de *contrôle d'exécution* consiste à exécuter cette trajectoire en faisant l'hypothèse que la loi de commande est suffisamment précise et robuste pour suivre cette trajectoire sans erreur.

Pour une large classe de problèmes, cette approche a donné de bons résultats et nous pouvons dire qu'aujourd'hui la communauté scientifique en robotique mobile a développé un ensemble de techniques et d'algorithmes pour résoudre efficacement le problème dans le cadre d'un environnement connu. De nombreuses expérimentations ont validé cette approche et permis de mesurer les limites de cette décomposition fonctionnelle. Tout roboticien sait que trois principaux problèmes se posent lors de l'exécution d'une trajectoire planifiée :

1. La géométrie et la position des obstacles dans W ne sont pas connues exactement, ce qui peut engendrer des collisions lors d'un suivi exact de la trajectoire planifiée,
2. Le processus de localisation du robot dans W n'est pas exact, ce qui aura pour conséquence de ne pas suivre la trajectoire solution,
3. L'apparition d'obstacles non modélisés dans W peut entraîner un échec d'exécution.

Pour résoudre ces problèmes, il est possible, par exemple, d'augmenter la taille du robot, d'utiliser des informations sensorielles extéroceptives pour améliorer la localisation du robot et de mettre en oeuvre des techniques locales d'évitement d'obstacle. Nous pouvons noter qu'à l'aide de ces améliorations, des expérimentations de navigation en robotique mobile sur de longues distances [Jensen 02, Thrun 99, Alami 00] sont aujourd'hui envisageables.

Mon travail de thèse a permis de résoudre le problème de calcul d'une trajectoire solution pour pouvoir effectuer un créneau entre deux obstacles de manière automatique. Mais il suffit qu'une des erreurs citées soit présente et le contrôle d'exécution sera en échec.

A contrario, lorsque nous réalisons un créneau avec notre voiture, nous ne calculons pas précisément la trajectoire à suivre mais notre apprentissage nous a permis de sélectionner les amers les plus pertinents pour nous guider pendant la manoeuvre sans connaître exactement le modèle. Nous réalisons une succession de tâches référencées vision.

En asservissement visuel, la tâche est définie localement dans le plan capteur au travers d'amers détectables par la vision, le but est d'utiliser l'erreur de consigne entre amers perçus et amers désirés pour calculer la commande du robot. Ce type de commande est très robuste aux erreurs

puisqu'on travaille directement dans l'espace de mesure du capteur [Samson 91, Espiau 92, Chaumette 06].

Une des solutions pour planifier des trajectoires plus robustes lors de l'exécution est donc de définir les primitives sur lesquelles la loi de commande et le processus de localisation pourront se baser. Nous pouvons alors nous demander comment obtenir automatiquement les amers les plus pertinents pour la localisation et la commande [Deng 95, Sala 06], ou bien essayer de faire de la navigation référencée capteur [Laugier 98, Leven 99, Large 00, Swain-Oropeza 01].

Dans le travail que nous avons effectué, notre objectif était de fournir, pour une trajectoire sans collision, les amers de l'environnement qui serviraient à contrôler l'exécution de la trajectoire par une boucle d'asservissement qui prendrait en compte les erreurs entre les perceptions prévues en phase de planification et les perceptions mesurées en réel en phase d'exécution. Ce travail correspond au travail de thèse de Abed Malti [Malti 05] co-encadré avec Florent Lamiraux.

A partir d'une trajectoire solution, d'un ensemble de capteurs et d'amers visibles par les capteurs, nous attribuons aux différents capteurs l'ensemble des amers visibles le long de la trajectoire. Nous obtenons ainsi des paires d'amer-capteur, notés $\overline{AC}$, le long de la trajectoire pour lesquelles nous définissons une fonction de coût. Cette fonction traduit l'importance de la paire $\overline{AC}$ dans la réalisation de la tâche. Cette approche est générique dans le sens où elle peut s'appliquer à tout type de robot, de capteur et d'amer.

Notre problème n'est pas de localiser le robot sur des amers mais de les utiliser au mieux en vue d'une commande référencée capteur. Ce travail fait le lien entre la planification de mouvement qui garantit une trajectoire de référence sans collision et la commande qui assure l'exécution de ce mouvement. La solution de notre approche doit être vue comme une entrée pour de la commande référencée capteurs permettant l'enchaînement de mouvements référencés capteurs.

Nous avons aussi abordé ce problème dans le cadre du projet ANR RobEA *Egocentre* où nous avons montré que si la définition de tâches référencées capteurs et la structuration différentielle permettent de définir un cadre formel pour l'enchaînement de tâches, il est nécessaire de pouvoir caractériser les types de tâches à enchaîner en fonction des capteurs embarqués par le robot et des amers perceptibles dans l'environnement [31]

2.3.1 Définition d'une planification référencée amers

Comme nous voulons associer à chaque paire $\overline{AC}$ une fonction définissant le rôle de cette primitive lors de l'exécution de la trajectoire, il est nécessaire de définir la fonction de perception. Nous avons caractérisé la fonction perceptuelle d'un amer L par un capteur S par la fonction :

$$\begin{aligned} P_{S,L} : \mathbb{S} \times \mathbb{L} &\rightarrow I_{S,L} \\ (s, l) &\rightarrow P_{S,L}(s, l) \end{aligned}$$

Elle associe à chaque paire $\overline{AC}$ un motif dans l'espace image $I_{S,L}$ en prenant en compte le domaine de visibilité de L par S . Nous pouvons donc associer pour chaque amer une équation de localisation :

$$P_{S_{i_j}, L_j}(s_{i_j}(\mathbf{q}), l_j) = im_j \quad (2.5)$$

où $im_j \in I_{S_{i_j}, L_j}$ est l'image de L_j par S_{i_j} . l_j est supposé connu et im_j est la mesure retournée par le capteur. La seule inconnue dans cette équation est la configuration $\mathbf{q}$ du robot.

Si nous considérons que le robot suit une trajectoire qui appartient toujours à la même classe d'homotopie que la trajectoire solution, que la localisation se fait à haute fréquence en comparaison avec le déplacement du robot, alors la solution de l'équation 2.5 est dans le voisinage de la configuration de référence. Soit $\mathbf{q}_0$ cette configuration de référence autour de laquelle nous linéarisons le système :

$$\frac{\partial P_{S_{i_j}, L_j}}{\partial s}(s_{i_j}(\mathbf{q}_0), l_j) \frac{\partial s_{i_j}}{\partial \mathbf{q}}(\mathbf{q}_0)(\mathbf{q} - \mathbf{q}_0) = im_j - im_{0j} \quad (2.6)$$

Nous exprimons ainsi la relation entre la variation de la configuration $\mathbf{q}_0$ et la variation de la perception de l'amer dans l'espace image. Ce résultat est semblable à celui qu'on trouve en commande référencée vision, $\dot{s} = L.J.\dot{q}$, avec L matrice d'interaction et J matrice Jacobienne du robot. La seule différence est que notre équation 2.6 est définie dans un repère global lié à la planification globale de trajectoire. Puisque nous voulons planifier les couples $\overline{AC}$ les plus pertinents pour contrôler l'exécution de la trajectoire, il est nécessaire de pondérer chaque couple par une fonction d'intérêt. Pour cela, nous pondérons chaque équation 2.6 par un réel positif λ_j qui traduit l'intérêt du couple $\overline{AC}$ pour la tâche robotique. Nous obtenons alors le système d'équations suivant :

$$W(\mathbf{q} - \mathbf{q}_0) = IM - IM_0 \quad (2.7)$$

avec W égal à :

$$\begin{pmatrix} \lambda_1 \frac{\partial P_{S_{i_1}, L_1}}{\partial s}(s_{i_1}(\mathbf{q}_0), l_1) \frac{\partial s_{i_1}}{\partial \mathbf{q}}(\mathbf{q}_0) \\ \vdots \\ \lambda_m \frac{\partial P_{S_{i_m}, L_m}}{\partial s}(s_{i_m}(\mathbf{q}_0), l_m) \frac{\partial s_{i_m}}{\partial \mathbf{q}}(\mathbf{q}_0) \end{pmatrix}$$

et

$$IM = \begin{pmatrix} \lambda_1 im_1 \\ \vdots \\ \lambda_m im_m \end{pmatrix} \quad IM_0 = \begin{pmatrix} \lambda_1 im_{01} \\ \vdots \\ \lambda_m im_{0m} \end{pmatrix}$$

La solution au sens des moindres carrés est :

$$\mathbf{q} = \mathbf{q}_0 + W^+(IM - IM_0) \quad (2.8)$$

avec W^+ la pseudo-inverse de W .

Pour synthétiser notre approche, nous pouvons dire que localiser un robot sur des amers revient à résoudre un système d'équations fonctions de la configuration du robot et des perceptions attendues et planifiées. Si le système est sur-déterminé, la localisation revient à trouver la configuration qui minimise la somme des poids des résidus. Si la position des amers est identique dans la carte modèle et dans le monde réel, le choix des poids n'a aucun effet sur le résultat mais s'il existe une différence, le choix des couples $\overline{AC}$ aura une grande importance sur la réalisation de la tâche.

C'est la raison pour laquelle, dans notre approche, nous utilisons ces poids comme un outil de planification de trajectoires référencées capteurs. Nous en donnons la définition suivante :

Soit un robot mobile avec n capteurs $S_1, \dots, S_n$ et un environnement possédant m amers $L_1, \dots, L_m$. On associe à chaque amer l'ensemble des capteurs qui peuvent les percevoir $S_{i_1}, \dots, S_{i_m}$, une
Planification Référencée Amers, PRA, est définie par :

1. Une trajectoire de référence sans collision :

$$\begin{aligned} \gamma: [0, U] &\rightarrow CS \\ u &\rightarrow \gamma(u) \end{aligned}$$

ou $[0, U]$ est l'intervalle de définition de l'abscisse,,

2. Un ensemble de m fonctions continues réelles à valeurs positives $\lambda_1, \dots, \lambda_m$:

$$\begin{aligned} \lambda_j: [0, U] &\rightarrow \mathbf{R}^+ \\ u &\rightarrow \lambda_j(u) \end{aligned}$$

telle que $\lambda_j(u) = 0$ pour tout u tel que L_j n'est pas détecté par le capteur S_{i_j} quand le robot est dans la configuration $\gamma(u)$.

Le suivi de trajectoire via cette approche revient à estimer la configuration courante du robot au travers de la boucle de contrôle de la tâche en résolvant le système (2.7) autour de la configuration $q_0 = \gamma(u)$ avec les valeurs des poids $\lambda_j(u)$ pour l'abscisse u de la trajectoire.

Dans l'exemple de la figure 2.9, nous souhaitons garer le robot dans l'obstacle O_2 . Si nous utilisons les amers de O_2 dès le début de la trajectoire pour nous localiser nous risquons une collision avec l'obstacle O_1 en raison du décalage de la trajectoire lié à la position réelle de O_2 . Il suffit de s'asservir au début sur O_1 puis progressivement remplacer O_1 par O_2 en jouant sur le poids des primitives $\overline{AC}$ comme illustré sur la figure.

2.3.2 Application

Il nous faut maintenant définir une algorithmique générique permettant de calculer une PRA et voir comment cette approche peut s'intégrer dans une boucle de contrôle générale d'un robot mobile.

Algorithmique associée

Les entrées de l'algorithme sont :

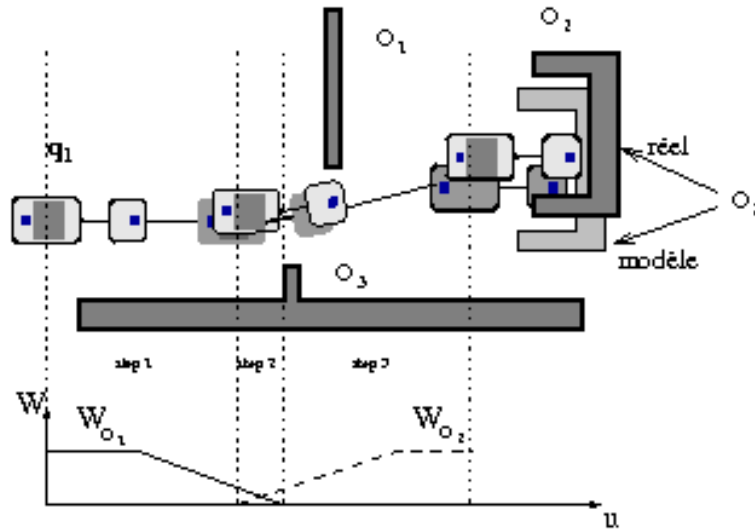


FIG. 2.9 – Exemple de planification référencée amers

1. Un modèle *Map* de l'environnement,
2. Un ensemble de capteurs $\mathcal{S}$ qui équipent le robot R ,
3. Un ensemble d'amers $\mathcal{L}_{env}$ de M détectables par $\mathcal{S}$,
4. Une trajectoire de référence sans collision $\gamma(u)$, $u \in [0, U]$,
5. Le nombre maximum k de couples $\overline{AC}$ à prendre en compte.

La sortie est un mouvement référencé amers, *PRA*, composé de $\gamma(u)$ et d'un ensemble de couple $\overline{AC}$ pondérés.

Pour une configuration le long de la trajectoire, nous calculons le poids de toutes les paires amer-capteur visibles pour en sélectionner les k meilleures. Le poids d'un amer peut dépendre de ses propriétés pour la localisation (visibilité, distance,...), de la tâche (franchissement de porte) ou bien représenter un risque de collision avec un objet. Ensuite, il faut calculer l'abscisse curviligne le long du chemin qui nous permet de garder le même ensemble d' $\overline{AC}$. Pour cela plusieurs solutions sont possibles. Par exemple, on peut utiliser les minimum et maximum des différentes fonctions pour en faire une approximation linéaire. Lorsque la fonction atteint un maximum, cela traduit une diminution de l'intérêt de l'amer pour la suite de la trajectoire. A l'inverse, si on atteint un minimum, soit l'amer n'est plus visible soit son intérêt va augmenter. Dans [21], nous proposons un ensemble de fonctions de pondération pour un capteur laser et des amers de type point ou segment et nous détaillons leur construction.

La détermination du nombre k est cruciale pour que la résolution du système 2.7 donne une solution correcte. Il faut que le système soit bien conditionné et il est souvent nécessaire de prendre plus de trois amers dans le plan. Nous avons montré qu'en calculant la valeur de conditionnement de la matrice de localisation pondérée, on peut ajuster automatiquement ce nombre [18].

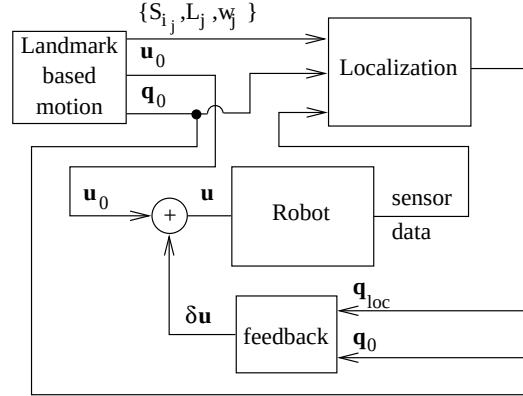


FIG. 2.10 – Schéma de contrôle basé sur la planification référencée amers

Boucle de contrôle référencée sur la planification référencée amers

Le robot utilise une entrée de commande $\mathbf{u}$ et génère en sortie des données sensorielles qui sont les images des amers dans l'espace capteurs. Ces images sont les entrées de la fonction de localisation qui les compare avec les images attendues par la perception du monde idéal du modèle. Elle retourne une estimée de la configuration $\mathbf{q}_{loc}$ du robot comme une perturbation sur la configuration de référence $\mathbf{q}_0$, en utilisant les couples pondérés $\{S_{i_j}, L_j, \lambda_j\}$. Cette estimation de la configuration du robot est l'entrée de référence de la boucle fermée de contrôle haut niveau qui la compare avec la configuration de référence $\mathbf{q}_0$ et retourne en sortie une correction de commande $\delta \mathbf{u}$. Il faut bien noter que la boucle fermée de bas niveau (le bloc noté *feedback* sur la figure 2.10) n'est pas spécifique de notre travail, tout contrôleur en boucle fermé qui stabilise le système peut être utilisé.

2.3.3 Résultats et conclusion

Pour illustrer notre approche, considérons le cas d'une tâche de parking où le robot doit se ranger entre deux murs parallèles. La figure 2.11 montre les amers utilisés pour réaliser la tâche et les deux segments définissant le parking (*seg*₁₁ et *seg*₁₂).

La figure de gauche montre la trajectoire planifiée sans collision à partir du modèle. A partir de cette trajectoire, la *PRA* sélectionne les cinq meilleurs couples $\overline{AC}$ en fonction de leurs pondérations.

Avant l'exécution de cette trajectoire, nous modifions l'environnement par rapport au modèle en décalant vers la droite les deux boîtes définissant le parking (ce qui correspond au déplacement des segments *seg*₈, *seg*₁₀, *seg*₁₁, *seg*₁₂). L'exécution de la trajectoire initiale en boucle ouverte engendrerait inévitablement une collision. Ce problème n'apparaît pas car, durant le mouvement, le robot corrige sa trajectoire en se basant sur les amers de l'environnement réel et la tâche est réalisée avec succès.

Nous avons apporté une première réponse à l'intégration de données perceptuelles au sein d'une

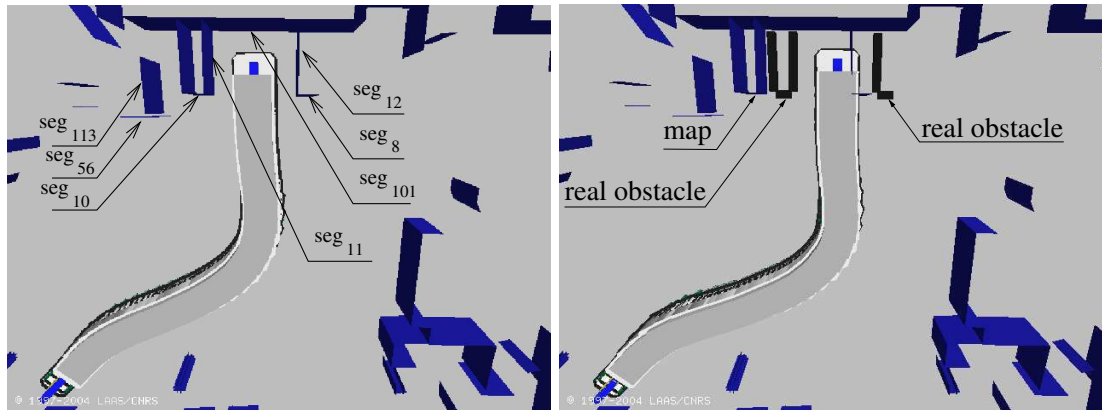


FIG. 2.11 – A gauche, la trajectoire de référence sans collision et les segments sélectionnés par *PRA*. A droite, nous avons modifié l’emplacement des boîtes, le robot adapte sa trajectoire à la nouvelle position du parking.

boucle planification-exécution. Cette approche est générique et donne de bons résultats, comme nous pouvons le voir dans les exemples de trajectoires dans [20, 18].

2.4 Recouvrement de surface

Le développement de la technologie dans les secteurs de la production permet de réduire le temps affecté aux travaux et d'améliorer leur qualité. Cette évolution se retrouve dans le secteur de l'agriculture qui utilise de plus en plus de matériel automatisé. L'intégration et le développement des capteurs, CPU, ... au sein des engins agricoles permet d'envisager maintenant la robotisation de certaines tâches. Un des intérêts de cette robotisation est de décharger l'opérateur de la conduite du véhicule (moissonneuse, tracteur,...) pour se consacrer principalement à l'utilisation du matériel et des produits qui servent à réaliser la tâche (labour, moisson, épandage, traitement,...).

C'est ainsi que Renault Agriculture a développé une commande en boucle fermée pour commander un tracteur en se basant sur des informations GPS. Les expérimentations montrent qu'il est possible de s'asservir sur une trajectoire de référence avec une grande précision au vu de la tâche (± 5 cm). A la différence du chapitre précédent, le problème de localisation et de suivi de trajectoire ne se pose plus et l'environnement peut être modélisé de façon très précise avec des outils de géolocalisation.

Le problème se ramène alors à calculer une trajectoire de référence pour exécuter une tâche agricole en tenant compte de ses spécificités.

Nous pouvons trouver de nombreux travaux en géométrie algorithmique ou bien en robotique qui, à partir de la géométrie d'un champ et d'un critère à optimiser, traitent du problème de recouvrement de surface en considérant généralement une surface plane [Arkin 00, Choset 01, Huang 01, Gabriely 02, Yao 06].

Même si certaines approches sont satisfaisantes d'un point de vue algorithmique, elles ne permettent pas de répondre au problème concret qui se pose, essentiellement pour deux raisons :

- Un travail agricole qui nécessite la couverture complète d'un champ induit obligatoirement la création d'une zone fourrière où le tracteur doit effectuer ses manoeuvres. Cette zone fourrière dépend du type de tracteur, du type de travaux et doit être travaillée, en début ou en fin de trajectoire, pour assurer la couverture complète du champ.
- Une autre contrainte importante est la prise en compte d'une direction privilégiée de travail. Pour diminuer les glissements et les efforts du tracteur, il est nécessaire que le tracteur effectue son travail avec un très faible rayon de giration et la détermination de cette direction est un critère important pour effectuer le recouvrement.

Comme les approches existantes ne prenaient pas en compte ce type de problème, nous avons dû définir une algorithmique de planification qui considère à la fois la structure géométrique du champ et le type d'outils et de travaux à réaliser ainsi que la gestion automatique des zones fourrières.

Nos premiers travaux [23] ont modélisé le problème sous forme d'un graphe en y incluant les deux contraintes précédentes. Le calcul de trajectoire se ramenait alors à un problème de recherche de chemin hamiltonien. Cette solution s'avère satisfaisante dans le cas de champs convexes ou incluant un seul point de concavité et ne contenant pas d'obstacle. Nous avons étendu ces résultats aux champs avec obstacles [22] qui, dans certains cas, génèrent de nou-

velles zones fourrières. Par la suite, nous avons traité le cas plus complexe de recouvrement des champs non convexes, en incluant les zones fourrières, tout en minimisant les recouvrements [19].

Après avoir donné quelques définitions, nous présentons notre démarche scientifique et les différentes contributions.

2.4.1 Formalisation du problème

La figure 2.12 illustre les définitions suivantes :

- Direction de travail D_t : direction privilégiée pour travailler la totalité de la zone de travail.
- Trace : partie du champ travaillée le long d'un segment de trajectoire.
- Zone de travail ZT : zone où l'outil effectue le travail dans des conditions optimales.
- Zone fourrière ZF : zone où le tracteur effectue les manoeuvres pour changer de trace de travail.
- Zone d'entrée E et de sortie S : zones par lesquelles le tracteur peut entrer ou sortir du champ.

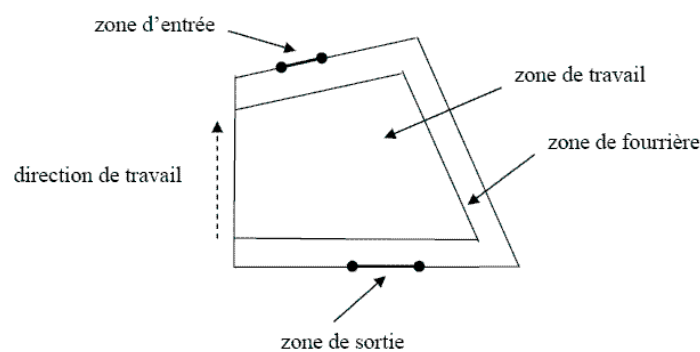


FIG. 2.12 – Définitions pour un champ

L'utilisateur définissant la géométrie du champ et les zones E et S , notre problème peut se définir ainsi :

Calculer une trajectoire partant de E et se terminant en S qui garantit la couverture complète du champ, ZT et ZF , tout en minimisant les recouvrements entre les différentes traces ainsi que le nombre de manoeuvres.

Il est important de réduire le nombre de manoeuvres dans ZF pour minimiser le temps de travail car elles se font à faible vitesse avec parfois des points de rebroussement nécessitant l'arrêt du véhicule.

La détermination de ZF doit se faire de manière automatique, elle dépend de D_t et sa largeur est fonction de plusieurs critères (le type de manoeuvre, la largeur de l'outil, le rayon de giration du tracteur). Pour effectuer le recouvrement complet du champ, la ZF doit être travaillée soit avant (e.g. moisson) soit après (e.g. labour, semis) le recouvrement de la ZT .

2.4.2 Cas des champs convexes

Nous avons commencé par considérer le cas simple d'un champ convexe sans obstacle sur un plan. Nous avons constaté que si chaque direction de travail engendre un nombre de manœuvres différent, la couverture complète du champ sera identique. La longueur de trajectoire pour le recouvrement sera quasiment la même pour les différentes directions. Seules les portions de trajectoire reliant ZT et ZF seront différentes, mais ces portions de trajectoire sont très petites par rapport à la longueur totale de la trajectoire pour effectuer le recouvrement.

En conclusion, le critère prépondérant qui doit être minimisé afin de choisir la direction de travail est le nombre de manœuvres. Nous avons montré dans [19] que, dans un champ convexe, la direction de travail optimale est toujours parallèle à un côté du champ. Il suffit donc de déterminer le côté du champ pour lequel le nombre de manœuvres est minimal pour connaître la direction de travail.

Une fois la direction de travail fixée, connaissant les caractéristiques du travail (place pour les manœuvre, largeur de l'outil, type de travail,...), nous en déduisons un ensemble de points topologiques qui définissent les traces à suivre dans la ZT et dans la ZF (voir l'exemple sur la figure 2.13). Une trace correspond à une portion de trajectoire qui doit être travaillée. Les traces de travail seront donc parallèles, décalées de la largeur de l'outil. Connaissant les traces à suivre pour recouvrir le champ, il suffit ensuite de calculer l'enchaînement à réaliser pour minimiser le recouvrement.

Nous avons résolu le problème en construisant un graphe G pour lequel :

- Les sommets correspondent aux points topologiques,
- Les arêtes correspondent à :
 - Des traces qui doivent être travaillées,
 - Des trajectoires entre traces (fonction du type de manœuvres),
 - Des trajectoires entre E , S et les traces,
 - Des trajectoires pour passer entre ZT et ZF

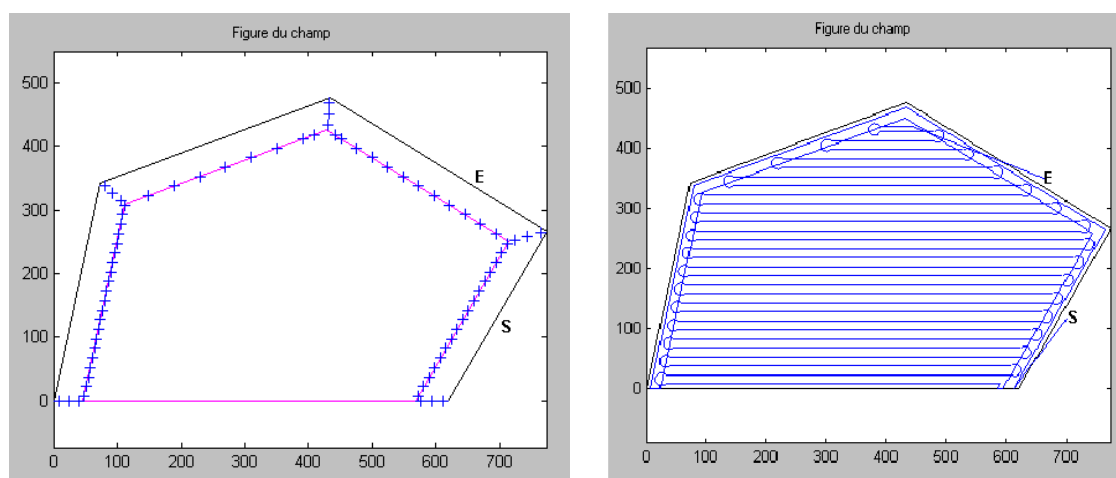


FIG. 2.13 – Points topologiques associés au champ (gauche) et trajectoire de balayage

Un étiquetage adéquat des arêtes permet de modéliser efficacement l'enchaînement entre les arêtes. Il suffit ensuite de chercher un chemin hamiltonien dans G . C'est un problème NP-complet bien connu pour lequel nous n'avons pas de garantie de solution.

En plus des arêtes correspondant aux changements de traces liées aux manoeuvres, il faut prendre en compte des changements spécifiques de traces (saut de traces). L'ajout de ces arêtes spécifiques et l'utilisation d'un algorithme basé sur une optimisation locale à chaque sélection du sommet suivant, nous permet de limiter les situations de blocage lors de la recherche. Ensuite, une heuristique basée sur une minimisation des points de départ initiaux permet de réduire fortement le temps de calcul.

La comparaison de notre méthode avec un calcul exhaustif de toutes les solutions montre que nos résultats sont satisfaisants tant du point de vue de la qualité de la solution (proche de la solution optimale) qu'au niveau du temps de calcul (réduction d'un facteur 10 à 20).

Le calcul de la trajectoire complète incluant les manoeuvres du tracteur est montré sur la figure 2.13 en prenant en compte le travail dans la ZF à la fin (travail de type labour).

2.4.3 Prise en compte des obstacles

A la suite de ce premier résultat, il était nécessaire d'introduire la notion d'obstacles qui peuvent se présenter sous diverses formes (pylone, bosquet, mare,...). Tout comme dans le cas du paragraphe 2.2, il nous faut préciser la notion d'obstacle. En effet, si le robot ne doit jamais entrer en collision, il faut prendre en compte la contrainte du faible rayon de giration lors de la phase de travail. Aussi, l'évitement se fera de manière différente en fonction de la taille de l'obstacle par rapport à la largeur de la trace de l'outil :

- Dans le cas d'un petit obstacle, (quelques largeurs de trace) il suffit de modifier localement la trace correspondant à une collision,
- Dans le cas d'obstacle large, il faut redéfinir automatiquement une nouvelle ZF autour de l'obstacle, qui devra être travaillée à son tour. Sans cette nouvelle ZF , une portion non négligeable de terrain ne serait pas travaillée.

Le problème des petits obstacles peut facilement être résolu en testant les collisions entre les traces initiales et les obstacles.

Mais pour les grands obstacles, nous avons dû développer un algorithme basé sur le principe des algorithmes de *plane sweep* pour définir un enchaînement permettant de travailler les ZF des obstacles après (ou avant) avoir travaillé les zones autour. L'idée est de balayer dans le sens vertical l'ensemble des obstacles pour prendre en compte l'apparition ou la disparition d'un obstacle pour savoir quand il faut créer une zone fourrière et quand la travailler.

Nous pouvons voir sur la figure 2.14 le résultat de notre méthode sans nouvelle ZF (à gauche) et avec nouvelle ZF (à droite).

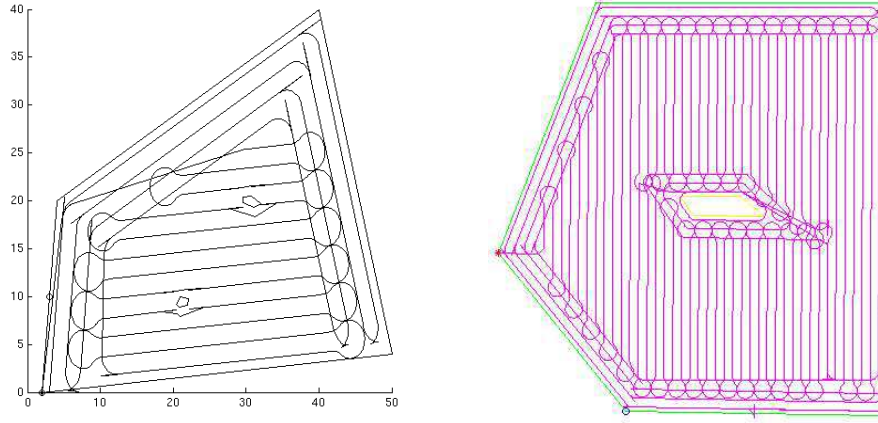


FIG. 2.14 – Recouvrement incluant des évitement d'obstacles

2.4.4 Cas des champs non convexes

Lorsque le champ présente une forte concavité, la trajectoire optimale ne peut pas garder une direction de travail constante. Il est donc nécessaire de décomposer le champ en prenant en compte la meilleure direction de travail dans chaque partie du champ. Nous appellerons *région* chacune des parties polygonales convexes du champ obtenue par décomposition. Nous allons faire une décomposition cellulaire du champ non convexe en régions convexes en prenant en compte la direction de travail dans chaque région.

L'idée est de couper le champ au point de concavité en un nombre discret de coupes en fonction des directions de travail possibles dans la région créée par ce découpage.

Il faut noter que la coupe va engendrer une nouvelle zone fourrière ZF puisque les deux régions auront des directions de travail différentes (voir figure 2.15) et il sera nécessaire de travailler cette nouvelle zone fourrière. Nous avons défini les coupes de manière à traduire les directions de travail possibles pour les régions adjacentes à ce point de concavité (voir détail dans [19]).

Notre méthode fait un découpage de manière récursive tant qu'il existe un point de concavité. Cependant, dans le pire des cas, il est possible que le choix d'une coupe ne supprime pas un point de concavité mais seulement un segment du champ. Il faut donc effectuer un nombre de récursivités égal au nombre de segments, ce qui donne une complexité algorithmique en $Np(k)$ où N est le nombre de sommets du champ, k le nombre de sommets concaves et $p(k)$ une fonction de la forme du champ vérifiant $p(k) < N$.

Nous pouvons ensuite comparer les différents découpages et leur associer un coût qui est la somme des coûts de travail de toutes les régions créées. Même si, en pratique, nous constatons que $p(k) \ll N$, cette approche exhaustive est très vite trop coûteuse dès que N et k deviennent grands. Aussi, nous avons développé une heuristique pour choisir localement la solution opti-

male de découpage à chaque itération. Cette heuristique quantifie l'efficacité d'une coupe en fonction du coût et de la superficie des régions convexes créées.

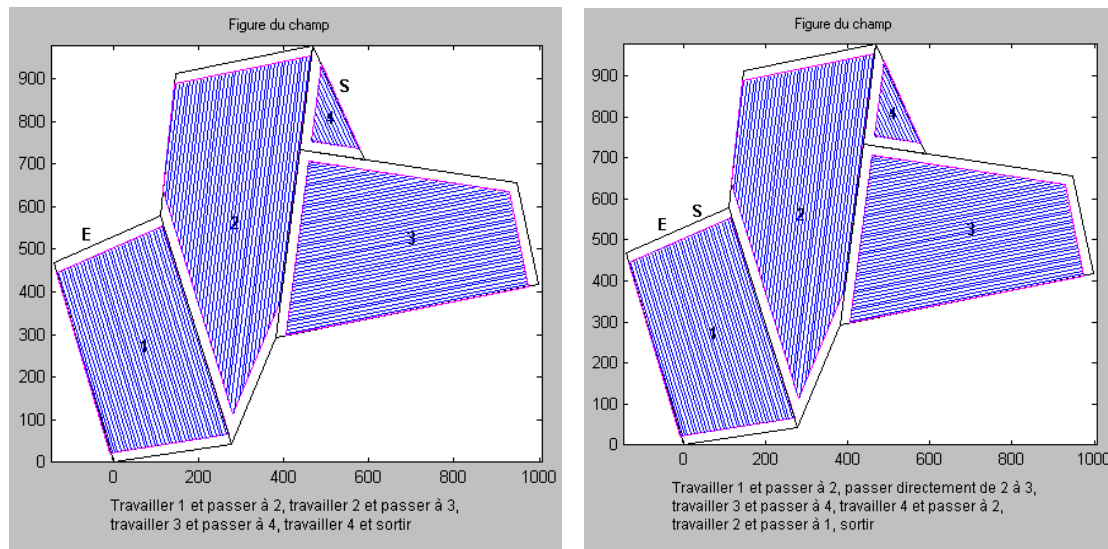


FIG. 2.15 – Exemple de balayage entre des points d'entrée et de sortie différents

L'algorithme retourne un ensemble de régions avec lequel on construit un graphe de régions. Un nœud correspond à une région et un arc correspond à une adjacence entre régions matérialisée par une zone fourrière. Pour avoir la garantie de couverture complète du champ, il est nécessaire d'avoir la garantie de parcourir tous les nœuds du graphe, sachant qu'il y a deux façons de parcourir un nœud :

- Le traverser sans travailler la région correspondante : passage seul,
- Le traverser en travaillant la région correspondante : travail et passage.

Cette distinction est obligatoire car, pour certains travaux, nous ne pouvons pas traverser une région déjà travaillée. Nous avons développé un algorithme *H-LIKE* qui effectue le recouvrement de surface et qui prend en compte les contraintes de travail et passage comme nous pouvons le voir sur la figure 2.15.

Pour valider notre approche, nous avons construit une base de champs et effectué différents tests pour mettre en évidence l'intérêt de nos algorithmes en étudiant plus particulièrement les points suivants :

- Comparaison du découpage exhaustif et de notre heuristique : quantitativement, la solution sous-optimale donnée par notre heuristique est toujours proche de la solution optimale en temps de travail ainsi qu'en nombre de manœuvres. Nos résultats sont proches de l'optimum (moins de 5% de différence) mais avec des temps de calcul fortement diminués (facteur 10 à 20).
- Influence des points d'entrée et de sortie sur la solution de l'algorithme H-Like : pour un même découpage, l'enchaînement entre régions dépend de ces points mais sans poser de problème particulier (voir figure 2.15). Il serait intéressant de pouvoir comparer plusieurs ensembles de *E* et de *S*.

- Influence du nombre de sommets total et de sommets concaves : l'analyse qualitative de la complexité algorithmique est difficile car elle est fonction de nombreux paramètres d'entrée (k, N, E, S , nombre de manœuvres dans les $ZF, \dots$). Néanmoins, on peut remarquer que l'étape la plus coûteuse en temps de calcul est l'étape de découpage. Nous constatons que la recherche de solution par un découpage exhaustif a une complexité exponentielle dans tous les cas alors que notre heuristique permet d'avoir un comportement quasi-linéaire.

2.4.5 Conclusion

Nous avons abordé cette problématique suite à une rencontre avec des industriels confrontés à un problème bien concret lié au recouvrement de surface. Elle était nouvelle pour nous et notre groupe de recherche, aussi nous l'avons abordée en partant du cas simple de champ convexe pour étendre notre approche aux obstacles et à la non-convexité. Nous avons apporté une réponse originale en introduisant la notion de point topologique et de zone fourrière. La notion de direction de travail, essentielle dans ce contexte, se retrouve à la base de nos différentes contributions.

La prise en compte des manoeuvres au niveau des zones fourrières dans l'algorithmique était une nouveauté que nous avons étendue au cas avec obstacle ainsi que dans le cas de champ non convexe. Il est apparu que la notion de convexité pour ce problème n'était plus la notion de convexité classique puisqu'elle dépendait de la direction de travail (le champ de la figure 2.16 est convexe pour la direction de la flèche)

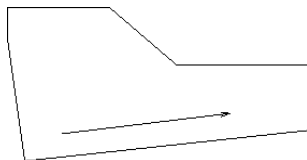


FIG. 2.16 – Convexe ou non convexe ?

2.5 Bilan

Nous avons abordé la problématique de la planification de trajectoires à travers le problème de la contrainte de roulement sans glissement. Notre solution est générique car elle peut s'appliquer à tout planificateur global donnant une solution dans C_{Libre} pour tout système, à la condition qu'il soit possible de définir une méthode locale respectant la propriété topologique locale. Cette approche s'est généralisée à l'ensemble des robots à roues et a donné lieu à de nombreux travaux sur les systèmes non-holonomes [Barraquand 93, Svestka 95, J.-P. Laumond 98].

Il faut noter que la non-holonomie a rapidement intéressé la communauté des automaticiens ; ils ont cherché à calculer des trajectoires pour de tels systèmes en respectant les contraintes dans l'espace des vitesses, mais sans prendre en compte les obstacles. Les travaux de [Fliess 95, Samson 95, Rouchon 94] ont montré qu'il est possible de construire une solution exacte si le

système initial vérifie la propriété de platitude ou de linéarisation par retour d'état. La propriété topologique locale a trouvé son extension avec les notions de commandabilité locale et de commandabilité en temps petit. Le calcul de trajectoires pour des robots respectant la non-holonomie s'est poursuivie, que ce soit en boucle ouverte dans le cadre de la commande optimale [Soueres 94] ou en boucle fermée dans le cadre de synthèse de lois de commande [Luca 98, Souères 98].

Nous avons considéré la planification de trajectoires pour des environnements et des robots polygonaux afin d'utiliser des outils de *computational geometry* car nous cherchions à caractériser des algorithmes exacts et complets. La description analytique exacte au niveau global et local a permis de répondre à cette question mais a aussi montré ses limites.

Calculer la frontière exacte d'un CS de dimension trois, (x, y, θ) , n'est pas trop difficile mais apparaît trop complexe pour se généraliser à des systèmes quelconques évoluant dans l'espace.

Aussi, lorsque nous avons traité le problème de la planification de trajectoires sur un terrain accidenté, nous n'avons plus considéré une algorithmique exacte et complète mais utilisé la notion d'espace discrétisé afin de construire des méthodes complètes en résolution.

Dans ce cadre, notre contribution avait pour but de proposer des solutions efficaces en tenant compte des capacités CPU embarquables dans les années 90. Le développement d'algorithmes géométriques, adaptés aux robots mobiles articulés, nous a permis de gagner en efficacité à la fois sur le calcul du placement et sur la validité des configurations mais également lors du guidage de la recherche de solution. Cette problématique s'est poursuivie depuis au travers de nombreux travaux et projets ([JPL-NASA 06], [Wettergreen 05], [CMU-NASA 06], [Zucker 10], ...).

Bien entendu, la notion d'incertitude est essentielle dans ce contexte. Autant nous pouvons considérer qu'un plan de bâtiment peut être assez précisément connu, autant il est illusoire de travailler sur un modèle exact dans le cadre de l'exploration.

Les deux types d'incertitudes prises en compte, modèle du terrain et localisation du robot, permettent d'accroître la robustesse des trajectoires solutions sans trop affecter l'efficacité du planificateur mais en sur-contrainant le problème. Il est évident que notre dernière approche visant à intégrer des tâches de recalage sur des amers permet d'envisager des déplacements sur de longues distances en y intégrant les informations issues des capteurs pour définir des tâches de déplacement asservies sur les amers ou sur une localisation proprioceptive lorsqu'aucun des amers n'est visible.

Ce travail nous a amenés à considérer le rôle de l'information sensorielle dans le cadre du *Motion Planning*. Si nous pouvons à la fois planifier une trajectoire libre de collision et y associer l'information perceptuelle nécessaire à sa bonne exécution en prenant en compte les capacités sensorielles embarquées sur le robot, nous améliorons alors grandement la robustesse de la solution. Nous retrouvons cette problématique dans d'autres approches de planification [He 08].

Cette réflexion est au centre du travail initié autour de la planification de mouvements référencés capteurs. Nous avons donné une définition générique qui s'appuie sur une trajectoire de référence libre de collision et associe les couples d'amer-capteur pertinents le long de la trajectoire. A chaque couple est associée une fonction de pondération qui traduit sa pertinence pour la

réalisation de la tâche. Nous avons considéré la combinaison d'une fonction prenant en compte la visibilité et le danger de l'amer. Même si les résultats sont satisfaisants, de nombreux problèmes restent ouverts comme la prise en compte de la perte possible d'un amer en cours d'exécution et l'apparition d'obstacles inconnus dans le plan.

Une piste de travail, qui me semble plus prometteuse, serait de ne plus s'appuyer sur une trajectoire géométrique pour construire la séquence de mouvements asservis sur des amers, mais de planifier directement la solution dans un espace qui tient compte de la pertinence des amers. Nous pourrions ainsi calculer toutes les classes de trajectoires solutions (classes d'homotopie) avant d'effectuer la recherche de solution. Les approches de *Motion Planning* probabilistes, qui seront présentées plus loin, permettraient d'envisager une telle recherche et des applications dans le cadre de tâche de navigation sont possibles car elles donnent la possibilité de travailler dans des espaces de grande dimension.

Il faudrait pour cela directement construire un graphe, une *roadmap*, dans un espace issu du produit cartésien de l'espace des configurations du capteur et de l'espace des configurations des amers en maximisant le poids des primitives amer-capteur durant la recherche de solution. Nous pourrions ainsi générer des noeuds dans la *roadmap* qui traduisent leurs utilité en terme de localisation ou d'asservissement.

A l'opposé, nous avons étudié le cas de recouvrement de surface en supposant que tous les problèmes de modélisation, de localisation et de commande étaient résolus pour mieux nous focaliser sur la spécificité des tâches considérées. Au lieu de vouloir résoudre uniquement le problème de recouvrement, nous y avons associé les contraintes spécifiques à une application agricole. Nous aurions pu chercher à résoudre ce problème d'un point de vue purement *computational geometry* mais nous avons préféré essayer de développer des solutions réalistes centrées sur une application possible. Nous avons donc souvent introduit des heuristiques de recherche de solution qui nous ont fait perdre la garantie de solution optimale en faveur d'une réduction drastique des temps de calcul. La qualité de nos solutions sous-optimales, proches de l'optimum, et les temps de calcul réduits montrent la pertinence de notre approche.

Même si nous avons interrompu nos travaux sur la problématique du recouvrement de surface, ce thème est toujours d'actualité dans la communauté comme en témoigne les publications récentes [Oksanen 09, Choi 09, Mannadiar 10, Hu 10]. Cette problématique vise un large champ applicatif car le recouvrement de surface apparaît dans de nombreux cas, applications de nettoyage, applications de déminage ou des opérations de couverture de réseau. C'est donc une thématique qui donnera encore lieu à de nombreuses publications avec une extension intéressante aux cas de surfaces 3D comme envisagé dans les travaux de [Atkar 05].

Nous n'avons présenté dans cette partie qu'un nombre réduit de spécificités de la problématique générale de la planification des robots à roues. Il est évident que si plusieurs de ces thèmes sont encore bien actifs dans la communauté robotique, il en existe bien d'autres (environnement dynamique, multi-robots, ...). Aussi, si le problème initial de la non-holonomie peut être considéré comme résolu, il reste encore à y intégrer l'ensemble de la problématique dans des algorithmes efficaces.

Partie 3

De la planification au mouvement humain

Dans la partie précédente, nous nous sommes plus particulièrement attachés à développer des méthodes de *Motion Planning* adaptées au contexte de la robotique mobile. Si nous nous intéressons au problème plus générique du *Motion Planning*, une des difficultés des approches est liée à la dimension de CS et que nous abordions le problème d'un point de vue déterministe ou probabiliste, il faut souvent en arriver à construire un graphe qui représente le plus précisément possible la couverture et la connexité de C_{Libre} .

Même si les techniques probabilistes permettent de résoudre des problèmes hautement dimensionnés dans des espaces contraints, elles restent des méthodes de calcul *hors-ligne* qui nécessitent des temps de calcul et une place mémoire importants.

Nous avons donc voulu étudier plus en détail les méthodes de planification probabilistes génériques afin d'améliorer les performances et traiter des problèmes plus complexes. Cela nous a amenés dans un premier temps à combiner les avantages de différentes approches afin d'obtenir une méthode plus efficace.

Nous sommes bien conscients des limitations de ces approches qui explorent *aléatoirement* un espace pour en déduire implicitement une représentation de C_{Libre} . Aussi, sur certains exemples, les algorithmes mettent longtemps pour converger vers une solution alors que celle-ci, ou le début de solution, apparaît comme une évidence pour un utilisateur, qui se sert de son expérience, son raisonnement et sa connaissance du monde dans sa recherche de solution.

Suite à cette remarque, il nous a paru intéressant de voir comment nous pouvions insérer l'*intention* d'un utilisateur lors d'une recherche automatique de trajectoire afin de proposer une méthode couplant interaction utilisateur et planification automatique probabiliste.

Comme nous l'avons vu dans le paragraphe 1.1, toute planification probabiliste implique l'utilisation de méthodes locales pour construire le graphe ou l'arbre associé. Dans des espaces hautement dimensionnés, le calcul d'un chemin local peut être très complexe et demander un temps de calcul important. Dans un but d'efficacité, les chemins locaux sont souvent réduits à une simple interpolation linéaire, quitte à transformer le chemin solution dans une deuxième étape.

Si nous faisons exécuter ces chemins locaux à un robot humanoïde, les mouvements ressem-

bleront à des mouvements de robot et non à des mouvements humains car nous ne savons pas encore les caractériser efficacement. Aussi, en parallèle avec nos travaux sur la planification, nous nous sommes intéressés aux principes de contrôle moteur existant en neurosciences pour mieux en comprendre les fondements, dans un objectif, à plus long terme, de pouvoir les intégrer dans des méthodes de planification.

3.1 Planification probabiliste

Un des problèmes majeurs des méthodes probabilistes est la complexité en temps de calcul liée à la dimension de CS et à la complexité de l'espace de travail. Avec les méthodes probabilistes, le temps de calcul dépend moins de la dimension de CS qu'avec les approches déterministes, mais il est fonction du nombre d'objets dans l'espace de travail (lors des tests de collision) et du nombre de passages étroits dans W (faible connexité de certaines parties de C_{Libre}). Beaucoup de travaux cherchent à améliorer le guidage de l'exploration aléatoire de C_{Libre} et présentent des variantes pour améliorer les performances des algorithmes : chercher les passages étroits [Boor 99, Amato 98, Sun 05, Strandberg 04], réduire le nombre de noeuds [Siméon 00] ou le nombre d'itérations [Yershova 05, Dalibard 09a], guider la diffusion dans les sous-variétés intéressantes de C_{Libre} [Burns 07, Burns 05], améliorer l'échantillonnage [Hsu 05, Katz 06, Shawna Thomas 07].

Nous pouvons ainsi trouver des méthodes efficaces pour se déplacer dans un espace très contraint (type tube) ou bien pour explorer les frontières de C_{Libre} . Nous avons constaté que, lorsque le problème comporte plusieurs grands espaces libres séparés par des passages étroits, les algorithmes trouvent difficilement une solution. Nous avons donc décidé de nous focaliser sur ce type de problèmes que nous allons préciser par la suite.

Comme le graphe doit traduire la connexité de C_{Libre} tout en effectuant une exploration aléatoire, un des problèmes principaux va être de pouvoir créer des arêtes dans les *passages étroits* de C_{Libre} pour traduire cette connexité. On appelle passage étroit le sous-ensemble des q de C_{Libre} possédant un petit voisinage libre de collision et qui relie des sous-ensembles de C_{Libre} pour lesquels le voisinage des configurations sans collision est beaucoup plus large. Comme le voisinage des configurations libres est petit dans les passages étroits, la probabilité de tirer un échantillon dans cette zone est faible, comme on peut le constater sur la figure 3.1.

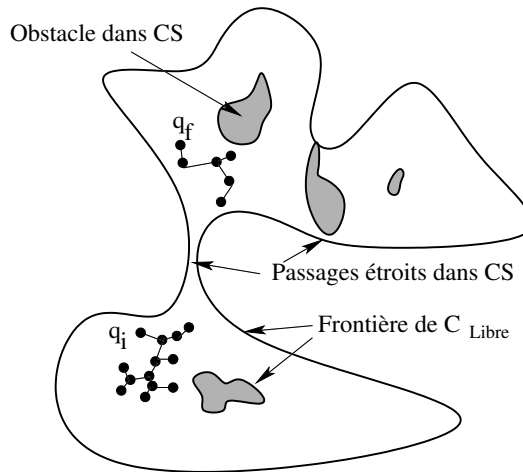


FIG. 3.1 – Illustration de la notion de passage étroit dans CS.

Il faut noter que lorsque des échantillons sont générés dans le passage étroit, ou proche d'une entrée, une méthode type *RRT* permet rapidement de faire diffuser l'arbre dans le passage étroit

pour trouver une solution. La difficulté pour l'algorithme est donc de trouver l'entrée du passage dans C_{Libre} .

Notre contribution à cette problématique est basée sur la construction d'un *RRT* qui utilise la notion des arbres locaux [Strandberg 04] avec la notion de visibilité de configuration dans C_{Libre} (principe des Visibility-PRM [Siméon 00]) pour contrôler la propagation des noeuds et des arbres dans la *roadmap*. Ce travail a constitué une partie de la thèse de David Flavigné [Flavigné 10].

Nous allons présenter l'intérêt et les inconvénients de ces méthodes avant de préciser comment nous pouvons combiner leurs principes pour améliorer la recherche de trajectoire solution.

3.1.1 Arbres locaux et notion de visibilité

Par définition, le *RRT* tend à se développer dans les régions inexplorées de C_{Libre} . Mais le coût de calcul du *RRT* augmente rapidement avec le nombre de passages étroits et, par conséquence, la différence de temps de calcul entre un *RRT* et un *PRM* décroît du même ordre.

L'idée de [Strandberg 04] est illustrée à la figure 3.2. Il introduit la notion d'arbres locaux, notés *LT*, pour augmenter les échantillons dans les régions difficiles de l'espace de recherche. Son algorithme de Local-Tree, noté LT-RRT, est un *RRT* qui comporte plus de deux arbres locaux et utilise la diffusion de ces nouveaux arbres pour sortir des passages étroits.

Cette approche est un compromis entre un pur *RRT*, si nous autorisons seulement la diffusion de l'arbre à partir de la configuration initiale, et une approche *PRM*, si chaque noeud introduit dans la *roadmap* peut être relié avec les autres noeuds mais ne peut pas être étendu.

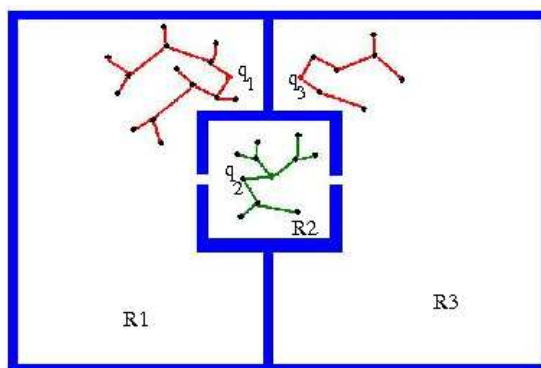


FIG. 3.2 – Un exemple d'arbres locaux. L'échantillonnage aléatoire a une forte probabilité de générer des noeuds dans les trois régions R_i . L'algorithme LT-RRT diffusera ces trois arbres dans les trois régions. La probabilité de connecter deux arbres sera plus forte s'il existe un arbre dans la région R_2 que s'il existe uniquement deux arbres dans les régions R_1 et R_3 .

Cette approche est séduisante mais nécessite la résolution des trois problèmes suivants :

- Quand faut-il créer un *LT* ?
- Quand est-il nécessaire de faire grandir un *LT* ?

– Quand faut-il fusionner deux *LT* ?

Il est évident qu’une réponse non appropriée à ces questions entraînera un temps de calcul important (à cause des nombreux tests de collision) et rendra l’approche inefficace.

[Strandberg 04] utilise une heuristique pour répondre à ces questions mais nous avons constaté une explosion combinatoire dès que le nombre d’arbres locaux augmente. Le temps de calcul croît rapidement lorsque le nombre de noeuds dans chaque arbre augmente ainsi que le nombre de connections possibles entre les arbres locaux.

Afin de pouvoir utiliser la notion d’arbres locaux, a priori intéressante pour capturer les passages étroits, il nous fallait pouvoir limiter le nombre de noeuds créés et le nombre possible d’arbres locaux.

Dans la littérature, les travaux de [Siméon 00] sont une variante de l’approche *PRM* dont l’objectif est de créer des *roadmap* avec un nombre minimum de noeuds. L’algorithme Visibility-PRM, noté VIS-PRM, utilise la notion de visibilité entre deux noeuds pour produire une *roadmap* de faible taille. Les auteurs définissent deux types de noeuds : les noeuds “gardiens” qui représentent la couverture de C_{Libre} et les noeuds “connecteurs” qui permettent de relier les gardiens entre eux.

L’algorithme ne garde que les configurations qui augmentent la visibilité de C_{Libre} (noeud gardien non visible des autres gardiens), ou permettent de connecter deux composantes de la *roadmap* (noeud connecteur).

Cette approche permet de générer des *roadmaps* qui présentent un faible nombre de noeuds mais avec l’inconvénient de trouver difficilement les passages étroits comme illustré sur la figure 3.3. Si deux noeuds gardiens sont visibles d’un passage étroit, la probabilité de créer un noeud dans la *roadmap* pour les connecter dépend de la taille du domaine de visibilité commun V_c . La probabilité de générer un échantillon dans V_c est beaucoup plus faible sur la figure 3.3 à droite que sur celle de gauche car les deux noeuds gardiens sont plus éloignés du passage étroit.

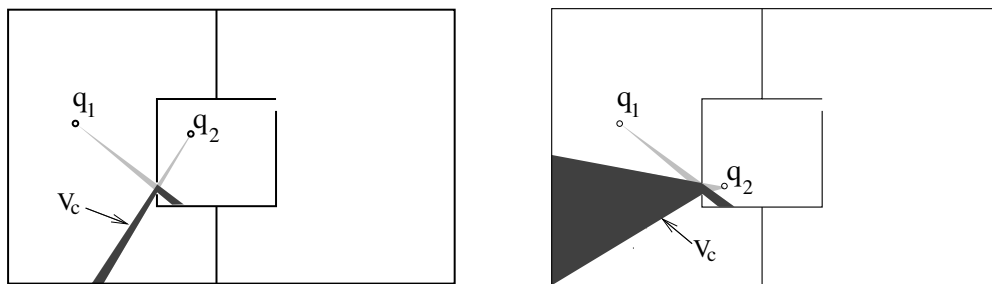


FIG. 3.3 – Région de visibilité entre noeuds gardiens en fonction de la proximité du passage étroit.

L’idée de notre méthode hybride, que nous appelons Visibility Local Trees (VIS-LT), est de combiner les stratégies des deux approches, VIS-PRM et LT-RRT afin de pallier aux inconvénients liés à ces approches.

3.1.2 Principes de l'algorithme VIS-LT

Comme les arbres locaux doivent être créés dans les régions non explorées, on peut utiliser la notion de noeud gardien pour savoir quand il faut créer une racine d'un nouvel *LT*.

La création de noeuds connecteurs permet de diminuer le nombre de composantes connexes de la *roadmap* en fusionnant les arbres locaux, de la même manière que dans le VIS-PRM.

Afin de résoudre le problème des passages étroits, il faut permettre et contrôler la diffusion des arbres locaux dont la racine va correspondre à un noeud gardien. Il nous faut permettre l'extension d'un arbre lié à ce noeud gardien, à la différence d'un VIS-PRM, mais en limitant son nombre de noeuds. Aussi, lors de l'expansion d'un *LT*, nous gardons seulement les noeuds qui augmentent le domaine de visibilité de l'arbre local, nous appelons ce troisième type de noeud, les noeuds "scouts", qui permettent d'accroître la visibilité de l'arbre.

Si la racine du *LT* est proche d'un passage étroit, la connection avec un autre *LT* sera rapide car le domaine de visibilité est grand ; dans le cas contraire, l'extension contrôlée à l'aide des noeuds scouts permet de se rapprocher des passages étroits.

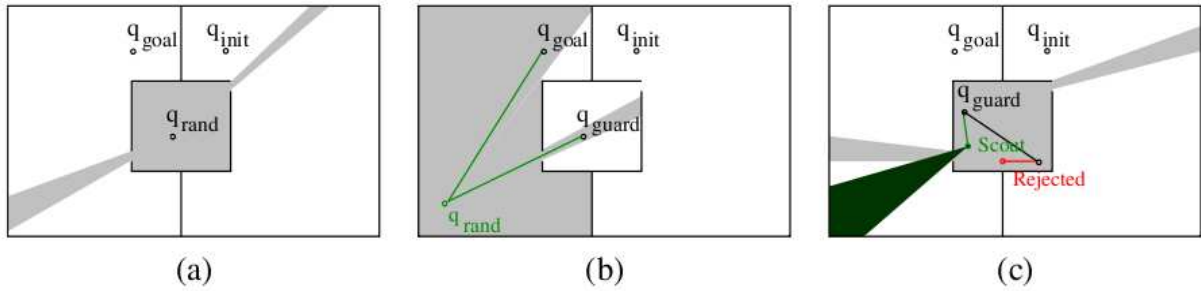


FIG. 3.4 – Illustration de la création des trois différents types de noeuds : gardien (a), connecteur (b) et scout (c)

A chaque génération de noeud aléatoire, l'algorithme VIS-LT, décrit dans [7], teste les connexions possibles avec les *LT* existants et génère ou non un type de noeud en fonction des connexions 3.3 possibles comme illustré sur la figure 3.4.

3.1.3 Résultats et conclusion

Dans l'exemple de la figure 3.5, le robot peut bouger en translation et en rotation sans contrainte dans l'espace (*free flyer robot*), ce qui se traduit par un CS de dimension 6. L'environnement est une succession de pièces reliées par de faibles ouvertures (au vu de la taille du robot) qui engendrent dans le CS une succession de passages étroits. Le problème est de passer par les différentes ouvertures pour calculer un chemin entre deux configurations. L'espace de recherche va se décomposer en des régions fortement contraintes et des régions libres importantes.

Sur la figure 3.5, la *roadmap* construite par notre approche est juxtaposée dans W sans tenir compte de l'orientation du robot et les murs extérieurs englobant les pièces ne sont pas représentés pour des raisons de visibilité de résultats.

Afin de valider notre approche, nous l'avons comparée¹ avec les algorithmes qui sont à la base de notre travail : le *RRT* (méthode de Laval), le *LT-RRT* (méthode de Standberg) et le *VIS-PRM* (méthode de Siméon). Nous pouvons voir sur le tableau de la figure 3.5 que *VIS-LT* résout le problème en des temps moindres, mais surtout avec un nombre de noeuds très inférieur au *LT-RRT* et *RRT*.

Lors d'une planification probabiliste, le choix des fonctions de base (échantillonnage, test de collision, plus proche voisin) a une grande influence sur les temps de calcul des méthodes probabilistes. Aussi, nous avons toujours utilisé les mêmes fonctions de base lors de nos tests de comparaison.

Algorithme	Iterations	Noeuds	Temps
RRT	12512	4004	492 s
LT-RRT	5358	2495	472 s
VIS-PRM	6063	13	648 s
VIS-LT	2098	465	135 s

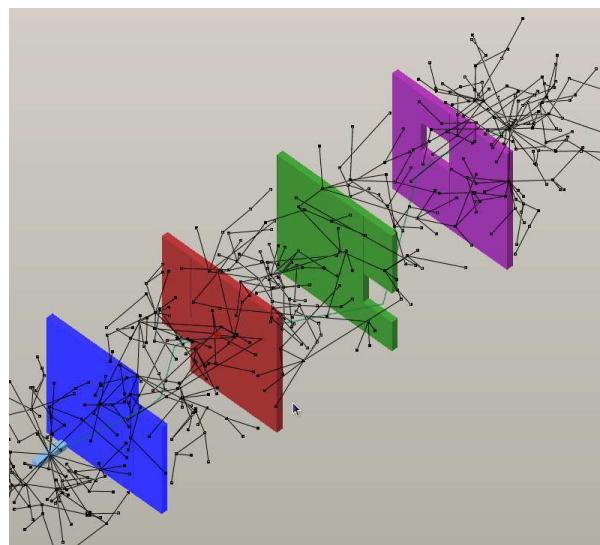


FIG. 3.5 – Exemple avec multiples passages étroits

Suite à une série de tests sur différents environnements, nous avons analysé le comportement de l'algorithme en fonction de différents critères en le comparant aux trois autres algorithmes déjà cités :

- Taille des passages étroits : ce paramètre permet de tester la difficulté du passage, nous avons pu constater que notre approche est bien moins dépendante de la taille de ces passages que les autres algorithmes sur lesquels il se base.
- Nombre de passages étroits : ce paramètre permet de refléter le comportement des algorithmes en fonction du nombre de composantes faiblement connexes. Ces tests nous ont permis de constater d'une part que l'augmentation du nombre de passages étroits est plus pénalisant que l'augmentation de la dimension de CS, d'autre part que *VIS-LT* contrôle bien la diffusion des arbres locaux à l'intérieur de chaque composante faiblement connexe et converge ainsi plus rapidement vers la solution.
- Nombre de composantes connexes : ce paramètre permet de comparer la création et la fusion des multiples composantes connexes du graphe en cours de construction. Là encore, le *VIS-LT* donne de meilleurs résultats, ce qui est cohérent avec les deux points précédents.

¹Dans la plateforme logicielle HPP (Human Path Planner) développée au LAAS-CNRS à partir de KineoWorks.

- Augmentation du nombre de noeuds : ce paramètre permet de mesurer l'intérêt de la création d'un noeud pendant l'exploration de CS. En faisant un test sur un exemple sans solution nous avons constaté que VIS-LT tend à diminuer sa création de noeuds tout comme VIS-PRM (qui donne la solution optimale).

L'analyse de ces résultats montre l'intérêt de notre contribution.

Notre objectif dans ce travail était de perfectionner les techniques de recherche automatique de trajectoires. En combinant les avantages de différentes approches, nous avons pu en développer une nouvelle, VIS-LT, qui permet d'améliorer efficacement la recherche pour un certain type d'environnement.

Le VIS-LT peut se classer entre une méthode construisant un graphe dense, coûteuse à construire, et une méthode construisant un graphe peu dense qui ne peut pas correctement représenter la connexité d'un environnement possédant de nombreux passages étroits. Si nous reprenons les deux points essentiels à résoudre par une approche probabiliste, à savoir obtenir une bonne couverture et une bonne représentation de la connexité de C_{Libre} , nous pouvons dire que notre approche apparaît comme un compromis intéressant.

Nous nous sommes attachés à améliorer la planification de trajectoires pour un objet rigide car notre objectif applicatif concernait la résolution de tâches d'assemblage comme nous allons le voir dans la section suivante, en y intégrant l'expertise d'un opérateur.

3.2 Planification interactive

L'objectif des méthodes de planification est de trouver automatiquement une trajectoire solution de manière générique. L'algorithme doit pouvoir trouver la (ou les) solution(s) quel que soit le type de robot et d'environnement. Nous avons vu dans le paragraphe précédent que de nombreux problèmes subsistent pour obtenir une méthode générique qui fonctionne dans tous les cas.

Dans l'exemple de la figure 3.6, la tâche robotique consiste à planifier l'insertion d'un objet quelconque dans le trou d'une plaque, ce qui revient à un problème d'insertion de clef dans une serrure. Pour introduire l'objet, nous allons devoir explorer un CS de dimension 6 (pour la clef), pour trouver l'entrée du trou à partir d'une configuration initiale. Trouver une trajectoire pour débiter l'insertion de l'objet revient à trouver un passage étroit dans CS, ce qui va demander un temps de résolution important.

Ce problème paraît très simple à un être humain, qui trouve immédiatement la trajectoire solution, alors qu'elle apparaît complexe pour les algorithmes de planification.

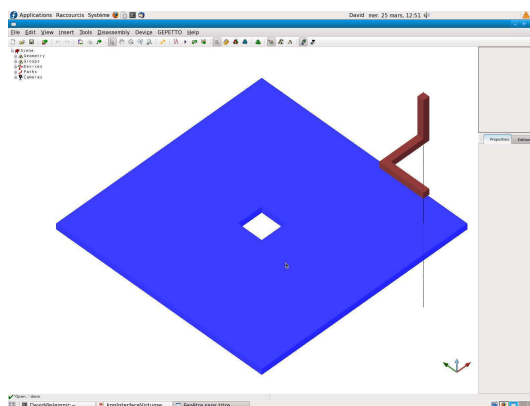


FIG. 3.6 – Exemple de type serrure

Si, avec le même exemple, la tâche robotique est de retirer l'objet déjà inséré dans le passage étroit, en fonction de la géométrie de l'objet comportant plusieurs concavités, l'opérateur humain va avoir plus ou moins de difficulté pour extraire la pièce, surtout si une partie de l'objet lui est cachée par la plaque. A l'opposé, ce problème sera facilement résolu avec une approche de type *RRT* modifié (voir[Ferré 04]).

A partir de ces deux exemples, nous constatons que la notion de *problème difficile* n'est pas la même pour un algorithme de planification et pour un utilisateur. Cela semble logique car un algorithme de planification n'est pas bâti sur un modèle neuronal de raisonnement mais à partir d'algorithmes respectant un traitement informatique des données représentant son modèle.

Partant de cette constatation, nous nous sommes attachés à étudier comment un utilisateur et un algorithme de planification pouvaient travailler ensemble à la résolution d'un problème en s'enrichissant mutuellement de leurs "compétences" pour développer un planificateur interactif. Le travail développé dans ce paragraphe sur la planification interactive se focalise sur des tâches d'assemblage et de désassemblage d'objets qui étaient au coeur du projet *AMSI*, projet ANR-

RNTL sur lequel nous étions impliqués.

Nous verrons comment nous avons abordé cette problématique sous deux aspects complémentaires avant de présenter des extensions possibles.

3.2.1 Planification et Interaction pour l'aide au mouvement

La conception des systèmes mécaniques induit une complexité de plus en plus croissante liée à la fois au nombre de pièces dans un volume réduit et aux différentes contraintes imposées sur le système. Cette conception est de plus en plus intégrée dans une démarche de *PLM, Product Life cycle Management*, qui tend à mettre en cohérence le cycle de vie du produit depuis sa conception jusqu'à sa fabrication et sa maintenance.

Comme le PLM gère, par différents logiciels, l'ensemble des données numériques relatives au produit, nous pouvons en extraire toutes les informations nécessaires à son utilisation.

Dans ce contexte, le problème mécanique d'assemblage consiste à calculer des trajectoires pour une ou plusieurs pièces mécaniques, issues d'un modèle numérique de données, appartenant à un ensemble plus complexe. La simulation de ces trajectoires permet de valider les phases de conception d'une part et les phases de maintenance d'autre part. Le critère essentiel de validation des trajectoires est celui de la non-collision entre les pièces mécaniques.

Cette phase de validation est primordiale pour réduire les coûts de conception si le modèle numérique est pertinent par rapport au prototype physique. Une erreur de conception doit être détectée le plus rapidement possible pour être remonté au niveau du process de conception général afin d'en limiter les impacts financiers et humains. De plus, ces phases d'assemblage doivent prendre en compte les "expériences métiers" des concepteurs afin d'en évaluer la pertinence.

Pour répondre à cette problématique, une première possibilité est de faire des simulations avec des techniques d'immersion en réalité virtuelle en utilisant des périphériques de manipulation haptique ou pseudo-haptiques [Lecuyer 04]. Les périphériques haptiques permettent à l'utilisateur de ressentir les forces et couples mis en jeu lors de la manipulation d'un objet dans l'environnement virtuel (contacts, frottements,...) et de tester ainsi la faisabilité d'une trajectoire. De nombreux projets, tels que Perf-RV 1 et 2, ont permis de développer des démonstrateurs et des prototypes utilisant la réalité virtuelle dans ce contexte en utilisant un couplage visuo-haptique ([Sreng 06]). L'utilisation d'un périphérique haptique en immersion nécessite un investissement important ; l'utilisation d'une souris 6D comme périphérique d'interaction dans le processus de conception est une alternative [Chen 05, Su 08]. Le gain en coût se fait bien sûr au détriment du retour haptique qui peut s'avérer nécessaire dans des environnements où le rendu visuel est peu explicite à l'utilisateur. A l'aide de ces périphériques, nous pouvons faire entrer directement l'opérateur dans la boucle par le biais des interactions temps réels à la fois visuelle et haptique. L'opérateur peut ainsi utiliser son intuition et son expérience métier pour définir et exécuter le mouvement de l'objet.

L'autre possibilité est d'utiliser un planificateur automatique de trajectoires comme décrit dans le paragraphe précédent, mais dans ce cas-là, il sera très difficile de prendre en compte les

contraintes utilisateur ou métier dans la recherche de solution.

L'objectif de notre travail est donc de rapprocher ces deux techniques de recherche de trajectoire d'assemblage de pièces mécaniques à partir de maquettes numériques. Ce travail a été au coeur du projet AMSI et constitue aussi une partie du travail de thèse de David Flavigné [Flavigné 10].

3.2.2 Planification interactive

L'idée de prendre en compte l'intention de l'utilisateur dans des algorithmes de planification se retrouve dans quelques travaux ([Bayazit 00, Rosell 08]) pour lesquels, en général, l'interaction entre l'utilisateur et la recherche automatique est simplifiée en découplant le processus en deux étapes distinctes. Dans une première étape, le planificateur recherche une solution, ensuite l'utilisateur intervient pour modifier cette trajectoire ou inversement.

Nous allons commencer par présenter un premier travail, fait en collaboration avec l'équipe de recherche DIDS de l'ENIT, basé sur l'idée d'une décomposition en deux étapes, avant de présenter nos derniers travaux liés au projet AMSI dans lequel nous intégrons l'interaction dans la boucle générale.

3.2.2.1 En environnement immersif

Dans le cadre d'un projet de recherche avec Alstom, le laboratoire LGP de l'ENIT s'est équipé d'une plateforme de réalité virtuelle permettant d'immerger l'utilisateur au sein de la simulation. Elle intègre un bras haptique Virtuoso 6D permettant la manipulation d'objet 3D avec un retour complet d'effort (voir figure 3.7).

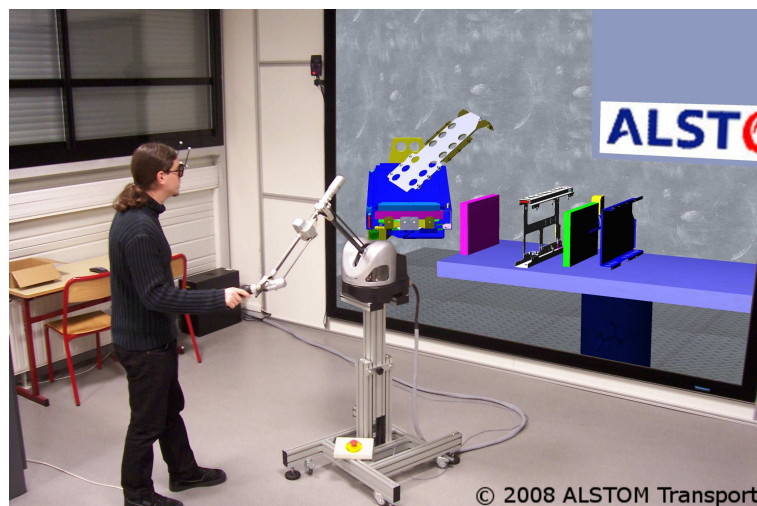


FIG. 3.7 – Environnement immersif pour l'assemblage à l'ENIT

Notre travail avec l'ENITn'a concerné que la partie notée *Path Planner* sur la figure 3.8. L'idée de base était de pouvoir construire un planificateur "temps réel" et interruptible par l'utilisateur dans un contexte immersif. La recherche de chemin solution devant s'exécuter en moins d'une seconde, il était impossible d'appliquer directement une méthode de type *RRT* sur l'ensemble de l'espace des configurations.

Aussi, nous avons choisi de décomposer l'étape de planification en trois sous-étapes :

- Initialisation d'un *Octree* de la structure de données de l'environnement qui traduit l'organisation de l'espace libre en fonction de la taille de l'objet manipulé,
- Calcul d'un chemin solution avec un algorithme *A** dans *Octree* afin de déterminer dans quelles parties il faut autoriser la recherche de trajectoire solution,
- Recherche de la solution avec un *RRT* bi-directionnel.

Une force de guidage est générée par ce chemin solution pour attirer l'objet vers ce dernier et amener l'utilisateur à le suivre. Si l'utilisateur n'est pas satisfait du chemin de guidage, il peut sortir de la zone de guidage autour du chemin, ce qui inhibe l'asservissement du bras haptique sur la trajectoire et relance automatiquement une nouvelle recherche par le *A**.

L'interaction utilisateur/algorithmique se fait en deux étapes, mais cette option permet d'offrir une première solution satisfaisante en prenant en compte les différents temps de calcul. Le lecteur peut trouver les détails du travail dans [16] ou [Ladezeve 10].

3.2.2.2 Dans un contexte de bureau d'étude

L'originalité de notre approche est de décrire une collaboration entre l'utilisateur et le calcul automatique pour laquelle la simulation interactive est considérée comme une aide au logiciel de planification, tandis que la planification de mouvement devient une aide pour l'opérateur. Cette approche interactive doit permettre à l'opérateur de contrôler le processus de recherche au niveau de l'algorithme lui-même. Il faut donc intervenir directement sur le développement des noeuds dans une recherche de type *RRT*. De même, l'algorithme doit retourner à l'utilisateur des informations via un retour haptique, des métaphores visuelles ou auditives.

Contrairement à l'approche précédente qui nécessite une immersion en environnement virtuel, nous avons voulu privilégier ici une utilisation non immersive où l'opérateur peut utiliser le

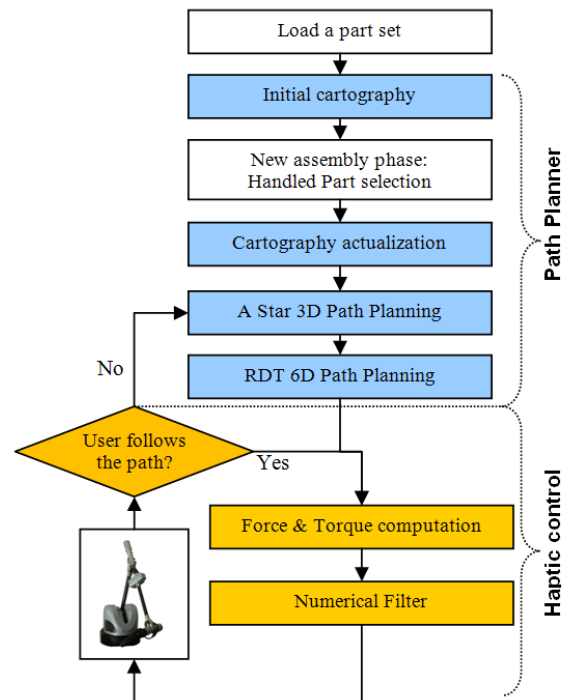


FIG. 3.8 – Architecture générale d'interaction utilisateur-algorithme [14]

planificateur interactif à partir d'un simple écran. De plus, nous n'avons pas souhaité utiliser exclusivement un périphérique haptique mais présenter une approche générique utilisable aussi bien avec une souris 6D qu'avec un bras haptique pour déplacer l'objet que nous nommons *avatar*. Pour le retour d'effort, nous utilisons le terme de *pseudo-force* car si le périphérique interactif est une souris 6D, ces pseudo-forces virtuelles sont calculées à partir du mouvement de la souris, sinon elles sont directement lues à partir du bras haptique.

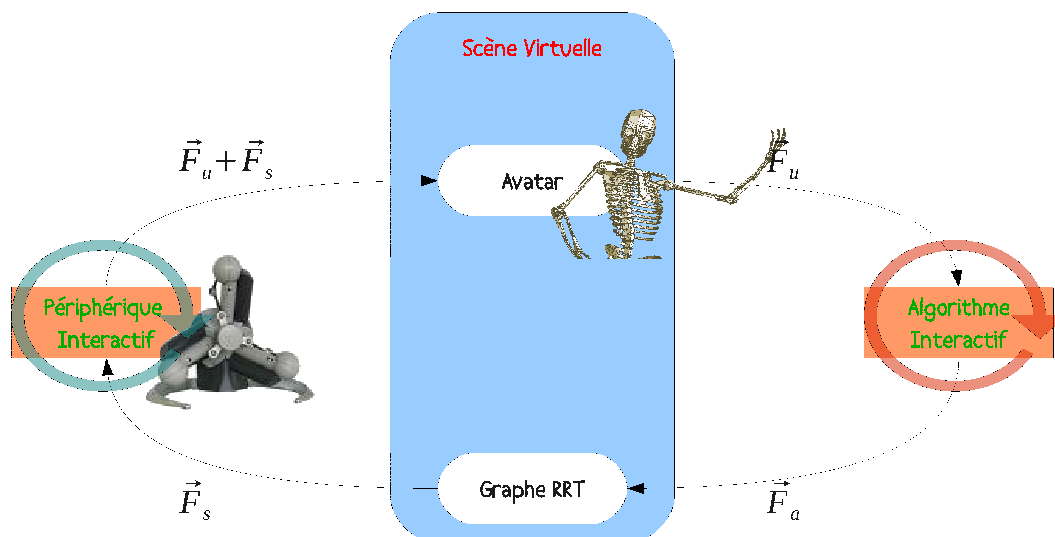


FIG. 3.9 – Boucle haut niveau de l'interaction entre l'utilisateur et l'algorithme de recherche automatique

Voici les étapes principales de l'algorithme interactif noté *IRRT* :

- On utilise l'intention de l'utilisateur pour calculer une pseudo-force F_u ;
- On calcule une pseudo-force F_a pour attirer l'avatar vers un sous-ensemble de noeuds du graphe en construction. F_a est transformée en une force F_s applicable sur l'avatar ;
- A partir de F_a et F_u , nous modifions l'échantillonnage afin de tenir compte d'une direction privilégiée ;
- Au fur et à mesure de la construction du graphe, nous étiquetons les noeuds de manière différente afin de pouvoir retourner à l'utilisation des informations sur la progression automatique du graphe.

Nous pouvons distinguer trois modes de fonctionnement de la boucle de haut niveau en fonction de l'utilisateur :

1. L'utilisateur déplace plus rapidement le périphérique interactif que l'algorithme ne se développe, auquel cas le développement du graphe poursuit l'avatar. C'est le mode dans lequel l'intention de l'utilisateur est prépondérante.
2. L'utilisateur ne bouge pas, ou peu, le périphérique alors que l'algorithme retourne des informations pendant son exploration. Dans ce mode, l'utilisateur ne sait plus comment faire avancer l'avatar et il peut utiliser le retour haptique ou visuel de la progression de l'algorithme.

3. L'utilisateur se déplace à une vitesse proche de celle de l'avatar. L'algorithme va développer le graphe dans la direction du mouvement de l'avatar mais en explorant des zones en dehors de l'intention de l'utilisateur.

Les boucles de l'utilisateur et de l'algorithme interactif n'étant pas bloquantes elles peuvent travailler à des fréquences différentes. Dans ce fonctionnement, lorsque l'utilisateur bouge l'avatar, son intention est prise en compte au travers de l'échantillonnage effectué ; un retour sensoriel ou visuel de la progression de l'algorithme l'aide dans sa prise de décision.

Nous pouvons voir sur la figure 3.10 la solution pour le problème présenté dans l'introduction de cette partie. La densité des noeuds traduit la dextérité de l'utilisateur à guider l'avatar.

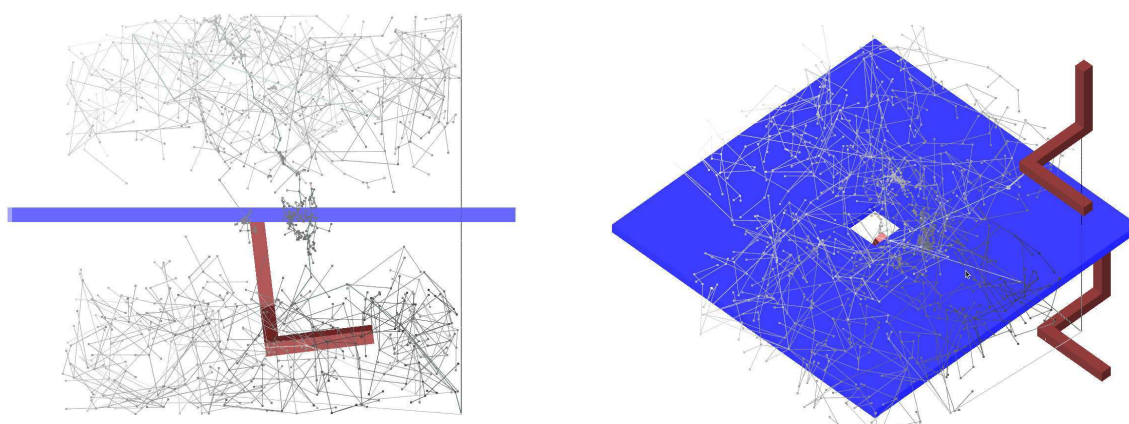


FIG. 3.10 – Un problème en coopération en dimension 6

3.2.3 Conclusion

L'intérêt du premier travail était de montrer la possibilité d'assister un utilisateur, immergé dans un environnement virtuel, grâce à des méthodes de planification automatique lors de tâches de manipulation en phase de conception de produit. La possibilité d'intégrer nos travaux dans un environnement complet d'immersion en 3D a permis de tester et valider l'approche en condition réelle. Notre méthode permet de réduire fortement les temps de calcul par rapport à un algorithme automatique de type *RRT* pour la recherche de solution dans un espace de dimension 6, cependant, elle est trop dépendante de la complexité de la scène. De plus, nous pouvons constater que l'interaction utilisateur/planification n'est pas en continu car, comme la recherche de solution ne se fait que par l'exploration d'une classe d'homotopie de solutions possible dans *CS* liée au résultat des cellules d'octree retournées par le A^* , il est inutile de continuer à faire explorer ces cellules par le *RRT* en phase de guidage du bras. C'est un point pénalisant de notre approche que nous pressentions mais qui n'était pas bloquant dans ce premier travail.

C'est pourquoi, en parallèle, nous avons exploré une autre voie pour définir un véritable planificateur interactif *IRRT* permettant d'avoir une boucle de *feedback* continue réglable entre

utilisateur et algorithme. L'intérêt du *IRRT* est d'utiliser les directions de déplacement de l'utilisateur pour biaiser l'échantillonnage et, ainsi, guider l'aléatoire. Il est bien évident que le simple retour visuel retourné par l'algorithme à l'utilisateur ne suffit pas. Il faudrait l'enrichir par de véritables métaphores haptiques et auditives en fonction de la progression de l'arbre de recherche et des échecs de développement. Une des difficultés est d'analyser le comportement de notre algorithmique car cette méthode est fortement dépendante des capacités de l'utilisateur à manipuler le périphérique et à percevoir le contexte de la scène 3D ainsi que des informations retournées par le graphe. Nous avons effectué de nombreuses simulations en faisant varier les paramètres qui nous semblaient les plus pertinents, à savoir la vitesse de progression du périphérique qui traduit la dextérité de l'utilisateur et la méthode d'échantillonnage ainsi que l'écart type associé qui traduisent la manière dont on veut influencer le guidage de l'opérateur. En conclusion, nous pouvons dire que ces paramètres sont importants sur le temps de résolution et qu'ils doivent être adaptés à la dextérité de l'utilisateur. Le *IRRT* est donc un algorithme paramétrable en fonction de la complexité de la tâche et de la dextérité de l'opérateur.

Comme la majorité des approches probabilistes, nous obtenons une solution parmi toutes les classes d'homotopie de solutions possibles. Si nous pouvions calculer l'ensemble des solutions nous pourrions prendre en compte plus facilement les critères métier sur le choix de la solution finale. De plus, nous pourrions comparer les solutions et les performances en fonction de l'influence plus ou moins grande de l'utilisateur par rapport à l'algorithme.

Un des problèmes qu'il reste à résoudre est la forme des chemins solutions qui sont en général une interpolation linéaire entre les configurations. Afin de rendre ces trajectoires locales plus réalistes, nous nous sommes intéressés aux caractéristiques et aux invariants du mouvement chez l'homme que nous allons décrire par la suite.

3.3 Mouvement naturel et robotique

Les méthodes de planification automatique de trajectoires fournissent des trajectoires sous forme d'une suite de chemins locaux. Ces chemins locaux respectent des contraintes d'optimisation sur le système mais sont très éloignés d'une trajectoire que réalise un opérateur humain. Si nous voulons que le planificateur donne une solution réaliste (semblable à celle d'un humain), il est nécessaire de s'intéresser au problème de la cognition spatiale chez l'homme.

Nous avons travaillé avec une équipe de neurobiologistes du CERCO impliquée dans l'étude des processus de représentation de l'espace et du mouvement. Cette collaboration s'est poursuivie par le projet ANR-RobEA *Egocentre* (2002-2005). Ce travail s'est concrétisé par le co-encadrement de la thèse de Minh Tuan (2006-2009) qui porte sur une application des principes moteurs neurobiologiques pour le contrôle du geste d'atteinte des robots humanoïdes. Le modèle est principalement basé sur les hypothèses que l'énergie des signaux des motoneurones est en continu minimisée le long de la trajectoire et que les efforts dynamiques et statiques sont traités séparément. Nous utilisons ce paradigme pour contrôler le robot humanoïde HRP2 en considérant un modèle dynamique du bras qui inclut, pour chaque degré de liberté, deux filtres passe-bas du second-ordre modélisant le système neuromusculaire défini par une paire de muscles antagonistes. Nous montrons que cette approche permet de générer des mouvements réalistes qui respectent les caractéristiques du mouvement humain.

Notre objectif, dans ce travail, était de comprendre les grands principes computationnels neurobiologiques pour évaluer leur application en robotique.

La robotique est par essence pluridisciplinaire et de nombreux travaux visent à concevoir des lois de commande biologiquement inspirées pour le contrôle du mouvement sur des robots. Dans le cas des robots humanoïdes, qui possèdent un grand nombre de degrés de liberté et qui doivent effectuer des trajectoires similaires à l'homme, cette approche peut donner lieu à d'intéressantes alternatives aux approches robotiques plus classiques.

3.3.1 Enchaînement de tâches et primitives sensorimotrices

Nous avons vu dans la partie 2.3 que, pour rendre une trajectoire planifiée robuste aux incertitudes, il était nécessaire de prendre en compte les amers de l'environnement sur lesquels le système va pouvoir se référencer. Cette notion de référence sur des amers se trouve au centre de la communauté robotique de la commande référencée-capteur qui a pour objet l'élaboration de boucles de contrôle établissant un lien direct entre le mouvement du robot et la variation d'un ensemble de données sensorielles.

L'exécution de chaque tâche élémentaire est généralement plus robuste aux erreurs de modélisation et de mesure que les méthodes nécessitant la localisation absolue du robot puisque chaque tâche garantit un comportement du robot en fonction des perceptions dans son plan capteur. L'inconvénient de cette approche est que chaque tâche est locale puisqu'elle est définie relativement à la perception d'un amer donné. Pour pouvoir exécuter des mouvements plus complexes, il est nécessaire de pouvoir enchaîner les tâches de manière dynamique afin de conserver la

fluidité du mouvement.

Il faut donc pouvoir unifier la définition des tâches sensori-motrices alors qu'elles impliquent des référentiels distincts et des données de nature différente, et relier le problème de régulation locale à la définition globale du déplacement. Nous nous étions intéressés à cette question à travers les travaux de thèse de [Malti 05] mais en nous focalisant sur le résultat de la planification globale sans prendre en compte les spécificités des tâches locales.

Les premiers contacts avec la communauté des neuro-sciences dans le projet de recherche *Ego-centre* nous ont permis de comprendre l'importance des mécanismes de changement de coordonnées intervenant dans la mise en correspondance des différentes cartes sensorielles. Il existe par exemple, à différents niveaux du cortex, des systèmes de références centrés sur la rétine (rétinotopique), sur l'oeil (oculocentré), sur le crâne (craniocentré) et sur le corps (égocentré) qui vont nécessiter des transformations successives. Des travaux récents montrent l'existence au niveau pariétal d'un codage multi-sensoriel dans un référentiel oculocentré qui prouve le rôle important de la vision chez le primate. En nous inspirant de ces éléments, nous avons proposé dans le projet *Egocentre* un cadre théorique pour modéliser le problème d'enchaînement de tâches référencées-capteurs pour des systèmes robotiques constitués d'une chaîne cinématique simple [31], [33].

Cette première confrontation avec le monde des neuro-sciences nous a permis de voir les liens possibles avec la robotique et c'est tout naturellement que nous avons poursuivi nos travaux dans ce domaine en envisageant d'appliquer des principes moteurs biologiquement inspirés à une tâche robotique.

3.3.2 Commande de mouvement biologiquement inspirée

Dans le cadre de la thèse de Tran Minh Tuan [Tran 09], co-encadré avec Philippe Souères au LAAS-CNRS, nous avons voulu appliquer des principes de contrôle moteur humain pour un mouvement d'atteinte à un robot humanoïde. La première partie de ces travaux a été menée en collaboration avec Emmanuel Guigon, CR dans l'équipe *ANIM* de l'*INSERM* U742.

Bien que la structure mécanique des robots humanoïdes soit très éloignée de la structure squelettique humaine, la connaissance de principes neurocomputationnels du mouvement semble un moyen prometteur de produire des mouvements anthropomorphes. Plusieurs auteurs ont proposé d'utiliser ces principes pour la commande des robots, en résolvant le problème de la redondance de ces systèmes de manière appropriée. Certains travaux ont tenté de reproduire la structure musculo-squelettique du bras humain et de développer des stratégies de commande reposant sur la dynamique des forces internes [Simmons 05], ou de répliquer des modèles biologiques d'activité musculaire [Northrup 01]. D'autres ont tenté de caractériser les trajectoires en cherchant à minimiser une fonction de coût particulière telle que la variance [Simmons 05], le couple d'effort [Matsui 06, Park 07], ou encore le jerk [Sha 06]. Des techniques de capture de mouvement ont également été utilisées par de nombreux auteurs pour reproduire des mouvements réalistes [M. 00, Fujita 04]. Enfin, d'autres équipes ont mis l'accent sur le choix d'architectures de contrôle visant à reproduire la capacité du cerveau à utiliser plusieurs référentiels et

sa plasticité [Hersch 06].

L'objectif de tous ces travaux est d'arriver à faire exécuter à un robot des trajectoires qui reproduisent des caractéristiques du mouvement naturel en respectant les invariants du mouvement. Par exemple, dans une tâche de saisie, nous savons que la trajectoire est quasi-rectiligne avec un profil de vitesse en cloche presque symétrique. La question est donc : comment construire une commande qui respecte ces invariants ?

Nous avons choisi de baser nos travaux sur un modèle du contrôle moteur récemment proposé par Emmanuel Guigon [Guigon 07] qui généralise une grande partie des modèles antérieurs et particulièrement bien adapté à la commande des robots humanoïdes (les variables d'entrée de commande sont les paramètres de position angulaire des liaisons). Ce modèle repose sur quatre principes de base qui donnent une description unifiée du comportement moteur. Ces principes sont :

1. Séparation : le système nerveux central opère un traitement séparé des efforts dynamiques (inertiels et dépendants de la vitesse) et des efforts statiques (élastiques, gravitationnels).
2. Contrôle optimal en boucle fermée : l'énergie des signaux neuronaux des muscles est continuellement minimisée le long des trajectoires.
3. Efficacité maximale : l'objectif moteur est d'atteindre le but avec une erreur nulle tout en minimisant le signal de commande.
4. Effort constant : tous les mouvements obéissant à un même ensemble d'instructions sont exécutés avec le même niveau d'effort.

Le principe de séparation, constaté lors d'études électromyographiques, exprime le fait que l'activité musculaire peut être décomposée en deux éléments : une composante tonique qui permet de compenser les efforts statiques, tels que les couples antigravitaires, et une composante phasique qui produit les efforts dépendants de la vitesse nécessaires pour accomplir l'acte moteur. Sur cette base, le principe d'optimisation permet de déduire les variables du contrôleur phasique minimisant l'énergie du signal des motoneurones.

Cette approche est particulièrement bien adaptée à la commande des robots humanoïdes, tels que le robot HRP2, car sur ce type de robot, la commande de mouvement asservit en position chaque liaison de manière indépendante à une valeur angulaire de référence. En effet, la robustesse des boucles d'asservissement bas niveau de chaque liaison permet de valider l'hypothèse que les forces statiques sont exactement contrebalancées. Nous pouvons donc synthétiser une commande sans considérer les efforts statiques. De cette manière, le problème d'optimisation peut être résolu sur la base d'un modèle simplifié comprenant seulement la composante phasique. Pour pouvoir implémenter ce modèle de commande sur le robot humanoïde HRP2, nous avons développé un modèle dynamique du bras du robot incluant, pour chaque degré de liberté, un filtre du second ordre représentant la dynamique du système neuromusculaire associé à une paire de muscles antagonistes, suivant le modèle proposé dans [der Helm 00]. Les entrées de commande de notre système représentent alors le signal de commande de motoneurones virtuels. Pour ce modèle, nous avons développé un algorithme efficace pour caractériser les trajectoires correspondant à la minimisation de l'énergie des signaux neuronaux. Pour cela, nous avons transformé le problème de commande optimale en un problème de programmation non

linéaire en utilisant la méthode de transcription directe que nous avons résolue à l'aide du solveur numérique Ipopt [Wächter 06].

Nous obtenons ainsi hors ligne une trajectoire ayant les caractéristiques d'une trajectoire humaine entre une configuration initiale du bras et une position finale de la main, à savoir des déplacements quasi-rectilignes de l'effecteur avec des courbes de vitesse en cloche présentant un pic unique.

Nous pouvons voir sur la figure 3.11 un ensemble complet de trajectoires de positions d'atteinte exécutées de la main du robot HRP2 comportant six d.d.l. dans le plan frontal. Cette approche permet également d'exécuter des mouvements de grande amplitude avec une très bonne précision.

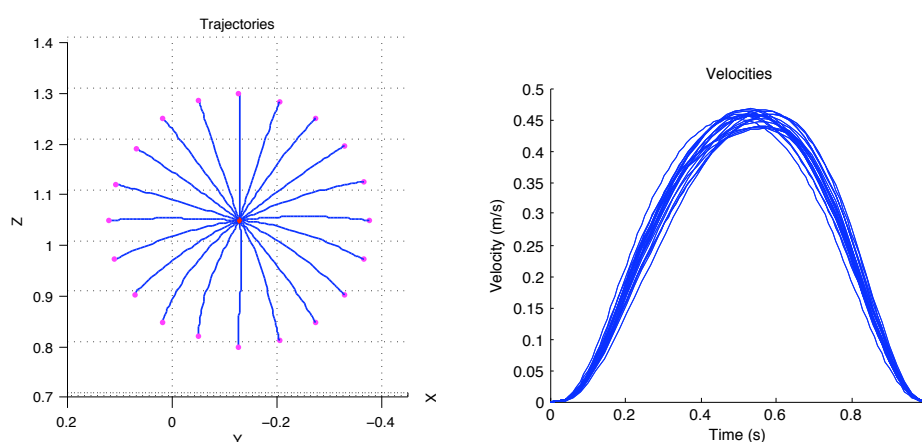


FIG. 3.11 – Mouvement d'atteinte de la main de HRP2 dans le plan frontal (à gauche) et courbes de vitesse associées (à droite)

Lors d'un mouvement de saisie, le mouvement de la tête et du torse jouent un rôle important, aussi nous avons étendu notre approche à un système simplifié de bras + torse + tête comportant neuf d.d.l.. Cet exemple (figure 3.12) montre la généricité de notre approche qui permet aussi d'inclure des contraintes du type butées articulaires. Il met également en évidence le coût calculatoire de la commande qui augmente fortement avec le nombre de d.d.l., ce qui implique un calcul hors-ligne incompatible avec une commande en boucle fermée ou une intégration dans un schéma de planification de trajectoire.

3.3.3 Mouvement d'atteinte visuellement guidé

Le travail réalisé précédemment fait l'hypothèse implicite que la position de la cible par rapport au corps est une donnée de l'entrée du problème. L'identification du repère utilisé par le Système Nerveux Central (SNC) pour le codage du mouvement est un sujet très débattu dans la communauté des neurosciences et nous avons déjà évalué l'importance du choix du repère dans le projet *Egocentre*. Les résultats expérimentaux tendent à montrer que l'erreur entre la position

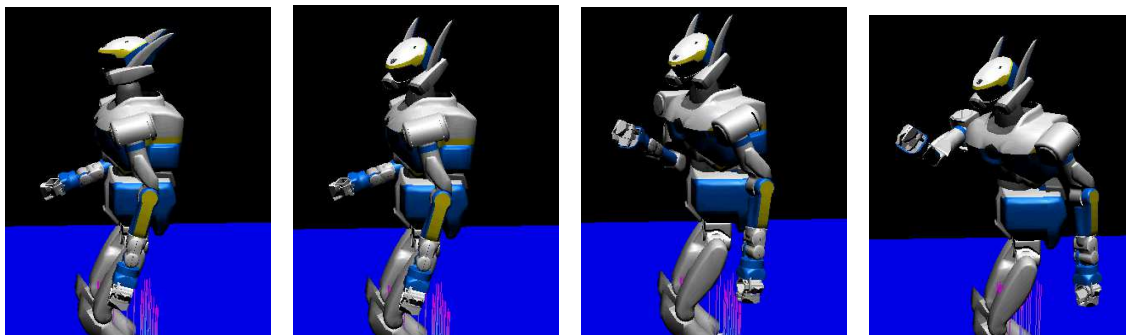


FIG. 3.12 – Mouvement d’atteinte incluant le torse et la tête

de la main et la cible, erreur appelée *vecteur de différence*, est la donnée d’entrée utilisée par le SNC pour planifier le mouvement d’atteinte [Shadmehr 05].

Si nous considérons un mouvement visuellement guidé, nous savons que la position de la cible est initialement construite dans le référentiel rétinien. Pour pouvoir coder le mouvement de la main pour atteindre la cible, il faut savoir quel est le référentiel utilisé par le SNC pour coder ce vecteur de différence. Le SNC effectue-t-il une transformation de la position de la cible dans un repère corps-centré ou bien construit-il son plan moteur dans un repère oculo-centré ?

L’objectif de notre travail était d’apporter des arguments de nature computationnelle pour contribuer au débat controversé relatif à l’identification du référentiel utilisé par le système nerveux central pour le codage de cette tâche. Nous tenons à préciser que notre objectif n’était pas de proposer un modèle biologiquement plus plausible que les modèles existants, mais plutôt de comparer les transformations nécessaires à la mise en oeuvre de deux schémas de commande, l’un corps-centré et l’autre oculo-centré.

Dans les deux cas, nous avons utilisé des principes éprouvés de contrôle moteur pour reproduire les principales caractéristiques cinématiques du mouvement humain tout en prenant en compte le retard et les biais sensoriels. Les deux schémas de commande décrits dans [10] utilisent les mêmes lois de commande pour actionner le bras à partir d’un vecteur de différence exprimé dans des repères différents et une même loi de mouvement pour contrôler le mouvement de la tête et de l’oeil.

En première approximation, les deux schémas de commande fournissent des résultats assez similaires avec des trajectoires qui respectent les caractéristiques cinématiques du mouvement humain. Mais dans une étude plus détaillée, nous avons pu mettre en évidence les points suivants :

1. D’un point de vue computationnel, la transformation du vecteur de différence main-cible dans un référentiel corps-centré n’est pas nécessaire pour la réalisation d’un contrôle précis du mouvement ;
2. Le fait d’exprimer le vecteur de différence dans un référentiel corps-centré peut induire une incurvation de la trajectoire ;
3. Le schéma de commande corps-centré se révèle moins robuste par rapport aux biais proprioceptifs et aux délais sensoriels que le schéma oculo-centré.

Nous pouvons voir l'effet du biais et du retard sensoriel sur des trajectoires d'atteinte, sur la figure 3.13, dans le plan sagittal (à gauche) et dans le plan frontal (à droite).

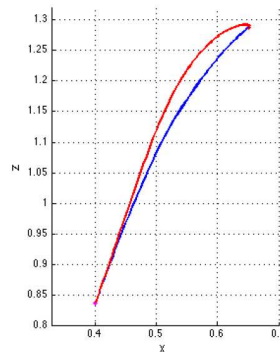


FIG. 3.13 – Mouvement d'atteinte dans un schéma oculo-centré (en bleu) et un schéma corps centré (en rouge)

Notre travail apporte des arguments de nature calculatoire pour renforcer l'idée que le SNC utilise un codage oculo-centré du vecteur de différence, plutôt que d'exprimer cette erreur de consigne dans un référentiel corps-centré. On évite ainsi le choix de ce repère corps-centré (torse, hanche, bassin,...) et de multiples transformations qui sont entachées d'erreur et temporellement décalées. Nous avons également montré que ce résultat était aussi valide dans le cas d'une cible mobile.

Ce travail, présenté à la conférence Progress in Motor Control 2009 de neurosciences du mouvement [13], a été sélectionné parmi les cinq meilleurs papiers de la conférence IEEE Robio 2009 [10].

3.3.4 Intérêt des primitives motrices

Il est largement admis que les mouvements chez l'homme minimisent un critère, même si ce critère n'est pas connu précisément et que de nombreux travaux essayent de le déterminer en résolvant des problèmes d'optimisation inverse. Mais il semble difficile d'imaginer que le SNC doit résoudre à chaque mouvement un problème d'optimisation en temps réel. Beaucoup de neurobiologistes pensent que le cerveau mémorise des stratégies ou des primitives motrices lors de son apprentissage et qu'il utilise ces primitives pour produire de nouveaux mouvements plus complexes.

L'idée d'une structuration de la commande en primitives est très intéressante pour les roboticiens car elle permet de résoudre le problème de la redondance du robot par rapport à la tâche et par conséquent la coordination des liaisons qui est un problème difficile et coûteux en temps de calcul. Il faut alors pouvoir résoudre le problème de la synthèse d'une base canonique de mouvements à partir desquels des mouvements plus complexes pourront être générés. Une fois que la base de primitives motrices est caractérisée, la complexité du problème de commande est simplifiée car il suffit d'identifier les facteurs représentant la contribution de chaque primitive

au lieu de déterminer directement les signaux de contrôle. Le nombre de primitives doit être très petit par rapport au nombre de paramètres nécessaires pour coder les signaux de contrôle. Ainsi, il suffit d'identifier le poids de chaque primitive pour réaliser un mouvement, ce qui implique beaucoup moins de variables qu'en requiert le calcul des signaux complets de commande.

Nous avons souhaité reproduire des mouvements optimaux à partir d'une approche *primitive motrice* en nous basant sur nos travaux effectués sur la génération de mouvements d'atteinte (voir paragraphe 3.3.2). L'objectif était de proposer une méthode permettant de générer les mouvements plus rapidement que par des techniques d'optimisation sans utiliser une base de données de type *lookup table* qui est obligatoirement de taille trop importante.

Il nous fallait donc savoir si nous pouvions :

- Caractériser un ensemble réduit de primitives qui représentent un large répertoire de mouvements ayant les caractéristiques du mouvement humain sur un bras comportant six d.d.l., étape de caractérisation ;
- Produire de nouveaux mouvements à partir des primitives obtenues qui présentent de bonnes performances en terme de précision et de temps de calcul, étape de généralisation.

Caractérisation : nous avons décidé d'extraire notre ensemble minimal de primitives Φ_i à l'aide de deux bases de données. Une, générée par simulation à partir de l'algorithme d'optimisation basé sur le modèle de Guigon appliqué au bras de HRP2 à six d.d.l. (voir paragraphe 3.3.2) et une seconde, construite à partir de mouvements enregistrés au moyen d'un système de capture de mouvement.

A partir de ces deux bases de données, nous avons utilisé deux approches différentes pour extraire les primitives Φ_i . Une, basée sur l'analyse en composante principale (PCA), qui est une technique efficace fréquemment utilisée en neurosciences et en robotique. Et une seconde approche, plus originale, (OD) qui consiste à formuler le problème d'extraction des primitives comme un problème d'optimisation directe.

Pour les deux approches, l'erreur quadratique moyenne, entre les données originales et les données reconstruites, décroît rapidement en fonction du nombre K de primitives. Nous constatons que l'erreur devient faible dès que $K > 10$ et qu'il est inutile de prendre $K > 20$.

En comparaison, la méthode PCA donne un meilleur résultat que la méthode OD dès que $K > 14$, aussi nous avons décidé d'utiliser cette méthode pour la détermination des primitives qui vont être utilisées dans l'étape suivante.

Généralisation : pour trouver les coefficients α_i , permettant d'exprimer la trajectoire souhaitée comme une combinaison linéaire des primitives Φ_i , nous avons commencé par utiliser un modèle d'apprentissage de type réseau de neurones. Les résultats obtenus ne nous ont pas permis de générer de nouveaux mouvements de manière satisfaisante (erreur finale importante). Il est fort possible que l'apprentissage par réseaux de neurones ne prenne pas en compte la sensibilité des variables α_i qui a une grande importance sur le mouvement. Nous avons alors proposé une méthode originale, consistant à calculer le poids des différentes primitives, en minimisant l'écart entre la trajectoire engendrée et une trajectoire de référence de type minimum jerk qui respecte les caractéristiques cinématiques du mouvement humain [8].

Il est bien évident que le résultat dépend fortement du nombre K de primitives choisies pour

synthétiser le mouvement. Nos résultats montrent que l'erreur d'atteinte à l'instant final diminue alors que le temps de calcul augmente en fonction de K croissant. Sur des mouvements d'atteinte qui ont une longueur réduite (moins de 1m), nous montrons qu'au-delà de $K = 12$ le gain en précision est quasiment nul (précision de l'ordre du centimètre).

Si nous comparons cette approche avec l'approche d'optimisation basée sur le modèle de Guigon (paragraphe 3.3.2) nous divisons nos temps de calcul par un facteur 6 (20 s au lieu de 2 mns). Sachant que notre implémentation de la partie généralisation a été faite sous Matlab sans aucune optimisation, nous pouvons envisager encore un gain de temps d'un facteur 10 si cette partie était écrite avec un code C/C++ optimisé. Avec des temps de calcul de l'ordre de la seconde, nous ne sommes plus très loin d'une utilisation possible en boucle fermée, ce qui était inenvisageable avec le modèle de Guigon.

3.3.5 Conclusion

L'objectif de ces travaux était d'étudier les approches du contrôle moteur humain afin de proposer des méthodes de contrôle efficace pour des robots humanoïdes en vue de mieux comprendre le mouvement humain. Pour cela, nous nous sommes restreints au mouvement d'atteinte ce qui nous a permis d'évacuer le problème de la locomotion qui est un problème à part entière.

Nous sommes bien conscients des grandes différences entre un humain et un robot humanoïde, que ce soit au niveau de sa structure mécanique, de ses capteurs de perception, de ses actionneurs ou de ses capacités de traitement de l'information. Néanmoins, ces travaux ont été bénéfiques sur plusieurs points.

De manière locale, ce travail autour de l'application à la robotique humanoïde de principes du contrôle moteur nous a permis de :

- Montrer que l'utilisation d'un modèle biologiquement inspiré permet de faire exécuter des mouvements réalistes à un robot humanoïde. En partant d'un geste d'atteinte du bras du robot HRP2, nous avons étendu ce modèle pour contrôler des mouvements incluant le bras, la tête et le torse. Ces résultats, qui se basent sur le principe de la séparation des forces et la minimisation d'énergie des motoneurones, sont un modèle de contrôle intéressant pour la robotique humanoïde.
- Comparer les mouvements générés dans le référentiel oculo-centré et corps-centré en tenant compte des biais et des retards sensoriels qui sont connus expérimentalement. Même si la forme globale des trajectoires est semblable, les mouvements produits dans le référentiel oculo-centré sont plus robustes aux biais et aux retards.
- Montrer la faisabilité d'une approche basée sur des primitives motrices qui permet de calculer des mouvements en se rapprochant de la contrainte de temps réel. Un petit ensemble de primitives peut être extrait à partir d'une base de mouvements simulés ou expérimentaux et utilisé pour se généraliser à n'importe quel mouvement en des temps très courts.

Il serait souhaitable de continuer à évaluer la mise en oeuvre des principes de [Guigon 07] sur des tâches plus complexes telles que l'exécution de mouvements corps-complet, le contrôle postural et la locomotion.

3.4 Bilan

Dans le cadre de ce travail sur la problématique du *Motion Planning*, nous avons comme finalité de traiter des problèmes liés à la conception dans des maquettes numériques en y intégrant des caractéristiques propres à l'humain.

Notre premier objectif était de proposer une méthode de planification adaptée à des environnements pouvant comporter un grand nombre de passages étroits qui sont un problème important pour toutes les approches probabilistes. Nous avons développé une méthode basée sur le principe de diffusion des *RRT* ; elle utilise la création d'arbres locaux pour accélérer l'exploration de CS tout en limitant le nombre de noeuds grâce aux contraintes de visibilité.

Il serait intéressant d'étudier plus longuement l'influence de la diffusion des arbres locaux en fonction de l'augmentation de la visibilité des arbres locaux et il faudrait tester cette approche dans le cas de robots articulés de type humanoïde pour étudier leur comportement dans des espaces hautement dimensionnés.

Un des points négatifs de notre approche de VIS-LT est qu'elle a été bâtie pour bien fonctionner dans le cadre d'environnements mixtes (suite de libre/passage étroit), ce qui peut induire des comportements moins performants dans certains types d'environnement. Par exemple, pour un environnement de type "multiples petits objets sans passage étroit", la rapidité de diffusion d'un *RRT* montre que l'ajout de composantes connexes n'est pas un bon choix. Le *RRT* est plus rapide en terme de temps de calcul car à chaque itération, il n'y a pas de tentative de connexion avec d'autres arbres et l'extension se fait uniquement pour une seule composante, ce qui évite une dispersion et permet de densifier l'arbre (quasiment chaque échantillon donne lieu à une extension), plus que dans un LT-RRT où les extensions sont réparties sur plusieurs arbres et induisent donc une densité moins forte pour chaque composante. Dans ce type d'environnement, c'est la densité de la *roadmap* qui importe le plus. Aussi, il apparaît évident qu'une approche générique doit être une combinaison de méthodes différentes.

Cette combinaison devra s'adapter à l'exploration de l'environnement en fonction des résultats des tests de collision et des extensions des arbres qui se feront en ligne. Une autre solution, réaliste dans le cadre d'une application industrielle, serait de se servir de l'expertise de l'utilisateur pour étiqueter le type d'environnement (mixte, dense, libre, tunnel,...) afin d'en sélectionner le meilleur algorithme.

Nous pouvons constater que de nombreux travaux dans différents laboratoires visent à améliorer les approches probabilistes en modifiant les fonctions de base. Même si certains résultats apportent un gain significatif, il n'en reste pas moins qu'elles souffrent toutes de leur généricité. Toutes les approches probabilistes veulent rechercher une solution en effectuant une exploration aléatoire dans un CS pour lequel nous n'avons pas de représentation explicite. Comme, en fonction du type de système robotique, de la tâche robotique et de la complexité de W, l'espace CS va être très différent, il est difficile d'imaginer une méthode générique. Il semblerait qu'une des solutions soit de pouvoir sélectionner la méthode, ou les fonctions de base, à partir d'une bibliothèque de différentes approches existantes. La difficulté est alors de pouvoir faire dynamiquement et automatiquement cette sélection en assurant la continuité dans la recherche de solution.

Une autre solution est d'utiliser l'expertise de l'utilisateur et de sa vision de l'environnement pour guider la recherche aléatoire. Nous constatons que l'information que pourrait donner un utilisateur est une composante non prise en compte dans les algorithmes de planification automatique, fort intéressante.

Cette voie n'était pas explorée dans notre laboratoire et nous nous sommes attachés à développer une approche interactive permettant une coopération entre un utilisateur et un planificateur automatique. Notre travail s'est focalisé sur la manipulation d'objet en vue d'assemblage/désassemblage de pièces dans un contexte industriel, contexte particulier où l'expertise de l'humain doit être prise en compte. Les applications sont nombreuses et elles concernent tous les secteurs manipulant des objets en phase de conception dans des maquettes numériques (automobile, aéronautique,...).

Dans tous les cas, les trajectoires calculées par un algorithme probabiliste sont une concaténation de chemins locaux construits en respectant des critères d'optimisation tels que le temps, la distance, etc. En conséquence, la trajectoire solution qui en résulte ne ressemble pas à une trajectoire que réaliserait un utilisateur qui manipulerait l'objet. Nous constatons d'ailleurs, dans ce travail, que l'opérateur se sert de la trajectoire solution automatique comme un guide mais qu'il ne la suit pas exactement.

Forts de cette constatation, nous avons décidé de nous pencher sur le mouvement naturel chez l'homme pour pouvoir améliorer la qualité des solutions dans une prospective d'intégration complète de l'utilisateur.

De manière globale, d'une part ce travail a montré la possibilité de faire exécuter des mouvements réalistes à des robots humanoïdes en s'inspirant de modèles du contrôle moteur humain, d'autre part, met en évidence l'intérêt que constituent ces plates-formes robotiques pour tester des modèles théoriques de commande moteur. Il nous semble intéressant de pouvoir tester les principes de contrôle moteur avancés par les modélisateurs sur ces systèmes artificiels construits par l'homme. Bien évidemment, notre objectif n'est pas de reproduire exactement le mouvement de l'homme, mais d'étudier le comportement du robot, en fonction de sa propre structure et de sa dynamique, en réponse à la mise en oeuvre de ces principes. Le fait de mettre en pratique ces principes permet de soulever un ensemble de questions relatives au traitement de l'information, au temps de calcul, à la robustesse des stratégies mises en oeuvre, etc. Il devient ainsi possible de faire un retour intéressant vis-à-vis des modélisateurs en neurosciences en leur suggérant des idées pour renforcer ou affiner leurs modèles ou montrer la faiblesse de certaines de leurs hypothèses. En contre-partie, nous pouvons imaginer que les principes moteurs contribuent à mettre en évidence ou suggèrent une transformation de la structure et du mode d'actionnement des robots. Par exemple, actuellement les centrales inertielles sont placées dans le torse des robots humanoïdes alors que le système vestibulaire est placé dans la tête pour un humain, il serait intéressant d'évaluer ce changement de structure comme nous avons évalué l'effet du placement du référentiel sur le comportement d'une commande.

Partie 4

Prospectives

Au départ, nous avons traité la problématique du *Motion Planning* d'un point de vue purement géométrique dans le cadre de la robotique mobile ; par la suite, nous avons étendu l'approche à la prise en compte de données capteurs jusqu'à prendre en compte l'utilisateur. A la suite des travaux effectués, il nous semble intéressant de développer un certain nombre de points liés à la problématique du *Motion Planning*.

La prise en compte de l'utilisateur doit être maintenue, mais il faut s'attacher davantage à faire un lien entre l'espace de travail dans lequel raisonne l'utilisateur et l'espace des configurations dans lequel les algorithmes effectuent la recherche. Cette réflexion doit être associée à la notion de contraintes métier que les algorithmes doivent prendre en compte lors de tâches industrielles, ainsi que nous le verrons plus loin.

Récemment, les techniques développées (en particulier les algorithmes de recherche probabilistes) trouvent des applications dans le domaine du PLM (Product Lifecycle Management) où il est possible, à partir d'une maquette numérique, de valider la conception d'une pièce mécanique avant même une première fabrication. A l'heure actuelle, les méthodes ne considèrent que des objets rigides alors que, dans bien des cas, il est nécessaire de considérer des objets déformables pour réaliser la tâche (insertion de câbles, insertion par déformation, etc). Nous souhaitons débiter de nouvelles recherches autour de la planification d'objets souples ; un thème qui n'est pas encore traité dans notre laboratoire.

Un des intérêts d'introduire des humanoïdes ou des mannequins numériques dans une maquette numérique est de tester la validité d'un processus de conception ou d'apprentissage de geste. Aussi, nous souhaitons étendre nos travaux à la *Planification de mouvement humain réaliste* en cherchant à caractériser les critères permettant d'améliorer les méthodes de planification de mouvement humain existantes pour en améliorer la réactivité.

Il paraît évident que la prise en compte de critères performants lors de la construction des chemins locaux ne peut plus être ignorée. Les méthodes de *Motion Planning* permettent de déterminer des chemins locaux faisables dans des environnements complexes en prenant en compte des contraintes géométriques et cinématiques. D'un autre côté, les méthodes d'optimisation numérique sont capables de générer des trajectoires optimales pour des bras mani-

pulateurs ou des humanoïdes en prenant en compte la dynamique du système. Il nous semble intéressant de rapprocher les techniques de ces deux communautés afin d'améliorer les algorithmes de planification.

La réflexion menée en collaboration avec des chercheurs en neurosciences pourrait déboucher sur de nouvelles idées de traitement de l'information au sein des systèmes de planification si nous pouvions caractériser des invariants exploitables dans les algorithmes de planification. Ce travail n'est pas un axe prioritaire mais constitue une réflexion engagée sur le long terme.

Les deux axes thématiques que nous comptons aborder prochainement, et que nous présentons par la suite, comportent de nombreux points communs et complémentaires. La prise en compte de la dynamique et la gestion de système hautement dimensionné se retrouvera dans les deux axes, mais de manière différente car les contraintes liées à un objet déformable sont totalement différentes de celles contraignant le mouvement d'un système antropomorphe.

4.1 Planification d'objets souples

A travers le projet *AMSI*, il est apparu que de nombreux problèmes pratiques d'assemblage ou de désassemblage d'objet ne peuvent pas être résolus en considérant uniquement des objets rigides. Par exemple, l'insertion du toit de l'habitacle d'une voiture peut s'effectuer par une insertion en force après une déformation locale, la pièce est "clipsée" en force. Si on considère le placement de câbles ou de tuyaux dans un immeuble, nous devons considérer une structure d'objet qui autorise des déformations pour pouvoir planifier les différents réseaux. Il est donc indispensable de tenir compte de la non-collision entre l'objet manipulé et l'environnement mais aussi de certaines propriétés physiques de la déformation de l'objet.

Cette problématique présente de nombreuses applications, que ce soit dans les processus de validation d'assemblage ou dans des applications de conception mécanique. Aussi, nous comptons développer un nouvel axe de recherche dans notre laboratoire sur la *Planification d'objets déformables* en nous basant sur le principe des méthodes probabilistes existantes.

Les objets déformables sont plus complexes à traiter que les objets rigides pour plusieurs raisons :

- Pour décrire le mouvement d'un objet rigide, seul le mouvement d'un point lié au solide (centre de masse par exemple) est nécessaire. Pour un objet déformable, il faut pouvoir décrire le mouvement de chaque particule de l'objet qui va se déplacer de manière différente. Nous allons donc avoir un nombre de variables plus important pour décrire le modèle de notre objet, ce qui va se traduire par une recherche dans un CS de dimension élevée.
- Un objet déformable peut provoquer des auto-collisions qui devront être détectées, en plus des collisions avec l'environnement.
- La physique des objets déformables qui décrit les propriétés physiques de la matière (élasticité, raideur,...) est plus complexe à représenter et va nécessiter des calculs plus élaborés qui devront tenir compte de la dynamique de l'objet (par exemple, implémentation des équations physiques de l'élasticité ou bien assemblage de particules).

Dans le cadre de ce travail, le but n'est pas de faire de la simulation physique de déformation

d'objet. Il existe à l'heure actuelle des méthodes efficaces de simulation mécanique, mais qui sont encore trop coûteuses en temps de calcul pour être exploitables par des algorithmes de planification. L'objectif est de trouver une solution acceptable à un problème de planification de trajectoires pour un objet déformable, réaliste en terme de temps de calcul et en terme de mouvement de l'objet. Nous sommes confrontés à une problématique semblable à l'animation graphique où le rendu en temps réel est plus important que la précision de calcul. Ici, ce n'est pas le rendu en temps réel qui importe, il s'agit de développer une algorithmique de planification de trajectoires en des temps réalistes pour être utilisable dans des bureaux d'études, tout en respectant certaines propriétés de déformation compatibles avec les lois physiques de la matière. Il nous faudra en premier lieu proposer une modélisation cohérente pour prendre en compte à la fois les propriétés mécaniques de l'objet et les contraintes des algorithmes de planification probabilistes.

Avant de présenter le projet, nous allons brièvement récapituler ce qui existe au niveau modélisation pour la simulation physique et l'exploitation déjà effectuée en *Motion Planning*.

Simulation physique

La définition d'objets déformables est trop générale car elle concerne des objets aux comportements très variés et que nous pouvons classer en corps déformables structurés (type solide déformable), corps déformables non structurés (type sable) ou fluides (type liquide).

Dans le cadre de ce travail, nous ne considérons que le cas des solides déformables structurés puisque nous nous focalisons sur le problème d'assemblage/désassemblage d'objets.

Nous trouvons dans la littérature de nombreuses modélisations de solide déformable que nous classerons en deux catégories.

- ◆ **Modèles discrets** : ils reposent sur une discrétisation de l'objet en un nombre fini de particules sur lesquelles nous appliquons les équations de la mécanique du point. Un ensemble de forces entre les particules garantit la structure de l'objet. Nous pouvons ainsi distinguer :
 - Le système masse-ressort, très utilisé en robotique [Joukhadar 98] ainsi qu'en animation graphique car il est un des plus anciens modèles, facile à comprendre et facile à implémenter. Le système, modélisé par un ensemble de masses reliées entre elles par des ressorts, permet de construire des modèles 2D ou 3D. Ce modèle simple présente des inconvénients car il est nécessaire de bien calibrer les ressorts pour obtenir un mouvement réaliste, et de limiter leur élasticité sans imposer une raideur élevée afin de ne pas rendre le système instable. De plus, le coût de calcul du mouvement peut devenir important car la complexité de calcul va dépendre du nombre de masses et de ressorts.
 - Le modèle mixte déformable-rigide [Müller 05] qui considère aussi un objet comme un ensemble de particules mais sans forces de rappel entre elles. On estime une transformation géométrique entre la position de repos et la position déformée des particules représentant cet objet qui correspond à un mouvement rigide. Ensuite, le mouvement libre des particules est combiné avec le mouvement rigide pour calculer la nouvelle position des particules définissant l'objet. Il est bien adapté au jeu vidéo par sa rapidité de calcul et sa stabilité, mais peu précis sur la qualité de la déformation.

- ◆ Modèles continus : ils se basent sur une approche globale de l'objet afin de pouvoir estimer les déformations en tout point de son volume. Comme le calcul de la solution analytique est trop complexe, on calcule une solution numérique. Soit on discrétise le volume à l'aide d'un maillage, soit on reconstruit une structure sur laquelle vont être appliquées les équations du mouvement. La seule discrétisation effectuée se situe au niveau de la résolution des équations décrivant le mouvement. Nous pouvons ainsi distinguer :
 - Les approches avec maillage qui impliquent une discrétisation de l'objet en volumes élémentaires (réguliers ou non) dans lequel les sommets correspondent à un point discrétisé et les arêtes représentent les relations de voisinage entre les points. Des fonctions d'influence associées à chaque sommet vont permettre d'évaluer la déformation continue du matériau (par exemple méthode des éléments finis, FEM).
 - Les approches sans maillage (*meshless*) qui permettent de calculer les déformations d'une particule en redéfinissant son voisinage et ses dérivées spatiales à chaque pas de temps, sans avoir besoin d'un maillage volumique. Les noeuds sont répartis sans connaître a priori les relations de connexité entre eux.

Notre objectif n'est pas de contribuer à la simulation physique de corps déformables mais cette liste non exhaustive permet de présenter les grandes familles de modèles mécaniques utilisés dans le cadre de calcul de déformation d'objet qu'il faudra prendre en compte dans le développement d'un planificateur de trajectoires. Sur un modèle mécanique, on applique des lois de comportement qui vont déplacer ou déformer l'objet avec une approximation plus ou moins grande. Nous devons nous orienter vers un modèle particulier, principalement pour des raisons de coût de calcul lors de leur utilisation dans les algorithmes de planification.

Planification de trajectoires d'objets déformables

La prise en compte d'objets déformables en planification de trajectoires n'est pas nouvelle. Dans [Lamiriaux 01] les auteurs s'intéressent à la planification d'une mince plaque élastique. La simplification d'un objet à une surface leur permet de représenter la plaque par une surface de Bézier ; les déformations de la plaque sont considérées comme des états d'énergie minimum lorsqu'on la manipule par deux côtés. On peut alors générer aléatoirement un ensemble de contraintes de manipulation qui donnent des états possibles d'équilibre de la plaque. Puis, on génère aléatoirement dans CS des configurations correspondant à ces différents états respectant les déformations. Il suffit ensuite de tester la collision pour engendrer les noeuds dans une *roadmap* de type *PRM*. La méthode locale est ici plus complexe à calculer car elle doit interpoler entre des contraintes de manipulation. Les auteurs ont étendu ces travaux au cas d'un objet volumique en utilisant une modélisation masse-ressort. Le calcul de la minimisation d'énergie est très coûteux dans cette approche, mais il permet de planifier des trajectoires dans des cas relativement simples.

Une approche de type *PRM* est aussi utilisée par [Bayazit 02]. Le principe est de générer les noeuds en considérant le robot comme un objet rigide, puis comme déformable lors du calcul du chemin. Dans une première étape, ils construisent la *roadmap* en considérant l'objet comme rigide ; ensuite ils recherchent un chemin solution et testent la validité des configurations le

long du chemin, s'il y a collision alors on déforme le robot. Pour calculer cette déformation, il ne s'agit pas de calculer une minimisation d'énergie, comme dans les travaux précédents, mais d'avoir seulement un rendu réaliste. Ils utilisent soit une déformation géométrique soit une déformation de type *bounding box*. L'idée de découpler planification de la *roadmap* et déformation est similaire à celle du Lazy-PRM dans le but de diminuer les temps de calcul. Elle est intéressante si les chemins solutions ne nécessitent pas de déformation continue le long de la trajectoire, ce qui induit de nouvelles planifications.

L'idée d'une modélisation masse-ressort est reprise dans [Gayle 05] pour calculer les déformations du robot en résolvant une minimisation d'énergie élastique. Ils construisent une *roadmap* de type *PRM* pour un robot point, planifient un chemin solution et ensuite utilisent une simulation dynamique pour calculer le mouvement du robot en vérifiant les contraintes de non-pénétration et de conservation de volume tout le long de la trajectoire.

Alors que dans ces approches, seul le robot est déformable, [Rodriguez 06] considère le cas où tout l'environnement est déformable (le robot et les obstacles). Les objets sont modélisés à l'aide de la méthode aux éléments finis. L'approche construit un arbre de type *kinodynamic RRT* [LaValle 99] en calculant à chaque itération l'ensemble des forces agissant sur le système afin de calculer le nouvel état du système.

Nous pouvons noter qu'en dehors du problème des tests de collision pour calculer les forces liées au contact, il est nécessaire de travailler dans des espaces de grande dimension car la configuration $\mathbf{q}$ correspond à la position de toutes les masses, ou point, modélisant le robot. [Mahoney 10] propose une solution en travaillant dans un espace des configurations réduit, CS_r , après une phase d'apprentissage. On décrit un ensemble discret de configurations correspondant à des déformations possibles du robot. Une analyse en composante principale (PCA) permet d'en extraire une base de dimension réduite à partir de laquelle nous pouvons calculer une transformation linéaire entre les espaces CS et CS_r . Il suffit alors de construire une *roadmap* de type *PRM* dans cet espace réduit. Mais la transformation d'espace peut créer des passages étroits inexistant dans CS et l'échantillonnage dans CS_r peut engendrer des déformations non réalistes.

Chacune des approches présente des intérêts, mais aucune ne résout de manière satisfaisante le problème d'objets déformables réguliers.

Projet

Les travaux précédents portaient sur des modèles généraux d'objets déformables structurés. Nous remarquons qu'il y a une grande différence entre un objet déformable linéaire de type câble et un objet déformable quelconque.

Un objet déformable linéaire, noté ODL, de type câble ou fil, est de dimension une car il suffit de connaître l'abscisse curviligne pour déterminer un de ses points. La déformation de cet objet doit respecter les propriétés d'élasticité, de torsion et de cisaillement. Nous pouvons considérer que son volume ne va pas trop se modifier, à la différence d'un objet déformable quelconque qui peut se rétracter ou se dilater en fonction des pressions exercées.

Par contre, lorsqu'on manipulera cet ODL, il faudra prendre en compte les points de contact anguleux qui risquent de provoquer des cisaillements du câble et des efforts de traction qui vont dépendre des frottements engendrés par la trajectoire du câble.

Cette problématique a été peu étudiée dans la communauté du *Motion Planning*. [Saha 06] décrit une méthode de planification basée sur la manipulation de câble à l'aide de deux bras manipulateurs pour réaliser des noeuds. La problématique est ici simplifiée, seule la géométrie est prise en compte. Nous pouvons citer les travaux de [Moll 06] qui définissent une paramétrisation des courbes pour définir des q d'énergie minimale. A partir des points de contact avec les obstacles, ils calculent les courbes d'énergie minimale mais le calcul de ces points est un problème difficile.

[Kabul 07] modélise un câble par un système multi-corps rigides de type robot manipulateur avec deux d.d.l. à chaque liaison (deux rotoïdes). Le câble a une longueur constante et sa configuration est décrite par l'ensemble des valeurs angulaires. Ils planifient une trajectoire pour un robot "point" en calculant une *roadmap* proche des obstacles (point de passage proche des angles) et utilisent cette trajectoire pour guider le câble. Ils calculent l'ensemble des forces agissant sur chaque corps (traction, contact, non-pénétration) pour simuler l'avancement du câble le long du chemin en utilisant un modèle dynamique direct. Un des inconvénients est de devoir considérer le câble comme un système multi-corps articulé qui implique un espace de très grande dimension dès que le câble devient long.

Cette problématique nous semble intéressante à traiter car elle comporte à la fois des problèmes géométriques plus classiques à prendre en compte et des aspect dynamiques. De plus, elle va nous obliger à travailler dans des espaces de grande dimension, ce qui est aussi une des caractéristiques des systèmes antropomorphes comme nous le verrons au 4.2.1.

De plus, cette thématique a un champ applicatif large car le passage de câble se retrouve aussi bien lors de la construction d'une maison que dans l'industrie automobile ou aéronautique, mais avec des particularités différentes. Dans le cadre d'une construction, plusieurs câbles de natures différentes seront à considérer (tuyaux d'eau, câbles électriques, ...), qui seront des entités indépendantes mais devant vérifier des contraintes entre elles. Dans un avion ou une voiture, on trouve plusieurs kilomètres de câbles qui, souvent, sont regroupés en faisceaux. Ces faisceaux vont nécessiter des points d'accroche particuliers qu'il est nécessaire de prendre en compte. Il sera question de traiter cette problématique en y introduisant les contraintes métier propres aux champs applicatifs.

Dans le cadre de ce projet de recherche, nous allons travailler incrémentalement en considérant plusieurs étapes :

- En premier lieu, nous prendrons un seul câble avec un modèle de déformation simplifié. Ici, la planification de l'ODL peut se ramener à étudier la trajectoire d'une boule devant respecter des contraintes de courbure réalistes par rapport au modèle mécanique. Une des difficultés sera de contrôler le mouvement de cette boule pour tenir compte de la progression du câble.
- Ensuite, il faudra étendre la problématique à un ensemble de câbles devant respecter le même chemin (même classe de chemins homotopes). Ici, il est illusoire de vouloir traiter un ensemble de câbles comme un système multi-robots car la dimension de l'espace de recherche deviendrait trop grande. Il faudra donc adapter notre algorithmique sans la dupliquer à chaque

câble. Les contraintes entre câbles devraient nous guider sur le choix de solution.

- Enfin, il est souhaitable que l'algorithme de planification prenne en compte les contraintes liées aux faisceaux de câbles de type point d'accroche. On peut voir ces points comme des "points de passage" et il convient de considérer le cas où ils sont donnés par l'utilisateur au vu de sa connaissance métier, ce qui peut impliquer une mauvaise définition, et le cas où le planificateur de trajectoire calcule ces points.

Il sera nécessaire de bien maîtriser le comportement des algorithmes de calcul de collision car la gestion du contact sera centrale dans ce projet. Afin de réduire les temps de calcul, qui seront importants, il faudra acquérir une expérience dans les techniques de programmation par GP-GPU afin d'en évaluer la pertinence dans ce contexte.

Pour concrétiser ce travail, nous avons convenu de monter un projet de recherche avec le laboratoire LIST du CEA et la société KineoCam comme industriel toulousain. Le LIST-CEA (resp : C. Andriot) est spécialisé dans la modélisation et la simulation physique, il apportera ses compétences dans la modélisation de la déformation de câble tandis que KineoCam (Resp : E. Ferré) sera l'intégrateur des technologies logicielles de planification de trajectoire dont les prototypes seront développés au LAAS-CNRS (resp : M. Taïx). Un utilisateur final de l'industrie automobile ou aéronautique sera associé au projet afin d'en tester la validité.

4.2 Prise en compte de l'humain

Dans le domaine en pleine expansion de la conception ou de l'apprentissage de gestes à partir de maquette numérique, un enjeu réside dans l'introduction de mannequins numériques et de leur simulation lors de la manipulation de l'objet. Aussi, nous souhaitons étendre nos travaux à la *Planification de mouvements humains réalistes et leurs applications* en cherchant à caractériser les critères permettant d'améliorer les méthodes de planification de mouvement humain existantes pour en améliorer la réactivité. Les méthodes de planification et de commande de systèmes anthropomorphes qui sont développés dans le groupe Gepetto reposent essentiellement sur la géométrie et la cinématique. Ces méthodes s'avèrent insuffisantes pour résoudre ce problème et il nous sera nécessaire de prendre en compte la dynamique du système au sein de ces méthodes. Ce travail s'articulera autour des deux points présentés ci-dessous.

4.2.1 Planification de mouvements réalistes

Vouloir planifier des trajectoires réalistes pour des robots humanoïdes [Kuffner 01, Chestnutt 05, Candido 08, Dalibard 09b] est plus difficile que dans le cas d'un robot mobile. Un système humanoïde possède un grand nombre de d.d.l qui oblige à explorer des espaces de grande dimension et implique une grande redondance du système par rapport à la tâche, redondance qu'il faudra gérer pour obtenir une trajectoire réaliste.

Par exemple, si on veut planifier la trajectoire d'un personnage de type humanoïde ou avatar, nous pouvons planifier une trajectoire avec une approche probabiliste classique type *RRT* en imposant des contraintes de pieds au sol et de stabilité statique. Si on utilise une méthode lo-

cale d'interpolation linéaire entre les noeuds du graphe, la trajectoire ne ressemblera pas à une trajectoire de marche d'un être humain.

Une des approches possibles est d'utiliser des *pattern* de marche acquis par des techniques de capture de mouvement. A partir d'une trajectoire planifiée solution, il suffit de plaquer les *pattern* en utilisant un contrôleur de locomotion. Des travaux récents [Truong 10] montrent qu'on peut utiliser des contrôleurs de locomotion qui respectent bien le caractère holonome et non-holonome de la marche humaine, qui ont des invariants différents. La particularité de ce type de mouvement est que la direction de déplacement du personnage n'est pas toujours tangente à la direction de la trajectoire géométrique au sol, par exemple un personnage peut faire un pas sur le côté.

Nous pouvons appliquer cette caractéristique à un robot humanoïde en calculant le placement des pieds qui respecte cette caractéristique. Un des tout premiers résultats est indiqué sur la figure 4.1 où le robot doit faire des pas sur le côté pour passer entre deux obstacles.

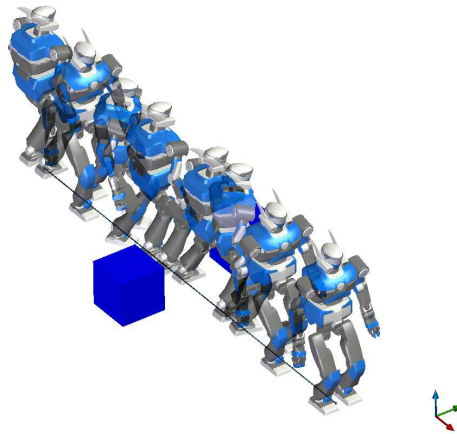


FIG. 4.1 – Marche d'un humanoïde respectant des contraintes de type marche humaine

Dans cette approche, la planification est scindée en deux parties, nous calculons une trajectoire géométrique sans collision respectant les contraintes réalistes, puis la marche vient se plaquer sur cette solution. Il est évident qu'il serait souhaitable de calculer directement des mouvements réalistes lors de la recherche de la trajectoire solution et cela non seulement pour des tâches de locomotion, mais aussi pour des tâches nécessitant des mouvements corps-complet.

Comme nous considérerons des robots de type humanoïde possédant un grand nombre de degrés de liberté, nous utiliserons une approche de type *RRT* pour le calcul de solution. La difficulté va être d'étendre le noeud de la *roadmap* à l'aide d'un chemin local prenant en compte les contraintes associées à ce type de système (stabilité,...) en des temps réalistes. Jusqu'ici les méthodes de planification et de commande de systèmes anthropomorphes que nous avons développés dans le groupe Gepetto reposent essentiellement sur la cinématique. Nous souhaitons les étendre à la prise en compte de la dynamique. Le temps de calcul important qu'un tel modèle implique sera une difficulté à résoudre. Il faut garder en tête que les méthodes pro-

babibilistes calculent un grand nombre de chemin locaux et que par conséquent cette étape doit être très rapide si nous voulons aller vers une réelle prise en compte de la dynamique. De plus, vouloir planifier des trajectoires pour des systèmes humanoïdes corps-complet implique la vérification d'un ensemble de contraintes à chaque configuration du chemin solution. L'espace des configurations admissibles devient alors une sous-variété de CS. Définir le chemin local comme une interpolation linéaire des paramètres de configuration comme cela est fait en planification probabiliste, n'est pas toujours possible.

Nous envisageons pour cela d'utiliser des techniques propres à la communauté de l'optimisation numérique afin de pouvoir calculer des trajectoires locales intégrant la dynamique dans des environnements complexes. Un des intérêts de cette approche est que l'utilisation de techniques de *Motion Planning* peut résoudre des problèmes de minimum locaux liés aux méthodes numériques. Ce travail est amorcé avec la thèse d'A. El Khoury.

A plus long terme, il sera intéressant d'étudier la prise en compte de caractéristiques ergonomiques dans le planificateur local de trajectoire car la solution ne sera pas unique, vu le degré de redondance du système par rapport à la tâche locale. Il faudra alors pouvoir considérer les efforts musculaires qui engendrent le déplacement du corps (bras, torse, ...). L'université de Tokyo (Pr Nakamura) a développé un modèle du système musculo-squelettique humain et un logiciel de simulation *sDIMS* qui permet de simuler l'activité du système musculo-squelettique à partir de données de capture de mouvement, de mesures EMG, et d'un retour d'effort. Il faudra alors trouver une solution pour intégrer une telle information dans nos algorithmes de planification.

4.2.2 Planification et contrôle moteur humain

La réflexion menée en collaboration avec des chercheurs en neurosciences est enrichissante et pourrait déboucher sur de nouvelles idées de traitement de l'information au sein des systèmes de planification.

Cette communauté tente de comprendre et décrire le fonctionnement du contrôle moteur humain. Pour répondre à cette question, les chercheurs considèrent à la fois le problème de la perception et de la commande. Pour déplacer la main vers une cible, le système nerveux central doit effectuer un codage de l'espace et de l'état interne du corps. Quels sont les mécanismes d'intégration multi-sensorielle qui sont à la base de cette construction ? Quels référentiels et quelles transformations sont utilisés pour mettre en correspondance les différentes données ? Selon quelle architecture est organisé le processus de commande ? Autant de questions voisines des préoccupations du roboticien. Chez l'homme, on sait qu'il existe différents niveaux de contrôle, allant du réflexe simple à l'élaboration de plans moteur complexes. Quels sont les processus qui permettent de relier la planification de haut niveau au contrôle du mouvement ? A cause du nombre important de d.d.l. de sa structure et du nombre important de muscles qui permettent son actionnement, le corps humain est fortement redondant par rapport à une tâche. Comment le SNC procède-t-il pour sélectionner un patron d'activation parmi une infinité de choix possibles ? Cette question, initialement formulée par [Bernstein 67], est centrale dans le travail des modélisateurs depuis le milieu des années 60. Une réponse à cette question pourrait être très utile pour savoir comment planifier des chemins locaux qui présentent des

caractéristiques similaires à l'homme. Le débat au sein de cette communauté est encore très ouvert et aucune de ces approches ne fait l'unanimité. Il apparaît que les modélisateurs sont de plus en plus intéressés par la vision du roboticien qui conçoit et met en oeuvre des machines physiques ayant une autonomie de perception et de commande. La robotique humanoïde a renforcé cet intérêt, puisqu'elle fait surgir des questions nouvelles, telles le contrôle d'une structure fortement redondante.

Nous avons récemment appliqué avec succès le modèle biologique de contrôle moteur développé par Emmanuel Guigon à la commande du geste d'atteinte du robot HRP2 []. Ce travail montre l'intérêt que constituent les plates-formes robotiques pour tester des modèles théoriques de commande de mouvement. Nous avons bon espoir qu'à long terme cette collaboration donnera de nouvelles pistes exploitables dans les futurs algorithmes.

Références

Références

- [Alami 00] R. Alami, R. Chatila, S. Fleury, M. Herrb, F. Ingrand, M. Khatib, B. Morisset, P. Moutarlier & T. Siméon. *Around the Lab in 40 Days...* In IEEE International Conference on Robotics and Automation, pages 88–94, San Francisco, CA, April 2000.
- [Amato 98] N. M. Amato, O. B. Bayazit, L. K. Dale, C. Jones & D. Vallejo. *OBPRM : An Obstacle-Based PRM for 3D Workspaces*. In Workshop on Algorithmic Foundations of Robotics, 1998.
- [Arkin 00] Esther M. Arkin, Michael A. Bender, Erik D. Demaine, Sandor P. Fekete, Joseph S.B. Mitchell & Saurabh Sethia. *Optimal Covering Tours with Turn Costs*, 2000.
- [Atkar 05] P. Atkar, A. Greenfield, C. Conner, H. Choset & A. Rizzi. *Uniform Coverage of Automotive Surface Patches*. The International Journal of Robotics Research, vol. 24, no. 1, pages 883 – 898, November 2005.
- [Avnaim 89] F. Avnaim & J.-D. Boissonnat. *Polygon placement under translation and rotation*. Informatique Theorique et Applications/Theoretical Informatics and Applications, vol. 23, 1989.
- [Barraquand 89] J. Barraquand & J.C. Latombe. *On non-holonomic mobile robots and optimal maneuvering*. Revue d’Intelligence Artificielle, vol. 3, no. 2, pages 77–103, 1989.
- [Barraquand 92] J. Barraquand & J.C. Latombe. *Numerical potential field techniques for robot path planning*. IEEE Transaction on System, Man and Cybernetics, vol. 22, no. 2, 1992.
- [Barraquand 93] J. Barraquand & J.C. Latombe. *Nonholonomic multibody mobile robots : controllability and motion planning in presence of obstacles*. Algorithmica, no. 10, pages 121–155, 1993.
- [Bayazit 00] O. B. Bayazit, G. Song & N. M. Amato. *Enhancing Randomized Motion Planners : Exploring with Haptic Hints*. In IEEE International Conference on Robotics and Automation, pages 529–536, april 2000.
- [Bayazit 02] B. Bayazit, J-M. Lien & N. Amato. *Probabilistic Roadmap Motion Planning for Deformable Objects*. In IEEE International Conference on Robotics and Automation, volume 2, pages 2126 – 2133, 2002.

- [Ben Amar 93] F. Ben Amar, Ph. Bidaud & F. Ben Ouezdou. *On modelling and motion planning of planetary vehicles*. In IEEE International Conference On Intelligent Robots and Systems, Yokohama (Japan), pages 1381–1386, 1993.
- [Ben Amar 95] F. Ben Amar & Ph. Bidaud. *Dynamic analysis of off-road vehicles*. In Proc. International Symposium on Experimental Robotics, pages 363–371, 1995.
- [Bernstein 67] Bernstein, editeur. *The coordination and regulation of movements*. Oxford, Pergamon Press,, 1967.
- [Boor 99] V. Boor, M. H. Overmars & A. F. van der Stappen. *The Gaussian Sampling Strategy for Probabilistic Roadmap Planners*. In IEEE International Conference on Robotics and Automation, pages 1018–1023, 1999.
- [Burns 05] B. Burns & O. Brock. *Toward Optimal Configuration Space Sampling*. In Proceedings of Robotics : Science and Systems, 2005.
- [Burns 07] B. Burns & O. Brock. *Single-Query Motion Planning with Utility-Guided Random Trees*. In IEEE International Conference on Robotics and Automation, 2007.
- [Candido 08] S. Candido, Yong-Tae Kim & S. Hutchinson. *An improved hierarchical motion planner for humanoid robots*. In IEEE-RAS International Conference on Humanoid Robots, pages 654 –661, 2008.
- [Canny 88] J. F. Canny. *The complexity of robot motion planning*. MIT Press, Cambridge, MA, 1988.
- [Chaumette 06] F. Chaumette & S. Hutchinson. *Visual servo control. I. Basic approaches*. Robotics Automation Magazine, IEEE, vol. 13, no. 4, pages 82 –90, 2006.
- [Chen 05] L. Chen & C. G. Brown. *A 3D Mouse for Interacting with Virtual Objects*. In IEEE International Symposium on Circuits and Systems, 2005.
- [Cherif 94] M. Cherif, Ch. Laugier, Ch. Milési-Bellier & B. Faverjon. *Planning the motions of an all-terrain vehicle by using geometric and physical models*. In Proc. IEEE International Conference on Robotics and Automation, San Diego (USA), pages 2050–2056, 1994.
- [Cherif 95] M. Cherif & Ch. Laugier. *On physically-based modeling and off-road vehicle motion planning*. In Proc. International Symposium on Experimental Robotics, pages 482–489, 1995.
- [Chestnutt 05] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins & T. Kande. *Footstep Planning for the Honda ASIMO Humanoid*. In IEEE International Conference on Robotics and Automation, pages 629 – 634, 2005.
- [Choi 09] Y.-H. Choi, T.-K. Lee, S.-H. Baek & S.-Y. Oh. *Online complete coverage path planning for mobile robots based on linked spiral paths using constrained inverse distance transform*. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2009.
- [Choset 01] Howie Choset. *Coverage for robotics - A survey of recent results*. Annals of Mathematics and Artificial Intelligence, vol. 31, pages 113 – 126, 2001.

- [Choset 05] Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A Kantor, Wolfram Burgard, Lydia E. Kavraki & Sebastian Thrun. *Principles of robot motion : Theory, algorithms, and implementations*. MIT Press, 2005.
- [CMU-NASA 06] CMU-NASA. *Life in the Atacama*. In <http://www.frc.ri.cmu.edu/atacama/>, 2003-2006.
- [Dalibard 09a] S. Dalibard & J.P. Laumond. *Control of probabilistic diffusion in motion planning*. Algorithmic Foundation of Robotics VIII, pages 467–481, 2009.
- [Dalibard 09b] S. Dalibard, A. Nakhaei, F. Lamiriaux & J.-P. Laumond. *Whole-body task planning for a humanoid robot : a way to integrate collision avoidance*. In IEEE-RAS International Conference on Humanoid Robots, pages 355–360, 2009.
- [Deng 95] Xiaotie Deng, Evangelos Milios & Andy Mirzaian. *Landmark Selection Strategies for Path Execution*. Robotics and Autonomous Systems, vol. 17, pages 17–3, 1995.
- [der Helm 00] F.C.T. Van der Helm & L.A. Rozendaal. *Musculoskeletal systems with intrinsic and proprioceptive feedback*. Neural control of posture and movement, Springer Verlag, NY, 2000.
- [Espiau 92] B. Espiau, F. Chaumette & P. Rives. *A New Approach to Visual Servoing in Robotics*. IEEE Transactions on Robotics and Automation, vol. 8, no. 3, pages 313–326, 1992.
- [Ferré 04] E. Ferré & J.P. Laumond. *An iterative diffusion algorithm for part disassembly*. In IEEE International Conference on Robotics and Automation, pages 3149–3154, april 2004.
- [Flavigné 10] D. Flavigné. *Planification de mouvement interactive : coopération humain-machine pour la recherche de trajectoire et l’animation*. PhD thesis, LAAS-CNRS, Université Paul Sabatier, Septembre 2010.
- [Fliess 95] M. Fliess, J. Lévine, P. Martin & P. Rouchon. *Flatness and defect of non-linear systems : introductory theory and examples*. International Journal of Control, vol. 61, no. 6, pages 1327 – 1361, 1995.
- [Fujita 04] Y. Fujita, Y. Nakamura, K. Yamane & Suzuki I. *Somatosensory Computation for Man–Machine Interface From Motion-Capture Data and Musculoskeletal Human Model*. IEEE Trans. On Robotics 21(1) :58-66, 2004.
- [Gabriely 02] Y. Gabriely & E. Rimon. *Spiral-STC : an on-line coverage algorithm of grid environments by a mobile robot*. In IEEE International Conference on Robotics and Automation, pages 954 – 960, 2002.
- [Gaw 86] D. Gaw & A. Meystel. *Minimum-time navigation of an unmanned mobile robot in a 2-1/2 world with obstacles*. In Proc. IEEE International Conference on Robotics and Automation, San Francisco (USA), pages 1670–1677, 1986.

- [Gayle 05] R. Gayle, M.C. Lin & D. Manocha. *Constraint-Based Motion Planning of Deformable Robots*. In IEEE International Conference on Robotics and Automation, pages 1046 – 1053, 2005.
- [Guigon 07] E. Guigon, P. Baraduc & M. Desmurget. *Computational motor control : Redundancy and invariance*. J Neurophysiol, vol. 97, no. 1, pages 331–347, 2007.
- [Hait 98] A. Hait. *Algorithmes pour la planification de trajectoires robustes d'un robot mobile autonome sur un terrain accidenté*. PhD thesis, LAAS-CNRS, Université Paul Sabatier, Juillet 1998.
- [Haït 96] A. Haït & T. Siméon. *Motion planning on rough terrain for an articulated vehicle in presence of uncertainties*. In IEEE International Conference On Intelligent Robots and Systems, Osaka (Japan), pages 1670–1677, 1996.
- [He 08] Ruijie He, S. Prentice & N. Roy. *Planning in information space for a quadrotor helicopter in a GPS-denied environment*. In IEEE International Conference on Robotics and Automation, pages 1814 –1820, May 2008.
- [Hersch 06] M. Hersch & Billard A.G. *A Biologically-Inspired Controller for Reaching Movements*. IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006.
- [Hsu 05] D. Hsu, G. Sánchez-Ante & Z. Sun. *Hybrid PRM sampling with a cost-sensitive adaptive strategy*. In IEEE International Conference on Robotics and Automation, 2005.
- [Hu 10] Guoqing Hu, Zhengwei Hu & Hongbo Wang. *Complete coverage path planning for road cleaning robot*. In International Conference on Networking, Sensing and Control, pages 643 –648, 10-12 2010.
- [Huang 01] W.H. Huang. *Optimal line-sweep-based decompositions for coverage algorithms*. In IEEE International Conference on Robotics and Automation, pages 27 – 32 vol.1, 2001.
- [J.-P. Laumond 98] S. Sekhavat J.-P. Laumond & F. Lamiroux. *Guidelines in nonholonomic motion planning for mobile robots*. Springer, 1998.
- [Jensen 02] B. Jensen, G. Froidevaux, X. Greppin, A. Lorotte, L. Mayor, M. Meisser, G. Ramel & R. Siegwart. *The interactive autonomous mobile system RoboX*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1221–1227, Lausanne, Switzerland, Oct 2002.
- [Joukhadar 98] A. Joukhadar, A. Deguet & C. Laugier. *A collision model for rigid and deformable bodies*. In IEEE International Conference on Robotics and Automation, volume 2, pages 982 –988 vol.2, May 1998.
- [JPL-NASA 06] JPL-NASA. *Mars Exploration Rover Mission*. In [http ://marsrovers.nasa.gov](http://marsrovers.nasa.gov), 2003-2006.

- [Kabul 07] Ilknur Kabul, Russell Gayle & Ming C. Lin. *Cable Route Planning in Complex Environments Using Constrained Sampling*. In ACM Symposium on Solid and Physical Modeling (SPM), 2007.
- [Katz 06] R. Katz & S. Hutchinson. *Efficiently biasing PRMs with passage potentials*. In IEEE International Conference on Robotics and Automation, pages 889–894, 2006.
- [Kavraki 96] L. E. Kavraki, P. Svestka, J.-C. Latombe & M. H. Overmars. *Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces*. IEEE Transactions on Robotics & Automation, vol. 12, no. 4, pages 566–580, June 1996.
- [Kuffner 01] Jr. Kuffner J.J., K. Nishiwaki, S. Kagami, M. Inaba & H. Inoue. *Footstep planning among obstacles for biped robots*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 1, pages 500–505 vol.1, 2001.
- [Ladezeve 10] N. Ladezeve. *Apports des méthodes de planification automatique dans des simulations interactives d’industrialisation et de maintenance en réalité virtuelle*. PhD thesis, Equipe du LGP de l’ENIT, INPT-Toulouse, Avril 2010.
- [Lamiriaux 01] F. Lamiriaux & L. Kavraki. *Planning paths for elastic objects under manipulation constraints*. Int. Journal of Robotics Research, vol. 20, no. 3, 2001.
- [Large 00] F. Large, S. Sekhavat, C. Laugier & E. Gauthier. *Towards Robust Sensor-Based Maneuvers for a Car-Like Vehicle*. In IEEE International Conference on Robotics and Automation, pages 3765 – 3769, 2000.
- [Latombe 91] J.C. Latombe. Robot motion planning. Kluwer Academic Publishers, 1991.
- [Laugier 98] C. Laugier, T. Fraichard, I.E. Paromtchik & P. Garnier. *Sensor-based control architecture for a car-like vehicle*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 1, pages 216–222, October 1998.
- [Laumond 86] J.P. Laumond. *Feasible trajectories for mobile robots with kinematic and environment constraints*. In International Conference on Intelligent Autonomous Systems, Amsterdam, 1986.
- [LaValle 99] S. M. LaValle & J. J. Kuffner. *Randomized Kinodynamic Planning*. In IEEE International Conference on Robotics and Automation, pages 473–479, 1999.
- [LaValle 00] S. M. LaValle & J. J. Kuffner. *Rapidly-Exploring Random Trees : Progress and Prospects*. In Proceedings Workshop on the Algorithmic Foundations of Robotics, 2000.
- [LaValle 06] S. M. LaValle. Planning algorithms. Cambridge University Press, Cambridge, U.K., 2006. Available at <http://planning.cs.uiuc.edu/>.

- [Lecuyer 04] A. Lecuyer, M. Vidal, O. Joly, C. Megard & A Berthoz. *Can haptic feedback improve the perception of self-motion in virtual reality ?* In Int. Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pages 208–215, 2004.
- [Leven 99] P. Leven, S. Hutchinson, D. Burschka & G. Färber. *Perception-Based Motion Planning for Indoor Exploration*. In IEEE International Conference on Robotics and Automation, pages 695 – 701, 1999.
- [Liegeois 93] A. Liegeois & C. Moignard. *Optimal motion planning of a mobile robot on a triangulated terrain model*. Lecture Notes in Computer Science, vol. 708, pages 51–65, 1993.
- [Lozano-Pérez 83] T. Lozano-Pérez. *Spatial Planning : A Configuration Space Approach*. IEEE Transactions on Computing, vol. C-32, no. 2, pages 108–120, 1983.
- [Luca 98] A. De Luca, G. Oriolo & C. Samson. Feedback control of a nonholonomic car-like robot. Springer, 1998.
- [Lumelsky 87] V. J. Lumelsky & A. A. Stepanov. *Path Planning Strategies for a Point Mobile Automaton Moving Amidst Unknown Obstacles of Arbitrary Shape*. Algorithmica, vol. 2, pages 403–430, 1987.
- [M. 00] Mataric M. *Getting Humanoids to Move and Imitate*. In IEEE Intelligent Systems, 2000.
- [Mahoney 10] A. Mahoney, J. Bross & D. Johnson. *Deformable robot motion planning in a reduced-dimension configuration space*. In IEEE International Conference on Robotics and Automation, pages 5133 –5138, 2010.
- [Malti 05] A. C ; Malti. *Planification et exécution de mouvements référencés sur des amers*. PhD thesis, LAAS-CNRS, Université Paul Sabatier, Décembre 2005.
- [Mannadiar 10] R. Mannadiar & I. Rekleitis. *Optimal coverage of a known arbitrary environment*. In IEEE International Conference on Robotics and Automation, pages 5525 –5530, 3-7 2010.
- [Matsui 06] T. Matsui, M. Honda & N. Nakazawa. *A new optimal control model for reproducing human arm's two-point reaching movements : A modified minimum torque change model*. In IEEE International Conference on Robotics and Biomimetics, pp 1541-1546, 2006.
- [Milési-Bellier 93] Ch. Milési-Bellier, Ch. Laugier & B. Faverjon. *A kinematic simulator for motion planning of a mobile robot on a terrain*. In IEEE International Conference On Intelligent Robots and Systems, Yokohama (Japan), pages 339–344, 1993.
- [Moll 06] M. Moll & L.E. Kavraki. *Path planning for deformable linear objects*. IEEE Transactions on Robotics, vol. 22, no. 4, pages 625–636, 2006.

- [Müller 05] Matthias Müller, Bruno Heidelberger, Matthias Teschner & Markus Gross. *Meshless deformations based on shape matching*. In ACM SIGGRAPH 05, 2005.
- [Northrup 01] S. Northrup, N. Sarkar & K. Kawamura. *Biologically-inspired control architecture for a humanoid robot*. In Proc IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Vol 2, pp 1100-1105, 2001.
- [Oksanen 09] T. Oksanen & A. Visala. *Coverage path planning algorithms for agricultural field machines*. Journal of Field Robotics, vol. 26(8), pages 651–668, 2009.
- [Park 07] J. Park, B. Lim, J. Haan & F. Park. *Optimal reaching and balancing for humanoid robots*. In Proc Robotics, Science and Systems Conference, Workshop on Robot Manipulation, 2007.
- [Reeds 90] J.A. Reeds & L.A. Shepp. *Optimal paths for a car that goes both forwards and backwards*. Pacific J. Math, vol. 145, pages 367–393, 1990.
- [Rodriguez 06] Samuel Rodriguez, Jyh-Ming Lien & N. M. Amato. *Planning Motion in Completely Deformable Environments*. In IEEE International Conference on Robotics and Automation, Orlando, FL, 2006.
- [Rosell 08] J. Rosell, C. Vazquez, A. Perez & P. Iniguez. *Motion Planning for Haptic Guidance*. Journal of Intelligent Robotics Systems, vol. 53, no. 3, pages 223–245, 2008.
- [Rouchon 94] P. Rouchon. *Necessary Condition and Genericity of Dynamic Feedback linearization*. Journal of Mathematical Systems, Estimation, and Control, vol. 4, no. 2, 1994.
- [Saha 06] M. Saha & P. Ito. *Motion planning for robotic manipulation of deformable linear objects*. In IEEE International Conference on Robotics and Automation, pages 2478 –2484, May 2006.
- [Sala 06] Pablo Sala, Robert Sim, Ali Shokouf & Sven Dickinson. *Landmark Selection for Vision-Based Navigation*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3131–3138, 2006.
- [Samson 91] C. Samson, B. Espiau & M. Le Borgne. *Robot control the task function approach*. Oxford science publications, 1991.
- [Samson 95] C. Samson. *Control of Chained Systems. Application to Path Following and Time-Varying Point-Stabilization*. IEEE Transaction on Automatic Control, vol. 40, no. 1, 1995.
- [Sha 06] D.H. Sha, J. Patton & F.A. Mussa-Ivaldi. *Minimum jerk reaching movements of human arm with mechanical constraints at endpoint*. Int J Comput Syst Signals, vol. 7, no. 1, pages 41–50, 2006.
- [Shadmehr 05] R. Shadmehr & S.P. Wise, editors. *Computational neurobiology of reaching and pointing, a foundation for motor learning*. MIT Press, 2005.

- [Shawna Thomas 07] Xinyu Tang Shawna Thomas Marco Morales & Nancy M. Amato. *Biasing Samplers to Improve Motion Planning Performance*. In IEEE International Conference on Robotics and Automation, Kobe, Japan, may 2007.
- [Shiller 90] Z. Shiller & J.C. Chen. *Optimal motion planning of autonomous vehicles in three dimensional terrains*. In Proc. IEEE International Conference on Robotics and Automation, Cincinnati (USA), pages 198–203, 1990.
- [Siméon 00] T. Siméon, J.-P. Laumond & C. Nissoux. *Visibility based probabilistic roadmaps for motion planning*. Advanced Robotics, vol. 14, no. 6, 2000.
- [Simmons 05] G. Simmons & Y. Demiris. *Optimal robot arm control using the minimum variance model*. J Robot Syst, vol. 22, no. 11, pages 677–690, 2005.
- [Siméon 91] T. Siméon. *Motion planning for a non-holonomic mobile robot on 3-dimensional terrains*. In IEEE International Workshop On Intelligent Robots and Systems, Osaka (Japan), pages 1455–1460, 1991.
- [Siméon 93] T. Siméon & B Dacre-Wright. *A practical motion planner for all-terrain mobile robots*. In IEEE International Conference On Intelligent Robots and Systems, Yokohama (Japan), pages 1357–1363, 1993.
- [Soueres 94] P. Soueres, J.-Y. Fourquet & J.-P. Laumond. *Set of reachable positions for a car*. IEEE Transactions on Automatic Control, vol. 39, no. 8, pages 1626–1630, 1994.
- [Souères 98] P. Souères & J. D. Boissonnat. *Optimal trajectories for nonholonomic mobile robots*. Springer, 1998.
- [Sreng 06] J. Sreng, A. Lecuyer, C. Megard & C. Andriot. *Using Visual Cues of Contact to Improve Interactive Manipulation of Virtual Objects in Industrial Assembly/Maintenance Simulations*. IEEE Transactions on Visualization and Computer Graphics, vol. 12, no. 5, pages 1013–1020, 2006.
- [Strandberg 04] M. Strandberg. *Augmenting RRT-planners with local trees*. In IEEE International Conference on Robotics and Automation, pages 3258–3262, 2004.
- [Su 08] Y. Su, W. Zhu & T. Yu. *Virtual Assembly Platform Based on PC*. In International Conference on Audio, Language and Image, 2008.
- [Sun 05] Z. Sun, D. Hsu, T. Jiang, H. Kurniawati & J. H. Reif. *Narrow passage sampling for probabilistic roadmap planning*. IEEE Transactions on Robotics, vol. 21, no. 6, pages 1105–1115, 2005.
- [Svestka 95] P. Svestka & M.H. Overmars. *Coordinated motion planning for multiple car-like robots using probabilistic roadmaps*. In ICRA, pages 1631–1636, may 1995.
- [Swain-Oropeza 01] R. Swain-Oropeza, M. Devy & S. Hutchinson. *Sensor-based navigation in cluttered environments*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 3, pages 1662–1669, 2001.

- [Thrun 99] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte & D. Schulz. *MINERVA : a second-generation museum tour-guide robot*. In IEEE International Conference on Robotics and Automation, pages 1999–2005, Detroit, MC, May 1999.
- [Tran 09] M. T. Tran. *Approche neuro-robotique pour le contrôle des systèmes anthropomorphiques*. PhD thesis, LAAS-CNRS, Université Paul Sabatier, Novembre 2009.
- [Truong 10] Tan-Viet-Anh Truong, David Flavigne, Julien Pettre, Katja Mombaur & Jean-Paul Laumond. *Reactive synthesizing of human locomotion combining nonholonomic and holonomic behaviors*. In IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob), pages 632 –637, 2010.
- [Wettergreen 05] D. Wettergreen, N. Cabrol, V. Baskaran, F. Calderon, S. Hueys, D. Jona, A. Luders, D. Pane, T. Smith, J. Teza, P. Tompkins, D. Villa, C. Williams & M. Wagner. *Second experiments in the robotic investigation of life in the Atacama desert of Chile*. In 8th International Symposium on Artificial Intelligence, Robotics and automation in Space, 2005.
- [Wächter 06] A. Wächter & T. L. T. Biegler. *On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming*. Mathematical Programming, vol. 106, no. 1, pages 25,57, 2006.
- [Yao 06] Zhiyang Yao. *Finding Efficient Robot Path for the Complete Coverage of A Known Space*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3369 –3374, 9-15 2006.
- [Yershova 05] A. Yershova, L. Jaillet, T. Siméon & S. M. LaValle. *Dynamic-Domain RRTs : Efficient Exploration by Controlling the Sampling Domain*. In IEEE International Conference on Robotics and Automation, 2005.
- [Zucker 10] M. Zucker, J.A. Bagnell, C.G. Atkeson & J. Kuffner. *An optimization approach to rough terrain locomotion*. In IEEE International Conference on Robotics and Automation, pages 3589 –3595, 2010.

Partie 5

Publications

Tableau récapitulatif

Thèse de doctorat

1. “*Planification de mouvement pour robot mobile non-holonyme*”
Thèse de l’Université Paul Sabatier, spécialité robotique, Toulouse, janvier 1991.

Revues internationales

2. “*A General Framework for Planning Landmark-Based Motions for Mobile Robots*”
A. Malti, M. Taïx et F. Lamiroux dans *Advanced Robotics*, vol. 25, no.11, pp. 1427-1450, juillet 2011.
3. “*Algorithms for rough terrain trajectory planning*”
A. Haït, T. Siméon et M. Taïx, dans *Advanced Robotics*, vol 16, no 8, décembre 2002.
4. “*A Motion Planner for Nonholonomic Mobile Robots*”
J.P. Laumond, P. Jacobs, M. Taïx et R. Murray, dans *IEEE Transaction on Robotics & Automation*, vol 10, no 5, pp. 577-593, octobre 1994.

Contributions à collection

5. “*Planning robust landmarks for sensor based motion*”,
M. Taïx, A. Malti, F. Lamiroux, dans *Springer Tracts in Advanced Robotics*, vol 44, pages 195-204, Springer, 2008.
6. “*Polyhedron placement for planar motions*”
T. Siméon, M. Taïx, P. Juhel et J.P. Laumond, dans *Advances in Factories of the future, CIM and Robotics*, Elsevier Science Publishers, pages 407-417, 1993.

Communications et Conférences avec actes

7. “*Improving Motion Planning in Weakly Connected Environments*”
D. Flavigné et M. Taïx. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taiwan, octobre 2010.

8. “*Humanoid human-like reaching control based on movement primitive*”
Minh Tuan Tran, Philippe Soueres, Michel Taïx, Manish N. Sreenivasa, Christophe Haldand. *IEEE International Symposium on Robot and Human Interactive Communication*, Italie, septembre 2010.
9. “*Méthode hybride de planification de trajectoires pour environnement mixte*”
D. Flavigné et M. Taïx. *Congrès Francophone de Reconnaissance des Formes et l’Intelligence Artificielle (RFIA)*, Caen, 19-22 janvier 2010.
10. “*Eye-centered vs body-centered reaching control : a computational approach to the debate*”¹
Minh Tuan Tran, Philippe Soueres, Michel Taïx et Benoit Girard. *IEEE Inter. Conf. on Robotics and Biomimetics (ROBIO)*, Guilin (Chine), 19-23 Décembre 2009, pp.568-573.
11. “*Interactive Motion Planning for Assembly Tasks*”
D. Flavigné, M. Taïx et E. Ferré. *International Symposium on Robot and Human Interactive Communication Toyama* (Japon), 29 sept- 2 oct 2009.
12. “*How to interface a human operator and a motion planning algorithm*”
D. Flavigné et M. Taïx. *Workshop A Improving Human Robot Communication*, Toyama (Japon), 28 Septembre 2009.
13. “*A computational Approach from robotics for testing eye-centered vs body-centered reaching control,*”
Minh Tuan Tran, Philippe Soueres, Michel Taïx et Benoit Girard. *Progress in Motor Control*, Marseille, (France), juillet 2009.
14. “*Haptic assembly and disassembly task assistance using interactive path planning*”
N.Ladeveze, J.Y.Fourquet, B.Puel, M.Taïx. *IEEE Virtual Reality*, Louisiane (USA), Mars 2009.
15. “*A principled approach to biological motor control for generating humanoid robot reaching movements*”
T.M.Tuan, P.Souerès, M.Taïx, E.Guigon. *IEEE International Conference on Biomedical Robotics and Biomechatronics*, Scottsdale (USA), 19-22 Octobre 2008.
16. “*Interactive motion planning in virtual reality environments*”
N.Ladeveze, J.Y.Fourquet, M.Taïx. *10th Virtual Reality International Conference*, Laval (France), 9-11 Avril 2008.
17. “*A paradigm of neurobiological motor principles for the control of humanoid robots reaching motions*”
T.M.Tuan, P.Souerès, M.Taïx, E.Guigon. *ESF-JSPS Frontier Science Conference for Young Researchers Experimental Cognitive Robotics*, Kanagawa (Japon), 9-15 Mars 2008.
18. “*Planning robust landmarks for sensor based motion*”
M. Taïx, A. Malti, F. Lamiriaux. *European Robotics Symposium (EUROS)*, Prague (République Tchèque), 26-28 Mars 2008.

¹sélectionné parmi les cinq meilleurs papiers de la conférence

19. "Algorithme de planification de trajectoires pour engins agricoles dans des polygones non convexes"², M. Tran, M. Taïx et P. Souères. *Recherche, Innovation et Vision du Futur, RIVF*, Université de Can Tho, Vietnam, février 2005.
20. "Sensor-landmark motion planning in mobile robots " A. Malti, F. Lamiroux et M. Taïx. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japon, 28 septembre - 2 octobre 2004.
21. "Sensor-landmark succession for motion planning along a planned trajectory " A. Malti, F. Lamiroux et M. Taïx. *Mechatronics and Robotics 2004 (MECHROB'04)*, Aachen, Allemagne, 13-15 septembre 2004.
22. "Path Planning for Complete Coverage with Agricultural Machines" M. Taïx, P. Souères, H. Frayssinet et L. Cordesses. *Field and Services Robotics (FSR'2003)*, Japon, 14-16 juillet 2003.
23. "Farmwork path planning for field coverage with minimum overlapping" M. Taïx, S. Fabre, P. Souères et L. Cordesses. *IEEE International Conference on ETFA*, Nice, France, 16-18 octobre 2001.
24. "Robust motion planning for rough terrain navigation" A. Haït, T. Siméon et M. Taïx. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99)*, Kyongju, Corée, 17-21 octobre 1999.
25. "A landmark-based motion planner for rough terrain navigation" A. Haït, T. Siméon et M. Taïx. *5th International Symposium on Experimental Robotics (ISER'97)*, Barcelone, Espagne, 15-18 juin 1997.
26. "Polyhedron placement for planar motion" T. Siméon, P. Juhel, M. Taïx et J.P. Laumond. *International Conference on CAD/CAM, Robotics and Factories of the Future*, Metz, France, août 1992.
27. "Motion Planning for a Planetary Rover" R. Chatila, J.P. Laumond, T. Siméon et M. Taïx. *International Advanced Robotics Programme on Robotics in Space*, Pise, Italie, juin 1991.
28. "Automatic Planning of a Grasp for a "Pick and Place " Action" P. Violero, I. Mazon et M. Taïx. *IEEE International Conference on Robotics and Automation*, Cincinnati, USA, mai 1990.
29. "A Complete Iterative Motion Planner for a Car-like Robot" P. Jacobs, J.P. Laumond et M. Taïx. *Journées de Géométrie Algorithmique, INRIA*, Sophia-Antipolis, France, juin 1990.
30. "A motion planner for car-like robots based on a mixed global/local approach" P. Jacobs, J.P. Laumond et M. Taïx. *IEEE International Workshop on Intelligent Robots and Systems*, Tsuchiura, Japon, juillet 1990.

Rapports de recherches

²Prix du meilleur papier dans la session "Optimization and Operational Research"

31. "*Projet EGOCENTRE : enchaînement dynamique de tâches référencées multi-capteurs pour l'exécution de mouvements complexes en robotique mobile*" A. Argentieri, P. Baronne, V. Cadenat, S Celebrini, F. Chaumette, P. Chiron, P. Danès, J.Y. Fourquet, B. Gao, N. Mansard, V. Padois, P. Souères, M. Taïx, Y. Trotter
2èmes Journées du Programme Interdisciplinaire ROBEA, Toulouse France, 27-28 janvier 2004, pp.135-142
32. "*Sensor-landmark based motion in mobile robots*"
A. Malti, F. Lamiriaux et M. Taïx
Rapport LAAS N°03567, décembre 2003, 7p.
33. "*Enchaînement dynamique de tâches référencées capteurs pour l'exécution de mouvements complexes en robotique mobile*"
D. Bellot, V. Cadenat, S Celebrini, P. Chiron, P. Danès, M. Djeddou, J.Y. Fourquet, V. Padois, P. Souères, M. Taïx, Y. Trotter
Actes des Journées du programme Interdisciplinaire ROBEA, 24 et 25 octobre 2002
34. B. Gorla, J-Y. Fourquet & M. Taïx.
"*Robotics*", programme *COMETT*,
95 pages, janvier 1994.(2 autres versions : espagnol et français).
35. "*Génération des déplacements contraints et non-contraints*"
R. Chatila, B. Dacre Wright, M. Devy, S. Lacroix, J.P. Laumond, F. Nashashib, T. Siméon, M. Taïx
Rapport de recherche LAAS No 93036 - Groupement RISP, Projet VAP.
Toulouse - janvier 1993.
36. "*Rapport de fin de phase 3 du projet V.A.P. -Thème 4 : Génération des déplacements autonomes*"
R. Chatila, J.P. Laumond, T. Siméon et M. Taïx
Rapport de recherche LAAS No 90019 - (diffusion restreinte) Toulouse - janvier 1991.
37. "*Fast and exact trajectory planning for mobile robots and other systems with non-holonomic constraints*"
P. Jacobs, J.P. Laumond, R. Murray et M. Taïx
Rapport de recherche LAAS No 90318,
Toulouse - septembre 1990.
38. "*Modules Architecture for Controlling a Mobile Robot with a Sensor-Landmark Based Motion*"
Abed C. Malti, Florent Lamiriaux et Michel Taïx

III

Les cinq publications jugées essentielles

- “*Improving Motion Planning in Weakly Connected Environments*”
D. Flavigné et M. Taïx,
dans *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taiwan, octobre 2010.
- “*A principled approach to biological motor control for generating humanoid robot reaching movements*”
Minh Tuan Tran, Philippe Soueres, Michel Taïx et Emmanuel Guigon,
dans *IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechanics, BIOROB08*, pages 783 -788, 2008.
- “*Planning robust landmarks for sensor based motion*”,
M. Taïx, A. Malti, F. Lamiriaux,
dans *Springer Tracts in Advanced Robotics*, vol 44, pages 195-204, Springer, 2008.
- “*Algorithme de planification de trajectoires pour engins agricoles dans des polygones non convexes*”, M. Tran, M. Taïx et P. Souères,
dans *Recherche, Innovation et Vision du Futur, RIVF*, Université de Can Tho, Vietnam, février 2005.
- “*Algorithms for rough terrain trajectory planning*”
A. Haït, T. Siméon et M. Taïx,
dans *Advanced Robotics*, vol 16, no 8, décembre 2002.