

*Projet Sharp**Programmation automatique et systèmes  
décisionnels en robotique**Rhône-Alpes*

THÈME 3B

*R* *apport*  
*d'Activité*

2002



# Table des matières

<b>1. Composition de l'équipe</b>	<b>1</b>
<b>2. Présentation et objectifs généraux</b>	<b>2</b>
<b>3. Fondements scientifiques</b>	<b>3</b>
3.1. Introduction	3
3.2. Planification de mouvements	3
3.2.1. Contraintes cinématiques.	3
3.2.2. Contraintes additionnelles de l'environnement.	4
3.2.3. Incertitude géométrique.	4
3.3. Architecture décisionnelle pour l'autonomie de mouvement.	5
3.4. Synthèse du mouvement virtuel et interactions 3D	5
3.4.1. Modèles et algorithmes de base.	6
3.4.2. Interaction 3D.	6
3.4.3. Synthèse du mouvement.	7
3.5. Outils probabiliste pour la géométrie et la robotique	7
<b>4. Domaines d'application</b>	<b>8</b>
4.1. CAO-Robotique	8
4.2. Route automatisée	8
4.3. Réalité virtuelle	9
4.3.1. La robotique d'intervention.	9
4.3.2. Les applications médicales.	9
<b>5. Logiciels</b>	<b>9</b>
5.1. Aladyn3D	9
5.2. OpenPI	10
5.2.1. Développement de nouvelles fonctionnalités de OpenPI	10
5.2.2. Application de OpenPI	11
<b>6. Résultats nouveaux</b>	<b>11</b>
6.1. Environnement dynamique partiellement connu	11
6.1.1. États de collision inévitable	11
6.1.2. Contraintes perceptives et connaissances partielles	13
6.1.3. Contraintes dynamiques et connaissances partielles	14
6.1.4. Détermination des trajectoires instantanées de fuite dans un environnement dynamique	15
6.2. Localisation et construction de carte (SLAM)	16
6.2.1. Invariants géométriques et théorie des graphes pour un SLAM robuste	16
6.2.2. Mise en correspondance implicite pour la localisation	17
6.3. Exécution du mouvement	17
6.3.1. Commande en boucle fermée du Cycab	17
6.3.1.1. Antécédents	17
6.3.1.2. Système commandable équivalent du Cycab	18
6.3.1.3. Linéarisation du Cycab	18
6.3.1.4. Loi de commande pour système linéaire	19
6.3.2. Evitement d'obstacle par inférence bayésienne	21
6.3.3. Expérimentations avec le Cycab	22
6.4. Synthèse du mouvement virtuel et interaction 3D	22
6.4.1. Modèles Physiques	22
6.4.2. Interactions 3D	25
6.4.2.1. Détection de collision.	25
6.4.2.2. Découpe 3D et interaction haptique.	25

6.4.3.	Simulateurs Médicaux	27
6.4.3.1.	Simulateur de Laparoscopie.	27
6.4.3.2.	Simulateur d'Anthroscopie	29
6.4.4.	Synthèse du mouvement virtuel	30
6.4.4.1.	Robot Virtuel.	30
6.4.4.2.	Locomotion.	30
6.4.4.3.	Suivi de chemin.	31
6.4.4.4.	Planification.	31
6.4.4.5.	Programmation comportementale bayésienne.	31
6.4.4.6.	Résultats.	32
6.5.	Outils probabiliste pour la géométrie et la robotique	33
6.5.1.	Programmation des robots	33
6.5.2.	Fusion de données sensorielles	35
6.5.3.	Construction de cartes sensori-motrices pour la mobilité	35
6.5.4.	Raisonnement probabiliste dans le domaine médical	37
<b>7.</b>	<b>Contrats industriels</b>	<b>37</b>
7.1.	Actions de développement Praxitèle et LaRA	37
7.2.	Projet Priamm « Studio virtuel »	37
7.3.	Robosoft	38
7.4.	CyberCars	38
7.5.	Carsense	38
7.6.	Arcos 2003	38
7.7.	Biba	38
<b>8.</b>	<b>Actions régionales, nationales et internationales</b>	<b>39</b>
8.1.	Actions régionales et nationales	39
8.1.1.	Projet Gravir-Leibniz.	39
8.1.2.	Pôles CNRS Structures et Machines Intelligentes.	39
8.1.3.	Projet Caesare.	39
8.1.4.	Projet Robea ParkNav, « Interprétation de scènes dynamiques complexes et planification réactive de mouvement ».	39
8.1.5.	Projet Robea « Commande de manipulateur mobiles non-holonomes ».	39
8.1.6.	Projet Robea « Un androïde doué de parole : Robotique Cognitive de la Parole et du Langage ».	39
8.1.7.	Projet Robea « Modèles bayésiens pour la génération de mouvement ».	40
8.2.	Relations bilatérales internationales	40
8.2.1.	Russie et Asie Centrale	40
8.2.1.1.	Institut Franco-Russe Liapunov.	40
8.2.2.	Pacifique et Asie du Sud	40
8.2.2.1.	Collaboration avec le Japon.	40
8.2.2.2.	Collaboration avec Singapour.	40
8.2.3.	Amérique du Nord	40
8.2.3.1.	Collaboration avec l'université de Vancouver (Canada).	40
8.2.3.2.	Collaboration avec l'université de Californie, Berkeley (USA).	40
8.2.3.3.	Collaboration avec l'université de Stanford.	40
8.2.4.	Amérique du Sud et Amérique Centrale	41
8.2.4.1.	Collaboration avec le Mexique.	41
8.2.4.2.	Coopération avec l'université de Brasilia (Brésil).	41
8.3.	Accueils de chercheurs étrangers	41
<b>9.</b>	<b>Diffusion des résultats</b>	<b>41</b>
9.1.	Animation de la communauté scientifique	41

---

9.2. Enseignements universitaires	41
9.3. Participation à des colloques, séminaires, invitations	42
<b>10. Bibliographie</b>	<b>42</b>



# 1. Composition de l'équipe

## Responsable scientifique

Christian Laugier [DR Inria]

## Secrétariat

Véronique Roux [AJT Inria]

## Personnel Inria

Thierry Fraichard [CR]

Sepanta Sekhavat [CR]

## Personnel CNRS Centre National de la Recherche Scientifique.

Emmanuel Mazer [DR]

Pierre Bessière [CR]

## Enseignant chercheur

Olivier Aycard [Maître de conférence UJF]

## Ingénieurs experts Inria

Juan-Manuel Ahuactzin Larios [2001-2004]

Olivier Lebeltel [2000-2003]

Kamel Mekhnacha [2002-2004]

## Collaborateur extérieur

Olivier Aycard [maître de conférence UJF]

## Chercheurs doctorants

Diego d'Aulignac [allocataire MENESR]

Francis Colas [boursier inria]

Christophe Coué [boursier Inria]

Ruben Garcia Senen [boursier du gouvernement mexicain, CONACYT/SFERE]

Carla Koike [boursière du gouvernement brésilien CAPES]

Frédéric Large [boursier Inria]

Ronan Le Hy [allocataire MENESR]

Mikaël Kais [Boursier INRIA, co-encadré avec I. Masaki, MIT]

César Mendoza Serrano [boursier du gouvernement mexicain, CONACYT/SFERE]

Cédric Pradalier [allocataire MENESR]

David Raulo [allocataire MENESR]

Kenneth Sundaraj [boursier ambassade de France]

Jorge Hermosillo Valadez [boursier du gouvernement mexicain, CONACYT/SFERE]

Miriam Amavicza [boursière mexicaine CONACYT]

## Stagiaires

Stephane Ayache [magistère informatique UJF]

Pierre Billiau [ingénieur ENSIMAG]

Stéphane Blondin [DEA IVR]

Habib Bouaziz-Viallet [ingénieur CNAM]

Anant Choksuriwong [DEA IVR]

Matthieu Dufourneud [DEA IVR]

Itab El Mortaji [magistère informatique UJF]

Sébastien Fave [licence UJF]

Denis Jacquet [ingénieur INSA Lyon]

Ruth Lezama Morales [Universidad de las Américas, Puebla (MX)]

Francisco Moo Mena [Univ. Autonome du Yucatan (MX)]

Priscilla Pek [DEA IVR, coopération NTU, Singapour]

Frédéric Raspail [DEA ISC]  
Michael Sdika [DEA IVR]  
Emy Sukowo [NTU]  
Adriana Tapus [DEA ISC]  
Yoann Toussaint [DEUG Univ. de Savoie]  
Alejandro Vargas Hernandez [master ITESM, Cuernavaca (MX)]  
Alejandro Dizan Vasquez Govea [Instituto tecnológico y de Estudios Superiores de Monterrey (MX)]  
Michel Verlinden [Univ. Catholique de Leuven, Belgique]  
Stéphane Viera [ingénieur CNAM]  
Cyril Zorman [magistère CCI, UJF]

#### Visiteurs

Zvi Shiller [professeur à University of California, Los Angeles]  
Oussama Khatib [professeur à Stanford University]

## 2. Présentation et objectifs généraux

Sharp est un projet de recherche commun entre l'Inria, le CNRS<sup>2</sup>, l'INPG<sup>3</sup>, l'UJF<sup>4</sup> Grenoble, depuis 1992 ; il est localisé à l'Inria Rhône-Alpes, et il appartient également au laboratoire Gravir<sup>5</sup> de la fédération Imag.

Le programme scientifique du projet Sharp porte sur le thème suivant : « modélisation et synthèse du mouvement et des interactions 3D en robotique ». Le terme « robotique » revêt ici un caractère particulier, dans le sens où il inclut à la fois des machines physiques (communément appelées « robots ») capables d'actions autonomes dans le monde réel, et des agents mobiles ou articulés (où « robots virtuels ») possédant des capacités de mouvements propres leur permettant d'évoluer de manière dite « intelligente » dans un monde virtuel soumis à des lois physiques semblables à celles du monde réel. Ce programme de recherche met l'accent sur les composantes « modélisation », « autonomie du mouvement », et « interaction physique ». Les principales applications visées sont celles de la CAO-Robotique, de la route automatisée, de la robotique d'intervention, de la santé, et des environnements virtuels dynamiques. Le programme de recherche associé s'articule autour des principaux axes suivants :

- Planification de mouvements : conception et implantation d'algorithmes de planification de mouvements ayant la capacité de prendre en compte à la fois les contraintes classiques de non collision, et celles provenant de la nature physique du monde dans lequel les robots (réels ou virtuels) évoluent (en particulier : contraintes cinématiques et dynamiques, incertitude) ;
- Architectures décisionnelles pour le mouvement : méthodologie pour le développement de structures décisionnelles pour le contrôle « intelligent » des mouvements, actions, et interactions d'agents évoluant dans des environnements dynamiques peu ou pas connus a priori ;
- Synthèse du mouvement virtuel et interactions 3D : modèles et algorithmes pour le traitement en temps réel ou en temps interactif de la dynamique et des interactions physiques de corps complexes en mouvement dans un monde virtuel (mouvements divers, déformations, collisions, changements interactif de topologie, interaction haptique) ;
- Outils probabilistes pour la géométrie et la programmation des robots : développement d'outils de calcul probabiliste pour la géométrie et la programmation de robots (réels ou virtuels), afin de traiter correctement (*i.e.* en exploitant le cadre formel de la théorie des probabilités et du calcul bayésien) les incertitudes et leurs impacts sur les problèmes de programmation et sur les problèmes d'interprétation de données sensorielles.

---

<sup>2</sup>Centre National de la Recherche Scientifique

<sup>3</sup>Institut National Polytechnique de Grenoble

<sup>4</sup>Université Joseph Fourier

<sup>5</sup>Laboratoire informatique Graphique, Vision, et Robotique, UMR CNRS 5527

L'activité de recherche précédente est à la fois valorisée et fertilisée par des activités plus appliquées qui visent au développement de solutions à des problèmes industriels. Plusieurs prototypes de recherche et expérimentations réelles (*e.g.* sur des robots, des véhicules, ou des systèmes de Réalité Virtuelle) sont ainsi réalisés en relation avec des industriels et les moyens robotique de l'Inria Rhône-Alpes ; certains de ces prototypes ont déjà donné lieu à des transferts de technologies en CAO-Robotique, et dans le domaine des effets spéciaux en vidéo professionnelle.

## 3. Fondements scientifiques

### 3.1. Introduction

Le thème central du projet Sharp est celui de « la modélisation et la synthèse du mouvement et des interactions 3D en robotique ». Ce thème de recherche est abordé en mettant l'accent sur les composantes « modélisation », « autonomie du mouvement », et « interaction physique ». Ceci se traduit par l'étude de quatre sujets principaux suivants : planification de mouvements, architecture décisionnelle pour l'autonomie du mouvement, synthèse du mouvement virtuel et interactions 3D, outils probabiliste pour la géométrie.

### 3.2. Planification de mouvements

La planification de mouvement est un problème fondamental largement traité en Robotique [57]. Notre volonté de produire des plans exécutables dans des environnements réels nous amène à prendre en compte de façon explicite des contraintes autres que la simple contrainte géométrique de non-collision habituellement considérée. Ces contraintes découlent d'une part de la nature physique du monde réel (cinématique et dynamique des systèmes robotiques considérés, interactions de contacts, incertitudes), et d'autre part de la nature des environnements considérés (systèmes multi-robots, obstacles mobiles, environnements évolutifs, etc.). Ces différentes contraintes posent des problèmes particuliers pour lesquels nous nous attachons à trouver des solutions, tout en cherchant à maîtriser la complexité algorithmique induite. Nos travaux de recherche portent principalement sur les trois volets suivants du problème : les contraintes cinématiques, les contraintes additionnelles de l'environnement, et l'incertitude géométrique.

#### 3.2.1. Contraintes cinématiques.

Les robots mobiles à roue, sont tous soumis à la contrainte de roulement sans glissement de leurs roues sur le sol. Cette contrainte se répercute par une contrainte cinématique dite « non holonome » sur les mouvements instantanés exécutables par le système. Ce problème intéresse tout particulièrement la communauté robotique depuis qu'il est apparu pour la première fois en 1986 [60]. La non holonomie contraint la forme géométrique des mouvements que peut effectuer le système. Cela dit, pour une grande majorité des robots mobiles considérés dans la littérature, il existe une bijection entre le chemin suivi par le système dans son espace de configuration de dimension  $n$  et la courbe 2D effectuée dans le plan de travail par un point caractéristique du robot. On peut alors raisonner dans le plan. Cependant, les robots mobiles sont aussi soumis à d'autres types de contrainte tels que l'existence d'une borne sur l'angle de braquage des roues avant d'une voiture, borne sur les vitesses et accélérations de braquage et linéaires, etc. Certaines de ces contraintes peuvent être traduites géométriquement au niveau du chemin 2D du point caractéristique du robot. Ce traitement nécessite quelquefois (c'est souvent le cas des bornes dans l'espace des commandes) un renforcement des contraintes ou la prise en compte d'un modèle plus fin du robot. Dans nos travaux, nous nous sommes efforcés d'enrichir le problème de la planification non holonome dans le cas des robots mobiles en y intégrant d'autres contraintes fréquentes pour ces systèmes. Dans le cas de robots de type « voiture », nous utilisons des chemins dont la courbure est continue et dont la dérivée de la courbure est bornée<sup>6</sup>. Les chemins ainsi obtenus sont quasi-optimaux et présentent un grand intérêt pratique [52]. Dans le cas du véhicule « Cycab » de l'Inria<sup>7</sup> [63], qui

<sup>6</sup>la continuité de courbure correspond à la continuité de la variation de l'angle de braquage des roues avant, tandis que la borne sur la dérivée de la courbure correspond au fait que la vitesse de variation de l'angle de braquage est limitée.

<sup>7</sup>Plate-forme expérimentale de l'Inria commercialisée par la société Robosoft, et au développement de laquelle le projet Sharp a largement contribué.

possède un modèle cinématique très particulier du fait de la rotation couplée de ses roues avant et arrière (ce qui permet d'accroître la manœuvrabilité du système dans des environnements fortement contraints), il faut faire face à des problèmes nouveaux de modélisation du système et de formalisation pour pouvoir résoudre le problème de planification de mouvement associé. Notre approche consiste à utiliser les outils de l'algèbre différentielle, et plus particulièrement ceux liés au concept de « platitude différentielle » [48], afin de pouvoir ramener le problème de la planification pour ce système 4D à un problème de recherche de courbes planes, comme c'est le cas pour les robots de type voitures.

### 3.2.2. Contraintes additionnelles de l'environnement.

Comme cela a été mentionné précédemment, un robot mobile est soumis à des contraintes non holonomes, se traduisant par une modélisation de la cinématique du système à l'aide d'une famille d'équations différentielles non intégrables. Par ailleurs, l'environnement dans lequel évolue le robot est à l'origine de contraintes qui ne peuvent généralement être intégrées directement dans cette formalisation, en particulier : la présence dans l'environnement d'obstacles ou d'autres robots, les interactions roue-sol pour un robot évoluant sur un terrain accidenté ou à adhérence irrégulière, etc. C'est pourquoi, il est généralement nécessaire d'utiliser des planificateurs dits « locaux » aptes à prendre en compte les diverses contraintes de déplacement, et de les combiner avec des schémas algorithmiques dits « globaux » pour permettre l'évitement des obstacles. Ceci peut être fait soit par des méthodes de type « roadmap », soit par modifications successives d'un chemin sans collision calculé auparavant. Le cas de la locomotion en milieu tout-terrain est un cas typique de la difficulté qu'il y a à modéliser les équations différentielles de mouvement du robot de manière « closed form ». Ceci s'explique par la présence de plusieurs facteurs qui entrent en considération, notamment la dynamique du robot, la cinématique de la chaîne du châssis si le robot est articulé, les interactions de contact roue-sol (glissement, roulement, déformation des surfaces de contact), et enfin le relief du terrain. Ceci signifie que des modèles mixtes (géométriques et physiques) couplés avec des techniques de calcul appropriées doivent être utilisés au niveau de la planification de mouvements, et que le schéma « local-global » précité doit être soigneusement adapté pour pouvoir maîtriser la forte complexité algorithmique induite [43].

### 3.2.3. Incertitude géométrique.

Dans certains cas, et notamment pour la navigation en robotique mobile, il est important que les chemins calculés soient « sûrs », *i.e.* pour lesquels le robot est assuré d'atteindre son but malgré les incertitudes qui peuvent affecter le robot (incertitudes sur la position du robot ou des obstacles, sur le résultat d'une commande appliquée au robot, sur une mesure capteur, etc.). En particulier, dans le cas d'un robot mobile équipé d'un système de localisation relative (odométrie), l'incertitude sur la position du robot, de par sa nature cumulative, peut devenir telle qu'elle rende impossible le suivi d'un chemin donné. Il devient donc nécessaire de considérer ces incertitudes dès la planification, afin d'engendrer des solutions sûres ; ceci se traduit en pratique par l'introduction de « stratégies de mouvement » qui peuvent inclure l'utilisation de capteurs, et de mouvements gardés ou compliants. Les premiers travaux dans ce domaine ont été réalisés dans le domaine de l'assemblage de pièces mécaniques. Un premier type d'approche consiste à calculer une séquence d'assemblage en supposant l'incertitude nulle, puis à modifier la séquence ainsi obtenue afin de prendre en compte l'incertitude. Ce type d'approche se prête bien à l'assemblage : la structure du plan final est, en général, proche de celle du plan initial, grâce à la présence de « guides géométriques » engendrés par les surfaces en contact. Ce n'est cependant pas le cas en robotique mobile. Une autre approche consiste à utiliser le concept de « pré-image » [64], *i.e.* l'ensemble des configurations à partir desquelles une commande donnée permet d'atteindre à coup sûr un objectif donné. La planification par chaînage arrière de pré-images intègre l'incertitude directement dans le processus de planification, mais elle pose des problèmes de complexité algorithmique tels que cela a limité son utilisation à des cas très simples. Notre objectif est d'aborder la prise en compte de l'incertitude dans des cas plus réalistes (robot non ponctuel et non holonomes, mouvements non rectilignes, etc.), ce qui nous amène à développer des modèles d'incertitude plus complexes et de nouvelles techniques de planification [45][51]

### 3.3. Architecture décisionnelle pour l'autonomie de mouvement.

Il est maintenant acquis que l'autonomie de mouvement dans un monde dynamique nécessite de passer par le couplage de deux types de fonctions : des « fonctions délibératives », dotées de capacités de modélisation et de raisonnement à différents niveaux d'abstraction, et des « fonctions réactives », qui privilégient le temps de réaction à un événement donné (l'impératif temps réel implique souvent des capacités de raisonnement plus réduites, voire nulles). Le problème abordé est alors celui du développement d'une architecture de contrôle visant à doter un véhicule de la capacité de se déplacer de façon autonome dans un environnement dynamique partiellement connu. Comme indiqué précédemment, ceci implique de coupler de manière appropriée des fonctions délibératives qui utilisent des modèles et des raisonnements variés, avec des fonctions réactives qui utilisent des données de capteurs et des comportements réactifs avec des constantes de temps différentes. Ce couplage se réalise par le biais d'une « architecture de contrôle » qui gère les interactions entre ces différentes fonctions. Trois grandes classes d'approches existent dans la littérature : délibérative [69], réactive [42], et hybride [36]. Dans notre cas, nous avons choisi d'utiliser une approche de type hybride à trois composantes (Fonctionnelle, Exécutive, Décisionnelle) [58], permettant de mieux gérer les contraintes de temps réel ainsi que le couplage des niveaux délibératifs et réactifs. L'idée de base, consiste à utiliser la notion de « manœuvre type » adaptée à un objectif et à un type de situation donnés (un « contexte »). Le principe consiste à disposer d'un ensemble de schémas de contrôle (ou « manœuvre asservie à des capteurs ») associés à chaque contexte (*e.g.* parking automatique, évitement d'obstacle, etc.), et paramétrés par des informations capteurs (*e.g.* vitesse du véhicule, distance aux obstacles, etc). Cette approche est bien adaptée au contexte des véhicules automatiques en environnement partiellement structuré pour lesquels il semble exister un nombre raisonnable de schémas différents ; elle permet d'intégrer correctement les différentes couches décisionnelles tout en assurant un meilleur contrôle des mouvements exécutés par le véhicule (trajectoire, vitesses, accélérations).

### 3.4. Synthèse du mouvement virtuel et interactions 3D

Les technologies de Réalité Virtuelle deviennent de plus en plus populaires dans de nouveaux secteurs d'application comme la robotique d'intervention, la santé, ou encore les loisirs (jeux vidéo en particulier). Toutes ces applications nécessitent un degré de réalisme élevé sur trois composantes essentielles du processus : le rendu graphique, la dynamique des scènes traitées, et l'interaction 3D ; elles imposent également des contraintes strictes de temps réel (ou au moins de temps interactif). L'approche généralement mise en oeuvre pour tenter d'obtenir ce réalisme au niveau de la dynamique des scènes virtuelles et de l'interaction, consiste à doter le monde virtuel de propriétés et de lois physiques semblables à celles du monde réel. C'est dans cette logique que se placent nos travaux de recherche sur la synthèse du mouvement virtuel et les interactions 3D, en s'appuyant naturellement sur les synergies qui existent entre les techniques de génération et de contrôle du mouvement en robotique, et celles habituellement mises en oeuvre en animation graphique (approches basées sur les modèles physiques). Cependant, les applications que nous considérons, qui mettent généralement en jeu des processus du monde réel (*e.g.* en robotique d'intervention ou dans le domaine médical), nous conduit à aborder le sujet sous un angle un peu différent : nous devons mettre l'accent sur le « réalisme physique » et sur les contraintes de « temps réel ou interactif ».

Compte tenu de la présence de nombreuses similitudes avec les problèmes de robotique, nous avons choisi de mettre en oeuvre une approche visant à considérer une scène virtuelle comme un espace soumis aux principales lois de la physique, et dans lequel évoluent et interagissent deux classes d'objets : (1) des « objets dits inertes », dont les mouvements et/ou les déformations résultent de l'application de champs de forces (comme la gravité) ou de l'interaction physique avec d'autres objets (manipulations, collisions) ; et (2) des « objets actifs » (ou « robots virtuels ») possédant des caractéristiques physiques prédéfinies (*i.e.* mécanique, cinématique, et dynamique) et dotés de capacités de mouvements propres. Certaines techniques de la robotique peuvent potentiellement être utilisées ou adaptées pour cela, comme par exemple la planification de mouvements, le contrôle du mouvement, les comportements réactifs, ou encore la détection des collisions ; la principale difficulté réside dans l'intégration de telles méthodes avec une approche basée sur l'utilisation de modèles physiques de corps solides ou déformables (comme par exemple les approches proposées en

animation graphique par [39] et [40]). Cette analyse, que nous avons faite lors de nos premiers travaux pour la prise en compte des interactions roue-sol lors de la simulation du comportement dynamique d'un véhicule tout-terrain [59], a été aussi faite plus tard par des chercheurs en robotique qui ont apportés des contributions intéressantes qui vont dans le même sens (en particulier les travaux de Mirtich à Berkeley [68] sur les collisions entre objets rigides, ceux de Ruspini & Khatib à Stanford [72] sur la simulation dynamique en robotique et sur l'interaction haptique, et ceux de Latombe & Kuffner à Stanford [57] sur la planification de mouvements pour l'animation graphique).

L'approche que nous mettons en oeuvre pour résoudre ce problème nous conduit à aborder quatre sujets complémentaires dont les orientations sont guidées par les contraintes énoncées précédemment : les modèles et algorithmes de base, l'interaction 3D, et la synthèse du mouvement.

#### 3.4.1. Modèles et algorithmes de base.

Il s'agit ici de développer des modèles et des algorithmes qui permettent de reproduire le comportement dynamique des divers composants d'une scène virtuelle en temps réel ou interactif, tout en conservant une certaine cohérence physique et en essayant de contrôler les problèmes numériques liés à l'intégration des équations de la dynamique. Ces modèles comportent une composante géométrique et une composante physique. La composante géométrique est essentiellement utilisée pour le rendu graphique, pour la détection des collisions et des contacts, et pour les interactions qui mettent en jeu des modifications de topologie ; la composante physique est utilisée pour le calcul du mouvement et/ou des déformations, et pour la détermination des effets des contacts et des collisions (le choix du modèle et de la méthode d'intégration numérique étant alors liés aux caractéristiques physiques des objets considérés). La méthodologie que nous appliquons pour modéliser des objets physiques complexes comporte trois grandes étapes : (1) construction d'un maillage 2D et/ou 3D à partir d'un modèle géométrique ou d'un ensemble de points mesurés sur l'objet considéré (le niveau de résolution utilisé étant évidemment lié aux caractéristiques de déformation à prendre en compte), (2) « placage » d'un modèle physique de déformation sur le maillage géométrique précédent (*e.g.* masses-ressorts, éléments finis...), (3) identification des paramètres du modèle sur la base de comportements requis ou de mesures faites sur les objets réels. Cette approche a déjà été validée expérimentalement lors de la modélisation d'un ligament croisé de genou à partir de données mesurées sur un genou de cochon, ou lors de la simulation des déformations d'une cuisse humaine soumise à des pressions appliquées par un sonde échographique virtuelle [77].

#### 3.4.2. Interaction 3D.

Il s'agit ici de développer des modèles et des algorithmes qui permettent de reproduire le comportement dynamique d'objets en interaction physique (contacts, articulations, collisions, actions produisant des modifications de structure comme la découpe, et interaction haptique). Ces modèles doivent être à la fois réalistes sur le plan physique, et efficaces sur le plan algorithmique. En effet, les interactions représentent généralement le goulot d'étranglement des systèmes de simulation, elles sont à l'origine des discontinuités qui altèrent profondément les procédures numériques de calcul, et elles constituent une base essentielle du réalisme que l'on attribue aux scènes dynamiques simulées. Les approches utilisées pour modéliser les différents types d'interactions sont par nature différentes, bien que de nombreux auteurs aient tentés d'étendre artificiellement leurs modèles de collision au traitement d'autres types d'interactions (*cf.* par exemple [68]). Nos travaux de recherche portent essentiellement sur trois types d'interactions : les contacts et collisions, les modifications interactives de topologie, et l'interaction haptique. Notre approche pour le traitement des contacts non restants (*e.g.* contacts de type roue-sol) et des collisions, consiste à utiliser des algorithmes optimaux de calcul de distances dérivés de la robotique (*e.g.* les algorithmes de Lin & Canny ou ceux de Gilbert & Johnson), à développer des algorithmes efficaces pour la détermination des caractéristiques géométriques des contacts et des interpénétrations, et à mettre en oeuvre des algorithmes de calcul de la réponse dynamique à la collision qui utilisent les lois connues de la mécanique et qui prennent en compte les problèmes d'intégration numérique associés [47]. Les modifications interactives de structures topologiques engendrées par certaines actions (déchirement, découpe) mettent en jeu des procédures complexes de remaillage et de redistribution des propriétés physiques pour lesquelles nous avons obtenu des résultats récents dans le cadre de la simulation

chirurgicale. L'interaction haptique, qui met en jeu un système mécanique articulé actif (*i.e.* de type robot), engendre des problèmes de contrôle (stabilité) et nécessite des fréquences d'échantillonnage de l'ordre du kilohertz afin de rendre le retour tactile interprétable par un opérateur humain ; or la simulation dynamique de scènes complexes, même lorsque le code est optimisé, n'arrive généralement pas à des fréquences de calcul supérieures à 100 Hz. C'est pourquoi, nous nous intéressons au développement de « modèles haptique locaux » restituant en temps réel les principales caractéristiques topologiques et mécaniques locales des scènes considérées [72] [38].

### 3.4.3. Synthèse du mouvement.

Le problème abordé ici est celui de la synthèse du mouvement pour des agents virtuels (ou robots virtuels). Cela englobe les aspects dynamiques du mouvement (c'est à dire la production d'un mouvement par application de forces ou de contrôles au niveau de certains organes des agents virtuels considérés), ainsi que les aspects de plus haut niveau liés au comportement et à la planification de mouvements. Notre objectif est de tenter d'appliquer des techniques que nous développons par ailleurs pour les robots réels (planificateurs géométriques, architectures décisionnelles, contrôleur de mouvements), tout en exploitant le fait que le monde virtuel permet dans certains cas de réduire la complexité globale (par suppression de certaines contraintes du monde réel, ou par introduction de « guides physiques artificiels »). Des premiers résultats sur ce point ont été obtenus pour planifier les mouvements de systèmes articulés divers, en surimposant des champs de forces fictifs sur un modèle physique de l'environnement, ou en introduisant un modèle physique simplifié d'interaction roue-sol au niveau de la composante locale d'un planificateur de trajectoire pour véhicule tout-terrain. D'autres résultats plus récents concernent le contrôle temps réel d'agents virtuels autonomes ; l'approche adoptée consiste à modéliser ceux-ci comme des robots se déplaçant dans un environnement virtuel, perçu au moyen de pseudo-capteurs, et utilisant pour se déplacer des fonctions de planification et de contrôle de mouvements.

## 3.5. Outils probabiliste pour la géométrie et la robotique

La prise en compte des incertitudes est un des problèmes importants de la robotique. Raisonner sur des informations incertaines pour programmer un robot constitue un des thèmes récurrents de cette discipline. En particulier, on peut se référer aux travaux déjà anciens de Taylor, Lozano-Perez et Mason [64], ou ceux menés dans notre équipe [75][45]. Pour des raisons de commodité de calcul, la plupart des chercheurs cherchent à s'écarter de la théorie des probabilités en adoptant des simplifications parfois excessives : représentation ensembliste des erreurs ou filtre de Kalman. Dans ce travail, nous souhaitons au contraire utiliser les modèles probabilistes, et plus spécialement le calcul bayésien, pour représenter l'incertitude. Nous cherchons en particulier à nous doter d'outils de modélisation et de calcul probabiliste pour la géométrie et la programmation des robots. Les résultats attendus ne doivent pas seulement nous permettre de représenter les incertitudes et de calculer leurs propagations, mais aussi de traiter, dans un cadre unifié, une large catégorie de problèmes robotique liée à l'incomplétude des modèles de représentation de l'environnement.

En cela nous nous inscrivons dans un mouvement international plus large : celui de la robotique probabiliste [74]. Ces approches reviennent en force dans ce domaine et ont données lieu à de nombreuses démonstrations dans des environnements réels montrant des niveaux de robustesse et de complexité jamais atteints [73]. Les conséquences de l'approche probabiliste se mesurent sur les problèmes de localisation, de fusion multi-capteurs, de contrôle et de planification, mais aussi dans le domaine des architectures de commande de robot. Elle donne une vision nouvelle à la notion de modularité en remettant partiellement en cause l'intérêt du découpage en niveau que l'on trouve généralement dans les architectures classique de commande de robot.

Les fondements théoriques de notre approche sont ceux des probabilités bayésiennes. Une formalisation rigoureuse et synthétique de cette approche peut être trouvée dans [56]. Edward T. Jaynes propose en fait une théorie complète et intégrée du raisonnement probabiliste appelée « Probability as Logic » ou PaL qui étend la logique aux probabilités. Notre approche est basée sur la réalisation d'outils informatique permettant de mettre oeuvre concrètement cette théorie. Plusieurs prototypes de moteurs d'inférence ont été développés

et testés sur des applications simples. Ils sont l'équivalent probabiliste des moteurs d'inférence logique tel que ProLog. Nous avons utilisé ces moteurs pour construire un système de CAO robotique probabiliste et un système de programmation de robot. Les techniques utilisées pour réaliser ces moteurs s'appuient sur des techniques de raisonnement symbolique pour la simplification des expressions, des techniques de Monte-Carlo pour l'intégration en grande dimension et sur l'utilisation d'algorithmes génétiques couplés à des méthodes de relaxation pour l'optimisation en présence de contraintes [66].

## 4. Domaines d'application

### 4.1. CAO-Robotique

En quelques années les systèmes de Conception Assistés par Ordinateur (CAO) se sont enrichis de nouvelles fonctionnalités. Ils ne sont plus seulement utilisés pour la conception de pièces mécaniques mais aussi pour la simulation, voire la planification, de procédés industriels complexes. On parle alors plus généralement de « prototypage virtuel ». La fonctionnalité de « fitting simulation » en est un exemple concret : la conception d'un montage mécanique ne consiste plus seulement à définir la géométrie des pièces mécaniques, mais inclut aussi la possibilité de tester la faisabilité de l'assemblage, voire de le planifier. Cette tendance spectaculaire vers la simulation et la planification atteint tous les acteurs de la CAO. Elle conduit à la conception d'outils informatiques permettant la conception d'usines virtuelles, où tous les processus de fabrication peuvent être simulés (par exemple gestes des employés, fonctionnement des machines, quantité de peintures déposée, etc.) pour être optimisés. Un autre domaine d'application important est celui de la médecine et plus particulièrement celui de la chirurgie assistée par ordinateur. Il s'agit dans ce contexte d'aider le chirurgien à planifier son intervention et à le guider durant celle-ci. Dans ce contexte, nos travaux sur l'algorithmique géométrique revêtent une importance particulière car ils permettent :

- d'améliorer les performances des outils de simulation et de planification en fournissant des algorithmes de calcul de distance performants et généraux ;
- de prendre en compte la dynamique de l'interaction entre corps rigides ou déformables ;
- de fournir des outils de planification pour des systèmes non holonomes ou pour des systèmes complexes ;
- de tenir compte des incertitudes et des erreurs de modèles.

### 4.2. Route automatisée

Depuis une dizaine d'années, les différents problèmes liés aux transports routiers (sécurité, efficacité, etc.) prennent de plus en plus d'importance. Ainsi, ils ont fait et font encore l'objet de grands programmes de recherche tant du côté du secteur privé que du secteur public (*cf.* le programme européen Eurêka Prometheus sur les transports routiers [1986-1994] et les actions de développement Inria-Inrets<sup>8</sup> Praxitèle [1994-1997] et LaRa [1998-2002]).

A l'heure actuelle, les recherches portent sur de nouveaux modes de conduite assistée par ordinateur, permettant une simplification et une sécurisation de la conduite, pouvant aller jusqu'à une conduite totalement automatique sur des voies réservées. C'est le concept de la route automatisée.

Ces réflexions esquissent une prospective sur les transports routiers automatisés du futur et proposent des éléments de réponse aux problèmes de sécurité, de débit des infrastructures, de vitesse de déplacement et, en même temps, de réduction de l'impact sur l'environnement. Si la route automatisée est encore un concept assez large, on peut cependant distinguer deux grandes orientations :

- des scénarios portant essentiellement sur « l'intelligence dans le véhicule » ; ces travaux intéressent les constructeurs car il y a un marché immédiat et évolutif ;

<sup>8</sup>Institut National de Recherche sur les Transports et leur Sécurité.

- des scénarii portant essentiellement sur l'aspect « système » (*i.e.* avec en particulier des aspects liés à l'infrastructure) ; ces projets ont un coût plus élevé mais ce sont les seuls qui permettent de réaliser une route complètement automatisée avec de meilleures performances.

Les exemples d'applications sont les trains de véhicules avec ou sans conducteurs, l'automatisation de certains transports publics, la route automatisée pour les camions, l'autoroute à haut débit, etc. La route automatisée concerne de nombreux domaines de recherche et implique au sein de l'Inria plusieurs projets de recherche. Le projet Sharp est impliqué dans cette problématique depuis de nombreuses années (participation au projet Prometheus et à l'action de développement Praxitèle) et contribue principalement sur les aspects liés aux techniques de déplacements en mode automatique ou semi-automatique (*i.e.* conduite assistée).

### 4.3. Réalité virtuelle

La réalité virtuelle est un secteur en pleine croissance, qui recouvre des secteurs d'applications très divers comme la robotique d'intervention, le graphique, la téléconférence et le multimédia. Compte tenu de nos orientations de recherches visant à privilégier le réalisme physique, les interactions, et l'efficacité des algorithmes mis en jeu, les applications que nous considérons dans le domaine de la réalité virtuelle sont les suivantes :

#### 4.3.1. La robotique d'intervention.

Il s'agit de mettre en œuvre des robots sur des tâches d'intervention ou de maintenance à exécuter dans des environnements hostiles et/ou éloignés (sous-marin, spatial, nucléaire, militaire). Nous considérons alors le monde virtuel reconstruit comme une interface évoluée entre l'opérateur et le robot plongé dans son environnement d'opération. Cette interface permet à l'opérateur d'interagir de manière naturelle avec le robot (graphique, système haptique), tout en prenant en compte implicitement les contraintes du monde réel (que ce soit pour des opérations de type téléopéré, ou pour des actions qui mettent en jeu des mécanismes de planification de mouvements). Par exemple, programmer une mission pour un véhicule d'exploration planétaire, pourra conduire à combiner des techniques de planification de trajectoires et/ou de guidage manuel, avec des techniques de simulation dynamique aptes à prendre en compte des phénomènes physiques tels que les glissements ou le patinage des roues lors de la navigation.

#### 4.3.2. Les applications médicales.

Il s'agit de doter le praticien ou le chirurgien d'outils informatiques lui permettant soit d'améliorer l'apprentissage de gestes médicaux, soit de l'aider à préparer certaines interventions chirurgicales par intégration de diverses données préopératoires et de résultats simulés des gestes opératoires projetés. Nos efforts actuels portent essentiellement sur le premier point. L'objectif est de produire des simulateurs permettant d'entraîner les praticiens et chirurgiens sur les techniques opératoires qu'ils devront utiliser (en particulier pour les nouvelles techniques de chirurgie non invasive qui nécessitent d'intervenir avec une réduction du champ visuel et gestuel), tout en donnant la possibilité de simuler de nombreux types de pathologie, même rares. Ceci devrait permettre au chirurgien néophyte d'acquérir ses réflexes sur des patients virtuels, et ainsi réduire le taux d'incidents lors des premières interventions sur des patients réels. Le travail correspondant est réalisé en relation avec des chirurgiens.

## 5. Logiciels

### 5.1. Aladyn3D

*AlaDyn3D* est un logiciel de *simulation dynamique*<sup>9</sup> développé comme une généralisation du système *Robot-Φ*<sup>10</sup> : il a été conçu pour être multi-modèles et multi-méthodes, avec des mécanismes permettant l'ajout de

<sup>9</sup>Un *simulateur dynamique* est un logiciel qui permet de calculer les mouvements, les déformations, et les interactions entre des objets rigides ou déformables évoluant dans un environnement soumis à des lois physiques classiques.

<sup>10</sup>*Robot-Φ* est un prototype de recherche. Un brevet logiciel a été déposé et certifié par l'Agence pour la Protection des Programmes (Inter Deposit Digital Number IDDN.FR.001.160005.00.R.P.1996.R.P.1996.000.21000). Une version particulière de *Robot-Φ* à été intégrée dans la machine « Psy » d'effets spéciaux 3D, développée et commercialisée par la société Getris-Images GDI ; un « plugin » pour le système 3D Studio Max (de Kinetix) a également été développé dans ce cadre.

nouveaux modèles et méthodes. Dans sa version actuelle, *AlaDyn3D* permet de traiter les mouvements et interactions physiques d'objets rigides simples (chaque objet étant représenté par un modèle géométrique et par un modèle dynamique), et de simuler les mouvements/déformations et interactions de corps déformables surfaciques (*e.g.* drapeaux) ou volumiques (*e.g.* foie). Les interactions sont alors détectées grâce à un algorithme de calcul de distance (optimisé) ; elles sont ensuite modélisées à l'aide de liaisons visco-élastiques non linéaires et de modèles de collision de type « impulsif » et « pénalité ». L'intégration des équations de la dynamique peut-être fait par deux méthodes différents : une méthode de type explicite (Newton-Euler ou Runge-Kutta), et une méthode implicite à pas de temps fixe ou adaptatif. Afin de faciliter les échanges avec d'autres applications, *AlaDyn3D* est doté de filtres qui permettent d'une part d'importer des modèles 3D à partir de modeleurs du commerce (3D Studio Max), et d'autre part d'exporter des animations 3D dans un format standard (Vrml 2.0) ou des animations 2D sous forme d'une séquence d'images.

*AlaDyn3D* est implantée en C++/STL/OpenGL. Une version stable et documentée fonctionnant sous Silicon Graphics/Irix 6.x et PC/Linux est disponible en accès ftp (*cf.* <http://www.Inrialpes.fr/sharp/modelisation/AlaDyn>).

Une nouvelle version d'AlaDyn est en cours de développement et sera bientôt disponible. Alors que la version précédente était essentiellement centrée sur la notion de « simulation dynamique », la nouvelle version se présente comme une plateforme de développement d'applications de Réalité Virtuelle plus générique, dans laquelle les équations de la mécanique Newtonienne ne constituent plus la seule manière de générer le mouvement et les interactions entre objets. Les modèles géométriques ainsi que les méthodes de détection de collision et de rendu graphique ont été optimisées de manière à respecter des contraintes de temps-réel. De plus, d'autres modèles et algorithmes de base développés au sein du projet SHARP ont été intégrés, notamment certaines méthodes de planification et de contrôle du mouvement, ainsi que des modèles d'interaction avec une interface haptique.

## 5.2. OpenPI

Après avoir développé plusieurs moteurs d'inférence probabiliste et les avoir appliqués à différents problèmes : Systèmes de CAO et programmation des robots. Nous avons développé un système nommé OpenPI pour « Open Probabilistic Language ». OpenPI est un API (« application programming interface ») permettant de développer des rapidement des applications autour de l'inférence probabiliste. Les techniques utilisées sont celles déjà expérimentées dans le cadre de la programmation des robots. A cette date, OpenPI est l'outil de programmation probabiliste de l'équipe Sharp. De plus, OpenPI a été utilisé dans divers projets à l'extérieur de l'équipe. La version actuelle est nettement plus élaborée que celles que l'ont précédé tant au niveau de la vitesse de calcul qu'au niveau de ces fonctionnalités.

### 5.2.1. Développement de nouvelles fonctionnalités de OpenPI

Cette année nous avons largement amélioré les performances du moteur et nous avons augmenté ses fonctionnalités. Le logiciel est disponible pour les plateformes Linux, Unix, PC Windows (Visual C++), MacOS9, MacOSX et Irix . Une première version multi-processeurs a été développée en collaboration avec l'équipe APACHE. Elle donne des résultats très satisfaisants. D'autre part, nous avons mené une réflexion sur l'intégration d'outils d'apprentissage. Cette réflexion nous a conduit à une spécification qui sera implémentée en 2003.

Dans le cadre du contrat CNRS « Mathstic » nous avons implémenté un nouvel algorithme de simplification symbolique mis au point par le département de statistiques et de mathématiques appliquées de l'Université de Marne la Vallée.

Nous nous sommes intéressés à la conception et au développement d'un outil graphique et interactif de création et de manipulation de réseaux bayesiens. L'utilité d'un tel système est de pouvoir modéliser simplement et rapidement un problème à l'aide d'un réseau bayésien et de faire de l'inférence sur ce réseau. Cet outil est actuellement intégré dans le système d'OpenPI.

Le travail réalisé pendant cette période nous permet de disposer d'un système de spécification et d'inférence probabiliste nous permettant d'attaquer des applications concrètes.

### 5.2.2. Application de OpenPI

Plusieurs applications du moteur d'inférence probabiliste sont d'ailleurs en cours de réalisation dans un cadre contractuel.

1. Une application dans le domaine de la pose de prothèse du genoux avec la société AESCULAP. Dans cette application on utilise OpenPI pour faire du raisonnement géométrique sous incertitude dans le but de minimiser les erreurs de positionnement de la prothèse.
2. Une application financière avec la Société Générale Asset Management. Dans cette application on cherche à déterminer un portefeuille boursier en analysant plusieurs indicateurs financiers.
3. Une application dans le domaine de la maintenance préventive avec la société MGE-UPS. On cherche à se donner les moyens de faire du diagnostic ou de la maintenance préventive.

Enfin plusieurs études de faisabilité ont été réalisées grâce au moteur d'inférence.

1. Optimisation de l'offre (demande la société ISLANDIS) : dans cette application on utilise les bases de données d'un fournisseur pour connaître la probabilité pour une offre d'être acceptée.
2. Calcul de disponibilité (demande de la DER) : on calcule la probabilité pour un équipement d'être opérationnel à 10, 20... 100 %

## 6. Résultats nouveaux

### 6.1. Environnement dynamique partiellement connu

Pour des systèmes soumis à des contraintes cinématiques et dynamiques (tels que les véhicules) et évoluant à des vitesses élevées, l'évitement en ligne et en temps réel d'obstacles (éventuellement mobiles) non considérés pendant la phase de planification, est une tâche extrêmement complexe et néanmoins indispensable. Ce problème contient deux aspects : la connaissance partielle de l'environnement et la mobilité des obstacles. Cette section présente nos derniers travaux concernant la prise en compte de l'incomplétude de notre connaissance par un schéma de planification itérative *ITP* ainsi que la prise en compte explicite de la mobilité des obstacles à travers le concept des *V-obstacles*.

#### 6.1.1. États de collision inévitable

**Participants :** Stéphane Blondin, Thierry Fraichard, Alejandro Vargas Hernández.

Alors que la planification de chemins se concentre sur les aspects géométriques du problème (absence de collision entre le système robotique

et les obstacles), et se traite dans l'espace des configurations du système, la planification de trajectoires (avec prise en compte de la dynamique du système) s'aborde de façon privilégiée dans l'espace des états du système (paramètres de configuration et leurs dérivées).

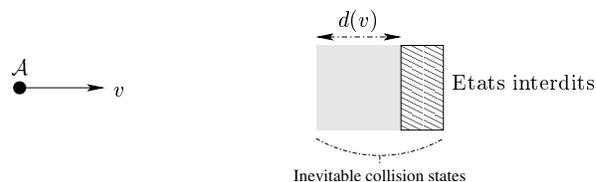


Figure 1. états interdits vs états de collision inévitable.

La notion de configuration interdite (dans laquelle le système est en collision avec un obstacle) est bien formalisée et se trouve au coeur de la problématique de la planification de chemins. De façon équivalente, on peut définir comme état interdit tout état dans lequel le système est en collision avec un obstacle ou dans lequel l'une des contraintes dynamiques du système est violée. Un exemple simple suffit pour se convaincre

qu'il peut être intéressant d'élargir cette définition : prenons le cas d'un robot ponctuel  $\mathcal{A}$  se déplaçant à direction constante et vitesse  $v$  variable en direction d'un mur (région hachurée de la Fig. 1). Selon la définition précédente, tous les états correspondants au mur sont interdits. En fait, en vertu de sa dynamique, il va falloir à  $\mathcal{A}$  une certaine distance  $d(v)$  pour s'arrêter. Il est alors plus intéressant de définir la notion d'*état de collision inévitable*, ie un état dans lequel, quel que soit la trajectoire future suivie par le système, une collision interviendra à un instant quelconque dans le futur.

Cette notion, évoquée au passage dans [61], n'a jamais été étudiée auparavant. Pourtant, c'est une notion qui nous semble pouvoir se révéler fructueuse (cf ses applications dans les sections 6.1.2 et 6.1.3). La principale difficulté réside dans la caractérisation de ces états de collision inévitable, difficulté que nous avons commencé à aborder cette année (Master de Alejandro Vargas Hernández [33], DEA de Stéphane Blondin [23]. séjour de Thierry Fraichard à Riken au Japon).

Nous nous sommes attachés à caractériser la *région de collision inévitable* induite par un obstacle donné (l'équivalent du  $C$ -obstacle de l'espace des configurations en quelque sorte). Nous avons procédé de manière incrémentale en caractérisant d'abord la région de collision inévitable pour un obstacle ponctuel  $p_i$  et pour une trajectoire  $\Gamma_j$  donnée du système  $\mathcal{A}$  (supposé ponctuel lui aussi dans un premier temps). Si l'on dénote par  $R_{\mathcal{A}}(p_i, \Gamma_j)$ , cette région alors la région de collision inévitable induite par  $p_i$  s'obtient par intersection sur l'ensemble des trajectoires possible :

$$R_{\mathcal{A}}(p_i) = \bigcap_{\Gamma_j} R_{\mathcal{A}}(p_i, \Gamma_j)$$

Pour un obstacle  $\mathcal{B}$  quelconque, la région de collision inévitable s'obtient par :

$$R_{\mathcal{A}}(\mathcal{B}) = \bigcap_{\Gamma_j} \bigcup_{p_i \in \mathcal{B}} R_{\mathcal{A}}(p_i, \Gamma_j) \quad (1)$$

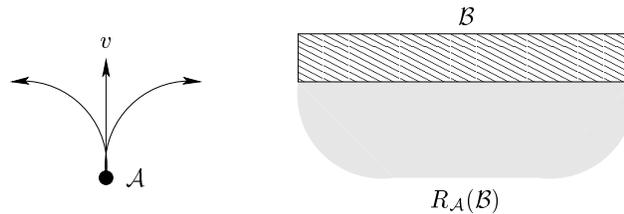


Figure 2. région de collision inévitable induite par  $\mathcal{B}$  pour un système  $\mathcal{A}$  de type voiture. La région grisée représente la région de collision inévitable pour une orientation donnée de  $\mathcal{A}$  (dirigée vers le haut).

L'équation (1) est la clé du calcul des régions de collision inévitable. La principale difficulté provient du fait que le nombre de trajectoires possibles est, en général, infini. Cela étant, il apparaît que les trajectoires « extrêmes » (eg freinage maximal), peuvent suffire, dans certains cas, pour caractériser ces régions. Ainsi, nous avons obtenu des premiers résultats intéressants dans le cas d'une voiture (contrôle de l'accélération et de la vitesse de braquage) et d'un obstacle fixe (Fig. 2). Nous avons aussi commencé à aborder le cas des obstacles mobiles avec là aussi des résultats intéressants. Ce travail doit être poursuivi et étendu à d'autres systèmes.

### 6.1.2. Contraintes perceptives et connaissances partielles

**Participants :** Stéphane Blondin, Thierry Fraichard.

En règle générale, un système robotique réel est équipé de capteurs extéroceptifs qui lui permettent de percevoir un environnement sur lequel il ne dispose, dans le meilleur des cas, que d'un modèle a priori partiel. Ainsi un robot de surveillance peut disposer d'une carte du bâtiment à surveiller (avec la position des murs et du gros mobilier) mais sans les positions exactes du petit mobilier ou de ses occupants). Dans une telle situation, l'approche standard en planification de mouvements « classique » (*ie* avec calcul a priori du mouvement à effectuer), consiste à raisonner sur le modèle disponible, à ignorer son caractère partiel et donc à déléguer au niveau exécution la gestion des problèmes liés à l'incomplétude du modèle (*eg* évitement des obstacles imprévus). Dans ce type de contexte, on doit être alors amené à se poser la question de la *sûreté* des mouvements calculés. Par *sûreté*, nous entendons la garantie que le système, lors de l'exécution du mouvement, ne se retrouve jamais dans une situation où il serait dans l'impossibilité d'éviter une collision.

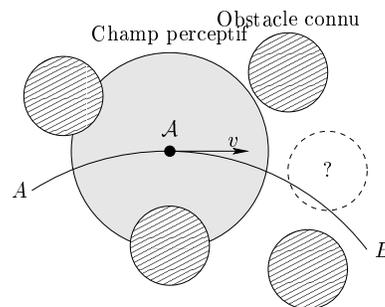


Figure 3. exemple de mouvement nominal sûr. La vitesse planifiée  $v$  doit permettre au système  $A$  d'éviter tout obstacle imprévu pouvant apparaître dans son champ perceptif.

Cette notion de sûreté dépend de deux facteurs : (a) les connaissances dont on dispose sur ce qui est inconnu, et (b) les capacités des capteurs. Prenons l'exemple illustré dans la Fig. 3. On a planifié un mouvement (*ie* position et vitesse), entre  $A$  et  $B$  pour le système  $A$ . Ce mouvement a été calculé afin d'éviter les obstacles fixes connus a priori (les disques hachurés). A un moment donné de l'exécution de ce mouvement, on peut supposer que  $A$  sera dans un certain état (position et vitesse). Ses capteurs lui permettront alors de percevoir une certaine partie de son environnement (le disque grisé). Si l'on sait que l'environnement est susceptible de contenir des obstacles fixes inconnus alors il est clair que  $v$ , la vitesse planifiée pour  $A$ , ne doit pas être trop importante afin de lui permettre de freiner si un obstacle imprévu devait apparaître dans son champ perceptif. Cette vitesse limite est déterminée par la portée des capteurs de  $A$  et sa dynamique propre. Même chose si l'on sait que l'environnement est susceptible de contenir des obstacles mobiles inconnus. Dans ce cas cependant, la vitesse limite dépend non seulement de la portée des capteurs et la dynamique de  $A$ , mais aussi de la dynamique des obstacles inconnus (*ie* leur vitesse supposée).

Cette notion de sûreté est en fait intimement liée à celle d'*états de collision inévitable* introduite dans la section 6.1.1. Afin d'être sûr, un mouvement planifié (qui n'est autre qu'une séquence continue d'états) ne doit comporter aucun état de collision inévitable (défini dans ce cas en fonction de la capacité des capteurs et des connaissances a priori faites sur l'inconnu).

Nous avons abordé ce problème de planification de mouvements sûrs cette année. Nous avons commencé par le cas obstacles fixes inconnus (DEA de Stéphane Blondin [23]). Après avoir caractérisé les états de collision inévitable dans ce cas-là, nous avons réalisé une première implantation d'un planificateur de mouvements sûrs basé sur un schéma général de planification stochastique [61]. Les premiers résultats obtenus sont encourageants (Fig. 4). L'extension au cas obstacles mobiles inconnus est prévue.

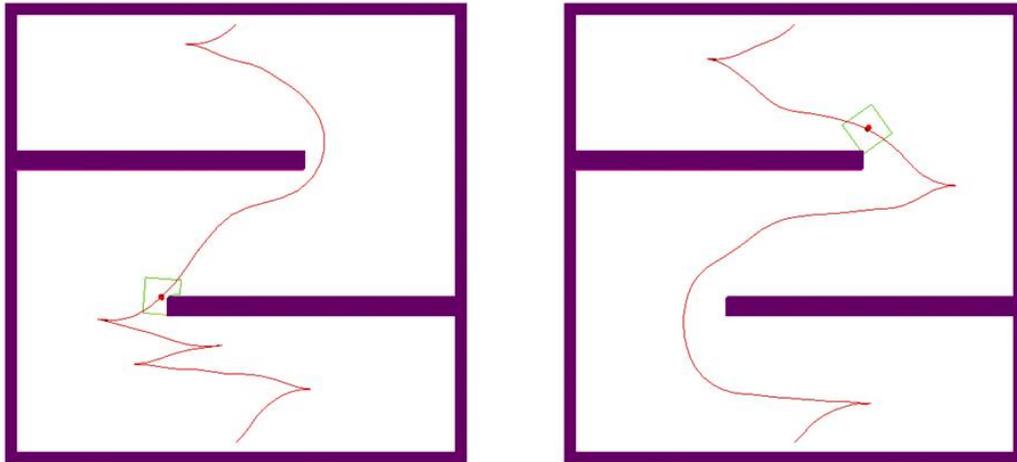


Figure 4. mouvement sans collision vs mouvement sûr. Le mouvement de droite prend en compte la possibilité de présence d'obstacles inconnus fixes et les capacités perceptives du système. le mouvement calculé ne rase pas les obstacles afin de permettre au système de percevoir les éventuels obstacles inconnu à temps pour les éviter [23].

### 6.1.3. Contraintes dynamiques et connaissances partielles

**Participants :** Thierry Fraichard, Alejandro Vargas Hernández.

En 2001, nous avons commencé à explorer le principe ITP (Iterative Trajectory Planning) pour traiter le problème de la planification de mouvements en présence d'obstacles mobiles dont le comportement futur n'est pas connu a priori. Planifier un mouvement en présence d'obstacles mobiles requiert la connaissance préalable des mouvements futurs de ces derniers. Faute de cette connaissance, on en est réduit à faire des prédictions. Dans ce contexte, la planification de mouvement se heurte à une double limitation : (a) limitation sur le temps disponible pour planifier un mouvement (fonction de la dynamique de l'environnement), et (b) limitation sur la durée de validité des mouvements planifiés (fonction de la durée de validité des prédictions). En d'autres termes, il faut pouvoir planifier vite et il n'est pas utile de planifier loin dans le futur.

ITP repose sur l'exploration de l'espace des état-temps du système. L'espace des états-temps, initialement introduit par Fraichard en 1992 [49][50], permet d'aborder la planification de mouvement en présence de contraintes dynamiques (dynamique du système et obstacles mobiles). Hsu *et al* à Stanford [55] ainsi que Frazzoli *et al* au MIT [53]) ont repris cette idée et utilisé des techniques stochastiques récentes pour explorer l'espace des états-temps. Hsu *et al* font l'hypothèse que de telles techniques permettent de recalculer une trajectoire jusqu'au but en un temps très faible. Ils résolvent donc le problème de l'incertitude liées aux prédictions par une replanification à une fréquence suffisamment élevée. Frazzoli *et al* sont plus modérés et envisagent que la contrainte au niveau du temps de réponse ne permettent pas de recalculer une trajectoire jusqu'au but.

Nous partageons l'avis de Frazzoli *et al*. Aussi avons-nous proposé ITP, un schéma de planification qui met en œuvre le double principe de la *planification itérative* et de la *planification sur un horizon temporel*. Ce schéma est explicité dans la Fig. 5. Le principe de fonctionnement de ITP est de restreindre la planification (et donc l'exploration de l'espace des états-temps) à l'horizon  $\delta_h$  afin garantir des temps de réponse compatibles avec  $\delta_c$ . ITP planifie donc itérativement des portions de trajectoires convergeant progressivement vers le but. Dans ce cadre, la principale difficulté à résoudre concerne la sûreté des trajectoires partielles ainsi calculées (en d'autres termes, quelle garantie a-t-on que cette trajectoire partielle ne va pas mettre le système dans un état

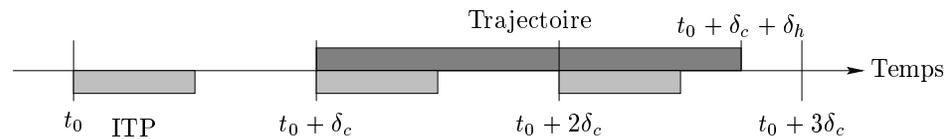


Figure 5. Planification itérative de trajectoire : le module ITP (Iterative Trajectory Planner) est appelé tous les  $\delta_c$ . A l'instant  $t_0$ , il calcule une trajectoire valide sur l'intervalle de temps  $[t_0 + \delta_c, t_0 + \delta_c + \delta_h]$ .  $\delta_h$  est l'horizon temporel de planification,  $\delta_h \geq \delta_c$ .

où une collision future sera inévitable ?). Frazzoli *et al* abordent ce point crucial en introduisant une simple notion de  $\tau$ -sûreté (un état-temps est  $\tau$ -sûr s'il est sans collision dans les  $\tau$  secondes qui suivent). Cette notion est très empirique et ne nous satisfait pas. Les concepts de « V-obstacles » (cf section 6.1.4) ou bien « d'états de collision inévitable » (cf section 6.1.1) nous semblent des réponses plus adaptées que nous comptons explorer.

Les travaux sur ce sujet de l'année 2002 ont porté sur la mise en place d'un environnement de test et d'un premier noyau ITP (Master de Alejandro Vargas Hernández [33]), ainsi que sur une première étude des états de collision inévitable dans le cas d'obstacles mobiles.

#### 6.1.4. Détermination des trajectoires instantanées de fuite dans un environnement dynamique

**Participants :** Frédéric Large, Sepanta Sekhavat.

Au cours des deux années précédentes, nous avons proposé le concept d'obstacle-vitesse. Les entrées de la méthode sont les trajectoires connues ou estimées des obstacles sur un horizon de temps donné, l'état courant du robot, ainsi que ses caractéristiques physiques permettant de déterminer le sous-espace de ses trajectoires de fuite potentielles. Le formalisme utilisé permet de représenter géométriquement cet espace et d'en déterminer les régions correspondant à des trajectoires en collision sur l'horizon de temps considéré. Nous avons obtenu une expression analytique de ces régions dans le cas de trajectoires de fuites rectilignes. La simplicité des calculs permet d'effectuer ces traitements en temps-réel et rend la méthode particulièrement adaptée à des environnements dynamiques nécessitant une réactivité importante du système contrôlé. Le lecteur peut se référer aux rapports d'activité de 2000 et de 2001 pour plus de détails sur la méthode.

Au cours de l'année 2002, nous avons essentiellement concentré nos efforts sur la mise en oeuvre des obstacles-vitesse dans des méthodes de (re)planification en ligne pour des robots mobiles. Le principe retenu consiste à développer des arbres de recherche à l'aide des obstacles-vitesse. Dans ce contexte, les problèmes qui se posent sont : 1) la façon dont sont choisies les branches à développer, 2) la façon dont sont créées les nouvelles branches, 3) la façon dont est propagée l'information lorsqu'une itération du contrôleur ne suffit pas.

1) La solution proposée pour répondre au premier problème consiste à privilégier les trajectoires permettant d'atteindre le but en un temps minimal. Pour cela, un poids  $P$  est associé à chaque noeud de l'arbre représentant un état du robot. Nous avons  $P(s) = t(s) + \tilde{t}(s)$  où  $t(s)$  est le temps nécessaire pour atteindre l'état  $s$ , et  $\tilde{t}(s)$  est le temps minimal estimé que le robot mettrait pour aller de  $s$  au but s'il n'y avait pas d'obstacles.

2) Deux critères ont été envisagés pour le second problème : la faisabilité de la trajectoire, ou sa globalité. Nous avons tout d'abord privilégié le premier critère pour que la trajectoire soit réalisable par le robot avant tout. Cela nous a amené à étendre l'arbre de dt en dt (dt étant la période d'itération du contrôleur du robot) □ Dans un deuxième temps, nous avons développé l'arbre de manière à utiliser les informations d'anticipation intrinsèques aux obstacles-vitesse, de manière à obtenir une solution approchée plus rapidement □ La trajectoire obtenue dans ce second cas ne donne qu'une estimation et doit être raffinée en fonction du temps restant disponible.

3) Le temps étant ici la ressource critique, nous avons proposé d'échelonner les traitements sur plusieurs itérations consécutives du contrôleur. Pour ce faire, nous vérifions à chaque itération que la trajectoire la meilleure est toujours valable et nous continuons de l'étendre. Si tel n'est pas le cas, le premier noeud en collision rencontré permet de supprimer le sous-arbre correspondant et de reprendre les traitements. La rapidité de la méthode permettant de garantir qu'à tout instant une trajectoire de fuite libre est disponible si elle existe, le système est maintenu en sécurité. Dans le pire des cas (premier noeud en collision), le système se comporte comme un module réactif. La figure 6 montre une trajectoire obtenue en 0.01s avec 20 obstacles dont 6 en mouvements, ainsi que son suivi. Il est à noter que cette trajectoire serait recalculée en temps-réel si l'environnement venait à changer par rapport à la prédiction initiale.

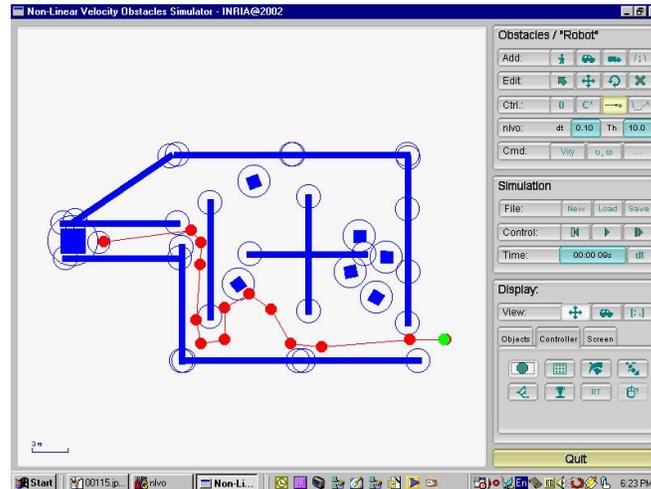


Figure 6. Calcul de trajectoire en ligne pour évitement d'obstacles mobiles.

## 6.2. Localisation et construction de carte (SLAM)

Les plateformes mobiles de l'Inria Rhône-Alpes ne possédaient jusqu'à cette année, que de moyens proprioceptifs de localisation (odométrie, gyroscope). Ces capteurs engendrent typiquement une erreur cumulative, non bornée sur la localisation. Dans le cadre d'expérimentations sur nos robots mobiles, il était donc nécessaire de développer un module de localisation extéroceptif fiable, robuste et aussi précis que possible.

Pour cela, en collaboration avec les Moyens Robotiques, nous avons choisi d'utiliser un système composé d'amers artificiels et d'un capteur télémétrique laser Sick (Fig. 7). Le capteur télémétrique Sick a été choisi pour sa portée supérieure à 30m, sa rapidité de réponse et surtout sa très grande précision. Les amers artificiels ont été choisis de façon à être aisément détectables et précisément localisables : il s'agit de cylindres particulièrement réfléchissants pour les rayons infra-rouge du Sick.

### 6.2.1. Invariants géométriques et théorie des graphes pour un SLAM robuste

**Participants :** Cédric Pradalier, Sepanta Sekhavat.

En 2001, nous avons développé un module de localisation rapide, sûr et efficace utilisant ces équipements. Les techniques utilisées avaient été présentées dans [71]. Le principal défaut de ce système venait de la nécessité de fournir une carte a priori des amers installés dans l'environnement. En général, il est difficile de donner une carte précise et pérenne de ces balises, nous avons donc étendu ce module de localisation de façon à ce qu'il soit capable de se construire une estimation de la carte des amers en temps réel pendant ses manoeuvres dans l'environnement. Le problème de localisation et construction de carte simultanés, appelé SLAM (Simultaneous Localization And Mapping) dans la littérature, est un problème actuellement très étudié par la communauté robotique.

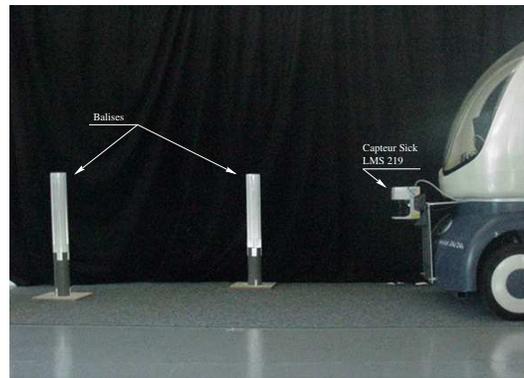


Figure 7. Le module de localisation extéroceptif composé d'un capteur laser Sick et des balises passives réfléchissantes.

Pour résoudre ce problème de manière robuste et efficace, nous avons combiné deux techniques récentes du domaine : la construction de carte par le « *Geometric Projection Filter* » et l'identification des amers observés grâce à la technique du graphe de correspondance. Par ailleurs, un important travail de conception a été mené de façon à développer des structures de données particulièrement efficaces permettant de traiter les données capteurs en temps réel. A titre d'exemple, le cycle complet d'identification des amers, localisation et construction de carte prend moins de 2 millisecondes sur un ordinateur de bureau.

Ces travaux ont donné lieu à une publication nominée pour le prix du meilleur papier dans la conférence internationale IROS 2002[19].

### 6.2.2. Mise en correspondance implicite pour la localisation

**Participant :** Cédric Pradalier.

Les travaux précédents sur la localisation ont montré l'importance de l'association de données dans ce processus. En 2002, nous avons donc fait un inventaire des méthodes existantes dans ce domaine capables de fournir des temps de réponse compatibles avec le temps réel. Il est apparu deux principales méthodes : l'utilisation des graphes de correspondances et l'utilisation d'une technique de programmation dynamique nommée JCBB « Joint Compatibility Branch and Bound ». Ces deux méthodes utilisent des techniques probabilistes pour déterminer une mise en correspondance explicite entre des observations et des références. Nous pensons que ces méthodes ne propagent pas suffisamment les incertitudes : en effet, à partir des observations, une mise en correspondance la plus probable possible est calculée, et acceptée comme certaine.

Nous avons proposé une méthode de localisation qui ne réalise pas cette mise en correspondance de façon explicite, mais calcule de façon implicite une distribution de probabilité sur les mises en correspondance possibles de façon à calculer une estimation de la position tenant compte au mieux de toutes les incertitudes du système. De plus, nous avons montré qu'il est possible d'exprimer les distributions impliquées de façon à obtenir des processus calculatoires très efficaces (moins de 10 millisecondes sur un ordinateur de bureau).

Ces méthodes de mises en correspondances ont été développées en utilisant les techniques d'inférence bayésienne étudiées au sein de l'équipe SHARP. Des tests sont en cours sur le robot réel et les premiers résultats sont suffisamment encourageants pour qu'un article soit en cours de rédaction.

## 6.3. Exécution du mouvement

### 6.3.1. Commande en boucle fermée du Cycab

**Participants :** Jorge Hermosillo, Sepanta Sekhavat.

#### 6.3.1.1. Antécédents

En 2001 nous nous sommes attaqués au problème de la planification de chemins pour les robots à double braquage coordonné et en particulier pour le robot Cycab (cf. [rapport d'activité 2001](#)). A cet effet, nous avons

exploité la propriété de *platitude différentielle* du système [48]. Le problème de planification de chemins en boucle ouverte ayant été résolu en 2001, cette année nos travaux se sont portés sur la problématique de la stabilisation du système autour d'une trajectoire de référence. Puisque le Cycab est un système plat, une solution naturelle a consisté à exploiter les propriétés de la sortie plate. En effet, il a été démontré qu'un système plat est un système linéarisable par bouclage endogène. Ce type de bouclage est appelé ainsi car il est « engendré » par les variables du système et leurs dérivées (aucune variable exogène au système est employée). Nous nous sommes donc attaqués à la recherche du bouclage linéarisant du Cycab et plus généralement de systèmes à double braquage coordonné.

### 6.3.1.2. Système commandable équivalent du Cycab

Dans un premier temps, nous nous sommes attaqués à un premier problème qui consiste à trouver le système équivalent du Cycab par bouclage endogène. Nous nous sommes donc concentrés sur une nouvelle étude de la dynamique de la sortie plate. Ceci nous a permis de trouver des nouvelles relations, complémentaires à celles rapportées précédemment (cf. [rapport d'activité 2000](#)). Ces relations établissent notamment le lien entre la commande du robot et la commande du système équivalent exprimé en coordonnées plates.

Plus précisément, le système à double braquage de type Cycab peut s'écrire comme un système non-linéaire sans dérive de la forme suivante :

$$\dot{x} = f(x, u), \quad (2)$$

où  $x = (x_r, y_r, \theta, \varphi)$  est le vecteur d'état et  $u = (v_r, \omega_\varphi)$  est la commande du robot. Nous avons établi (cf. [rapport d'activité 2000](#)) que la sortie plate  $y = (y_1, y_2)$  est un point  $H$  dans le repère du robot et dont les coordonnées sont fonction de  $x$  et de  $u$ .

Nous démontrons alors que le système équivalent dans l'espace plat peut s'écrire de la manière suivante :

$$\dot{y} = g(y, v), \quad (3)$$

où  $v = (v_H, \kappa)$  est la commande du système dans l'espace plat. Cette équivalence s'établit par l'obtention des nouvelles relations suivantes :

$$\begin{cases} \omega_\varphi &= \left(\frac{d\kappa}{dt}\right) \cdot \frac{1}{\mathbf{K}'(\varphi)} \\ v_F &= \frac{v_H - \omega_\varphi (\mathcal{M}' - \beta' \mathcal{N})}{\cos(\varphi - \beta - \pi) - \mathcal{N}\mathcal{F}} \end{cases} \quad (4)$$

$$\begin{cases} v_H &= v_F [\cos(\varphi - \beta - \pi) - \mathcal{N}\mathcal{F}] + \omega_\varphi [\mathcal{M}' - \beta' \mathcal{N}] \\ \kappa &= \mathbf{K}(\varphi) \end{cases} \quad (5)$$

où  $\mathbf{K}$ ,  $\beta$ ,  $\mathcal{N}$ ,  $\mathcal{F}$  sont des fonctions de la seule variable réelle  $\varphi$ .

Ces résultats nous permettent de compléter l'étude sur la platitude des robots à double braquage comme le Cycab, notamment du point de vue de l'équivalence. En effet, il est possible de démontrer l'équivalence entre les systèmes (2) et (3), en utilisant les relations (4) et (5) ci-dessus et celles des sorties plates. Une schématisation de cette relation d'équivalence est montrée dans la Figure 8. En outre, ces nouvelles relations nous ont permis de trouver la forme chaînée du Cycab, ce qui ouvre la possibilité d'explorer de nombreuses méthodes de stabilisation du système en un point.

Cette notion d'équivalence endogène nous a permis de poursuivre nos recherches dans le but de linéariser le système. Ceci afin d'utiliser des méthodes linéaires robustes et simples pour résoudre le problème du suivi d'une trajectoire de référence.

### 6.3.1.3. Linéarisation du Cycab

Nous avons abordé le problème de trouver le bouclage linéarisant en faisant appel à des méthodes de *linéarisation exacte* de systèmes non-linéaires. A cet effet, le calcul d'une extension par dérivation du système (3) donne un bouclage dynamique de la forme :

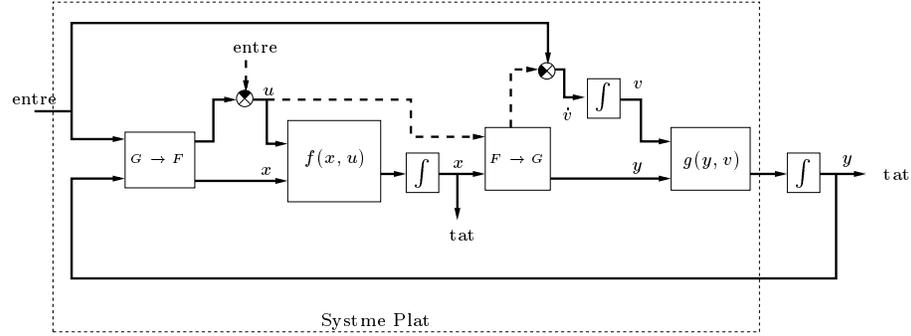


Figure 8. Équivalence endogène entre les systèmes (2) et (3). Les traits en continu montrent le flot « actif » lorsque la commande  $\dot{v}$  est présente, c'est à dire à dire que l'effet de  $u$  n'est pas considéré ce qui est indiqué par les traits discontinus.

$$\begin{pmatrix} \ddot{v}_H \\ \frac{d\kappa}{dt} \end{pmatrix} = \alpha(\chi) + \beta(\chi) \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}, \quad (6)$$

où  $\chi = \{x, v_H, \dot{v}_H\}$  est le vecteur d'état étendu et  $w = (w_1, w_2)$  est la nouvelle entrée auxiliaire.

Des résultats établis par ailleurs nous permettent de montrer que le système en boucle fermée ((2)-(6)) est difféomorphe au système linéaire commandable suivant, écrit sous forme canonique :

$$\dot{\tilde{y}} = \underbrace{\begin{bmatrix} \mathbf{A}_{c1} & 0 \\ 0 & \mathbf{A}_{c2} \end{bmatrix}}_{\mathbf{A}} \tilde{y} + \underbrace{\begin{bmatrix} \mathbf{B}_{c1} & 0 \\ 0 & \mathbf{B}_{c2} \end{bmatrix}}_{\mathbf{B}} \begin{pmatrix} y_1^{(3)} \\ y_2^{(3)} \end{pmatrix} \quad (7)$$

où  $(\cdot)^{(p)}$  représente la dérivée totale de  $(\cdot)$  d'ordre  $p$  et les matrices canoniques  $\mathbf{A}_{ci}$  et  $\mathbf{B}_{ci}$ ,  $i = 1, 2$  s'écrivent :

$$\mathbf{A}_{ci} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{B}_{ci} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$

où le nouvelle variable d'état est le vecteur  $\tilde{y} := (\tilde{y}_1, \tilde{y}_2)^T$ ,

avec  $\tilde{y}_i = (y^{(0)}, y^{(1)}, y^{(2)})^T$  pour  $i = 1, 2$ . En effet, (7) correspond à (3) étendu par deux intégrateurs.

La Figure 9 ci-dessous illustre ces résultats. La partie (a) de la figure montre l'extension dynamique de  $g(y, v)$  et le bouclage linéarisant du système. La partie (b) montre le système linéaire commandable équivalent (c'est à dire vu depuis la nouvelle entrée auxiliaire  $w$ ).

#### 6.3.1.4. Loi de commande pour système linéaire

En posant :

$$\begin{pmatrix} y_1^{(3)} \\ y_2^{(3)} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix},$$

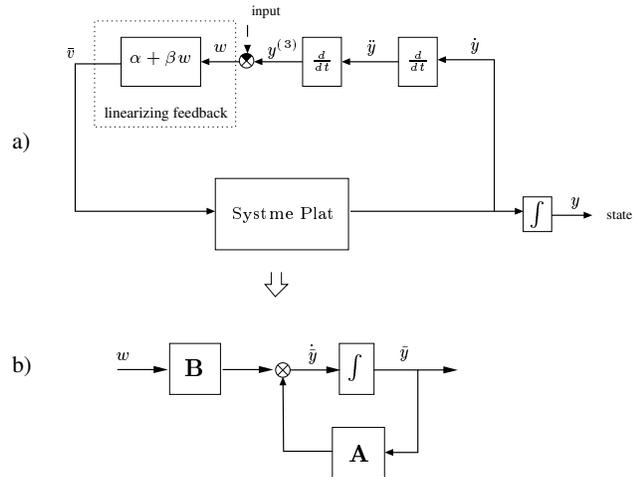


Figure 9. a) Bouclage linéarisant ; b) Système linéaire commandable équivalent sous forme canonique : (7).

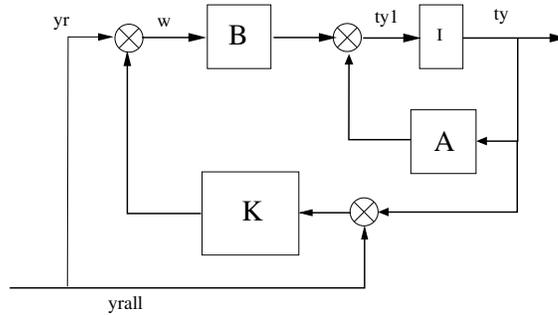


Figure 10. Loi de commande pour le système linéaire équivalent du Cycab.

le système (7) présente une structure linéaire que nous avons exploité pour stabiliser les trajectoires de (2)-(6). A cet effet nous avons utilisé des techniques classiques de la théorie de la commande de systèmes linéaires.

La Figure 10 illustre la loi de commande retenue.

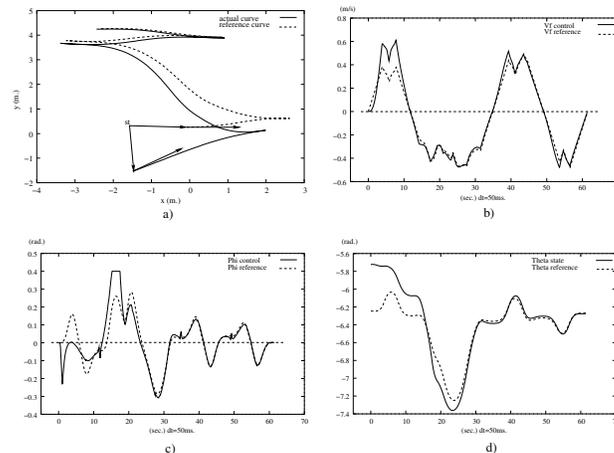


Figure 11. Résultats de simulation pour une manoeuvre de créneau, avec une erreur de départ de 1.5m en  $(x_F, y_F)$ ,  $\pi/6$  en  $\theta$  et 5% d'erreur sur le paramètre  $L$  (l'entre-axe) : a) Les courbes réelle et référence du point plat H ; b) La commande vitesse du robot ; c) La commande angle de braquage du robot- notez que cette commande est limitée aux bornes physiques réelles du robot ; d) L'orientation du robot pendant la manoeuvre.

Des résultats en simulation sont présentés dans la Figure 11.

### 6.3.2. Evitement d'obstacle par inférence bayésienne

**Participants :** Carla Koike, Cédric Pradalier, Sepanta Sekhavat.

Pour un robot destiné à évoluer en environnement urbain, il est impératif d'être capable d'éviter les collisions avec les piétons ou les autres véhicules. En 2002, nous avons donc spécifié et développé un module d'évitement d'obstacle utilisant les mesures de notre capteur laser et les concepts de la programmation bayésienne des robots développés dans l'équipe. Ne disposant que d'un seul capteur situé à l'avant du véhicule, notre module ne peut réagir aux obstacles présent à l'arrière du véhicule. Les contraintes à satisfaire sont les suivantes :

- Assurer une sécurité maximale dans un périmètre de sécurité prédéfini situé autour du capteur : tout objet situé dans cette zone doit entraîner un arrêt certain du véhicule. Les obstacles sont supposés se déplacer à vitesse réduite (piétons, véhicules manoeuvrants) sans chercher à entrer en collision avec le Cycab.
- La réaction à la présence d'un obstacle doit dépendre de la menace représenté par ce dernier : plus l'obstacle est loin, moins il va influencer la réaction du robot.
- Le module doit être capable de trouver des commandes efficaces et sûres, même en présence de plusieurs obstacles.
- Enfin, le module doit être capable de trouver un compromis efficace entre les commandes données par un système de contrôle de tâche de haut niveau et la sécurité du système.

Ce module d'évitement d'obstacle a été implémenté et testé sur le véhicule réel. A terme, il devrait être intégré comme module d'anti-collision sécuritaire dans notre architecture de commande. Le principe général est le suivant : chaque capteur fournit, sous la forme d'une distribution de probabilité, une description de l'ensemble des commandes motrices acceptables en fonction des obstacles qu'il détecte. Le contrôleur de haut niveau, quant à lui, spécifie sous forme d'une distribution de probabilité les commandes qui lui conviendraient

le mieux. Un processus d'inférence bayésienne est alors utilisé pour fusionner les distributions de probabilité issues des différentes sources. La commande qui convient probablement le mieux est alors sélectionnée pour être exécutée.

### 6.3.3. Expérimentations avec le Cycab

**Participants :** Jorge Hermosillo, Carla Koike, Frédéric Large, Cédric Pradalier, Sepanta Sekhavat, Michel Verlinden.

Une bonne partie de nos efforts dans l'année passée a été consacrée à la mise au point de notre plateforme expérimentale et de l'intégration de nos différents modules. Nous avons ainsi réalisé, avec l'aide des Moyens Robotiques, une démonstration mettant en avant les différents points développés durant les trois dernières années. Un scénario typique de nos démonstrations étant le suivant :

- Des balises réfléchissantes sont placées sur le parking de l'UR de Montbonnot, éventuellement après un pré-calcul de placement optimal des balises sur une carte élémentaire du parking vide. Voir le [rapport d'activité 2001](#), pour les techniques utilisées.
- Cycab est démarré sans aucune connaissance a priori et on lui fait exécuter un tour de piste en utilisant le module de conduite manuelle. Durant ce parcours, il construit automatiquement la carte des balises et celle des obstacles tout en se localisant par rapport aux balises détectées. Voir la section 6.2 pour nos travaux sur le SLAM.
- L'écran tactile du Cycab agissant comme interface, il présente à l'utilisateur une vision du parking construite à partir des tirs laser successifs fusionnés. Dans cette coupe horizontale de l'espace de travail (voir fig. 12), la position courante du Cycab est précisée. L'utilisateur peut ensuite grâce à l'écran tactile, choisir une configuration but.
- Le Cycab calcule alors automatiquement la trajectoire jusqu'à ce but. Le chemin est calculé grâce aux sorties linéarisantes obtenues pour le Cycab (Voir le [rapport d'activité 2001](#)). Une paramétrisation temporelle de ce chemin intégrant les contraintes de notre domaine de commande est alors calculée en utilisant des techniques d'optimisation. Nous avons ainsi obtenu l'unique planificateur (à notre connaissance) pour un robot général à double braquage.
- La loi de commande basée sur la platitude et présentée en section 6.3.1 est alors utilisée pour exécuter la trajectoire calculée. Il s'agit à notre connaissance du premier suivi de trajectoire en  $(x, y, \theta)$  pour un robot à double braquage.

Il nous reste encore à intégrer le module d'évitement d'obstacles imprévus. La figure 12, illustre une trajectoire calculée et exécutée par le Cycab dans l'environnement qu'il a automatiquement construit.

Cette démonstration extrêmement flexible est disponible à l'Inria Rhône-Alpes sur simple demande. Elle a aussi été exhibée à Sophia-Antipolis en novembre 2002 et sera présentée à Rocquencourt en février 2003.

## 6.4. Synthèse du mouvement virtuel et interaction 3D

**Participants :** Fouad Allaoui, Miriam Amavicza, Christian Laugier, César Mendoza, Roman Rodriguez, Kenneth Sundaraj.

### 6.4.1. Modèles Physiques

Au cours des années précédentes, nous avons mis l'accent sur l'utilisation de modèles de type *masses-ressort* linéaire et non-linéaire et différentes simulations médicales ont été développées sur cette base (cf. rapports d'activité antérieurs). Ces modèles présentent plusieurs avantages : leur facilité de mise en oeuvre, la rapidité de calcul qui permet de respecter les contraintes d'interaction, la possibilité de modéliser de comportements linéaire ou non linéaires. Néanmoins, leur manque de réalisme physique, la difficulté de réglage des paramètres du modèle et la dépendance topologique du modèle restent des limitations importantes de ce type d'approche. Cette année, afin de mieux prendre en compte le réalisme physique, nous nous sommes focalisés sur la mise en oeuvre de modèles de type éléments finis. Nous avons plus particulièrement travaillé sur une approche

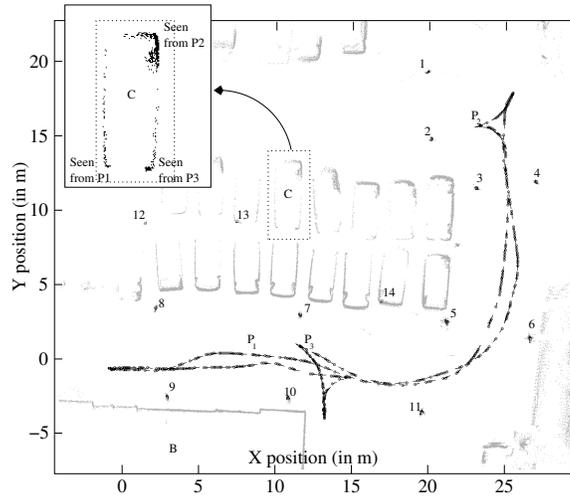


Figure 12. Carte de l'environnement construite automatiquement par le Cycab (les balises sont numérotées), ainsi que son estimation de la trajectoire calculée et effectuée.

développée par l'équipe en coopération avec l'Université de Stanford : le modèle LEM (Long Element Method).

#### 6.4.1.1. Modélisation par éléments finis explicites

Nous avons utilisé la méthode des éléments finis explicites (également connues comme masses-tenseurs) [70], [44], [46] pour notre simulateur médical. L'utilisation d'un type explicite des éléments finis nous a permis de développer, cette année, la découpe 3D. Nous avons modélisé les déformations à l'aide du tenseur de déformation de Green,  $\epsilon$ , représenté par une matrice symétrique  $3 \times 3$  : [54]

$$\epsilon_{ij} = \left( \frac{\partial \vec{x}}{\partial u_i} \frac{\partial \vec{x}}{\partial u_j} \right) - \delta_{ij}$$

où  $\delta_{ij}$  est la fonction de Kronecker,  $\vec{u} = [u, v, w]^T$  la position dans le repère de l'objet et  $\vec{x} = (\vec{u}) = [x, y, z]^T$  la fonction qui transforme la position dans le repère de l'objet dans le repère global. Pour la résolution numérique de la dynamique du modèle nous avons utilisé une technique d'intégration de type Newton-Euler.

Nos résultats sur la découpe se trouvent dans la section 6.4.2, et sur notre simulateur médical dans la section 6.4.3.

#### 6.4.1.2. Grande déformations avec le Modèle LEM.

Le modèle LEM (Long Element Method), a été conçu pour la simulation en temps-réel de tissus biologiques mous. Nous faisons alors l'hypothèse que les tissus peuvent être globalement considérés comme composés en majeure partie de liquide (ce qui est le cas en pratique). Dans ce modèle, le comportement mécanique de l'objet est caractérisé par les deux conditions aux limites suivantes :

- Le principe de Pascal qui établit que, dans un fluide incompressible en équilibre, les pressions se transmettent intégralement.
- L'incompressibilité du fluide implique que la conservation du volume doit être garantie lorsqu'il y a interaction externe et déformation de l'objet.

Le principe de cette approche, qui permet d'opérer avec un ordre de grandeur de moins que les méthodes d'éléments finis, est décrit dans le rapport d'activité 2000.

Cette année nous avons proposé une amélioration du modèle LEM afin de pouvoir traiter de grandes déformations. i.e. supérieures à 15 %. Le modèle de base que nous avons proposé l'année dernière était construit à partir d'un maillage triangulaire volumique ou surfacique. Cependant, certains éléments longs peuvent dégénérer et disparaître en fonction de la forme de l'objet modélisé (ou de l'objet déformé). Afin de prendre en compte ce problème (qui peut se produire lors de grandes déformations), nous avons introduit la notion de *discrétisation virtuelle* ou *élément long virtuel* dans les directions présentant ces pathologies ; cette approche permet de conserver à tout moment tous les degrés de liberté du système, et de traiter ainsi tout type de déformations.

En pratique, cela conduit à considérer, suivant le cas, deux composants de la déformation : (1) les effets de la pression interne du liquide dirigée normalement par rapport à la surface de l'objet et (2) les effets de la tension de surface.

Afin de mieux exprimer l'interaction de l'objet avec son environnement, nous avons également introduit les équations suivantes :

$$\begin{aligned}
 \text{Contrainte de contact} & : E_i \Delta L_i / L_i + P_{st}^i - \Delta P = \rho_i g \delta_i + P_{env} \\
 \text{Mouvement libre} & : E_i \Delta L_i / L_i + P_{st}^i - \Delta P = \rho_i g \delta_i \\
 \text{Points fixes} & : E_i = \infty \\
 \text{Application d'une force} & : \Delta L_i = C
 \end{aligned} \tag{8}$$

Dans la formulation initiale du modèle LEM, la loi de conservation du volume est la suivante :

$$\sum A_i \Delta L_i = 0$$

Pour des tissus mous au comportement très visqueux, cette équation est valide pour des déformations telle que celles dues à la gravité, au pincement, ou à la faible pression locale. Ceci n'est resté pas vrai dans le cas de grandes déformations, ce qui nous a conduit à utiliser le terme suivant :

$$\sum (A_i \Delta L_i + L_i \Delta A_i) = 0$$

Nous rappelons que la résolution du modèle LEM conduit à trouver le changement d'état,  $x$ , sous l'influence des forces externes (gravité, pression externe, etc.),  $b$ , exprime par :

$$x = K^{-1} b$$

où la matrice de rigidité  $K$  est supposée constante (ce qui n'est pas valide que pour des petites déformations). De grandes déformations conduisent donc à modifier la matrice de rigidité  $K$  au cours de la résolution du système. Comme les aires,  $A_i$ , apparaissent uniquement dans la dernière ligne de la matrice  $K$ , nous pouvons utiliser une méthode de mise à jour rapide comme la méthode de *Sherman-Morrison*. Une fois les matrices construites correctement, nous pouvons commencer la résolution numérique associée à l'intérieur de la boucle de simulation. Nous utilisons pour cela la méthode de résolution « *Quasi-Dynamique* » proposée l'année dernière par d'Aulignac [37]. Cette méthode suppose une combinaison d'une technique appelée *Successive Over Relaxation*, *SOR* et d'une technique d'itération *Newton Raphson*, (*NR*). Cela donne des résultats de simulation rapide et stables, même dans le cas de grandes déformations.

## 6.4.2. Interactions 3D

### 6.4.2.1. Détection de collision.

Nous avons cette année robustifié nos algorithmes, et développé une applet en Java 3D pour mettre en valeur et rendre disponible nos algorithmes. Il est maintenant possible de se connecter à notre site web pour tester et utiliser les algorithmes suivants :

- calcul de distance ;
- détection de collisions ;
- localisation de contact.

Ces algorithmes peuvent être appliqués sur n'importe quelle combinaison d'objets convexes ou concaves basés sur un maillage géométrique triangulaire.

Nous avons aussi commencé cette année à travailler sur les aspects suivants pour la détection de collisions d'objets déformables : les déformations globales, les auto-collisions, et les changements interactifs de voisinage dûs à une modification de topologie.

Quelques résultats préliminaires sur la déformation globale sont présentés dans l'applet java. Ces travaux se poursuivront en 2003.

### 6.4.2.2. Découpe 3D et interaction haptique.

Au cours des trois dernières années nous avons entamé une étude portant sur l'interaction haptique [76] et sur les modifications topologiques interactives des objets [41][67], telles que le déchirement et la découpe 2D. Pour la découpe, nous prenons en compte trois critères importants : (1) la préservation des principales caractéristiques physiques (volume, masse, élasticité), (2) la restitution d'effets visuels réalistes, (3)

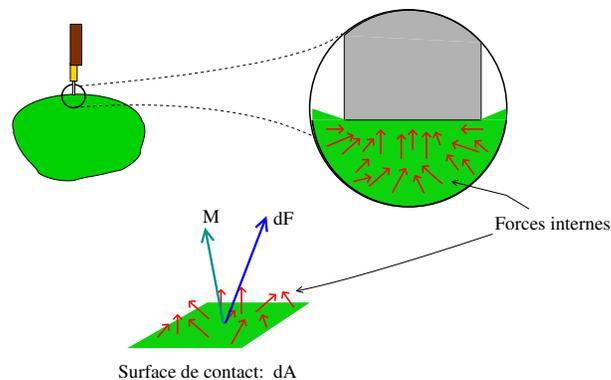


Figure 13. Forces intérieures pendant le contact entre l'outil de découpe et l'objet virtuel

le respect des contraintes de simulation en temps réel. Le travail antérieur portait sur des objets 2D (i.e. objets surfaciques comme des drapeaux, de la peau,...) représentés par des modèles de type *masse-ressort* [41].

Cette année, nous avons étendu ce principe aux cas 3D avec des modèles de type élément fini. Plus précisément, nous avons utilisé des *éléments finis explicites* [70], [44], [46] qui permettent de changer la topologie initiale de l'objet sans avoir recours à des étapes de pré-calcul comme cela est fait dans la littérature (e.g. [44]).

Notre approche vise à maîtriser la complexité du maillage (afin de conserver les performances interactives du système) en « *séparant* » les éléments du maillage (tétraèdres en l'occurrence),

plutôt qu'en les *divisant* ou *détruisant* comme cela est fait habituellement.

Les caractéristiques principales de notre algorithme de découpe 3D sont les suivants :

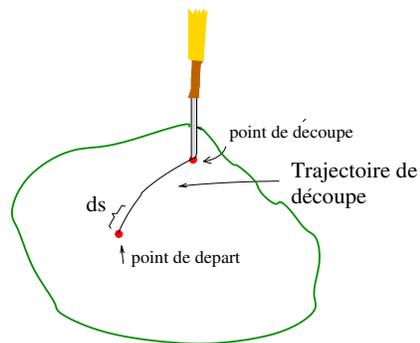


Figure 14. La contrainte de découpe cumulée

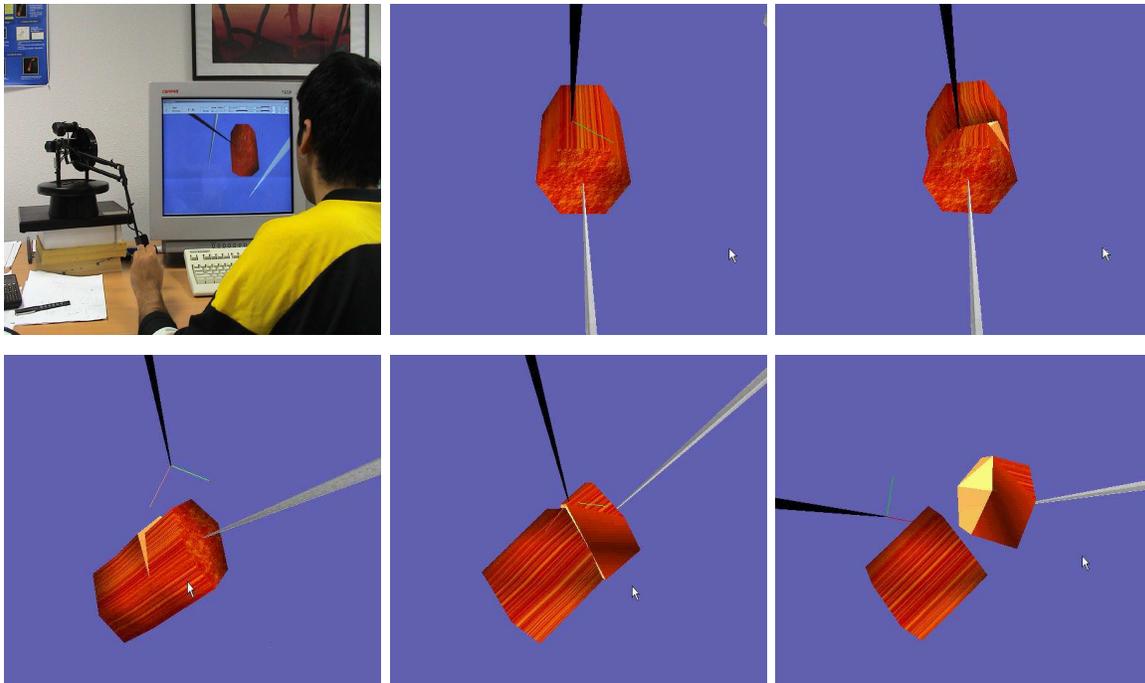


Figure 15. Découpe de tissu biologique en utilisant un dispositif de retour d'effort

- *Interprétation du mouvement et critères physiques* : Nous utilisons les mouvements de l'outil virtuel sur la géométrie de l'objet pour déterminer si l'utilisateur est en train de faire une découpe ou s'il est seulement en train de toucher l'objet. Ceci déterminera la *trajectoire de découpe*. Nous avons utilisé un critère de rupture pour déterminer quand le modèle cède à des essais contrainte/déformation imposés par l'outil virtuel. Ce critère est basé sur le tenseur des contraintes,  $\sigma$ . L'idée consiste à obtenir les contraintes sur la surface de contact entre l'outil et le tissu. Ces contraintes sont accumulées pendant le contact outil/objet, afin de trouver ce que nous appelons la *contrainte de découpe cumulée* :

$$K_c = \int_{s(0)}^{s(t)} \sigma_c(s) ds$$

où  $\sigma_c$  est la valeur de la contrainte à chaque pas de simulation,  $ds$  est le déplacement infinitésimale de l'outil sur la surface de l'objet entre les points  $s(0)$  et  $s(t)$ . Donc, le modèle « cède » (i.e. rupture locale) si la contrainte de découpe cumulée est plus grande que la contrainte propre du tissu biologique.

- *Séparation de tétraèdres* La trajectoire de découpe détermine quels tétraèdres doivent être séparés. Afin de mieux prendre en compte les changements de voisinage entre les différentes primitives (facettes, sommets, tétraèdres) dus à la modification de la topologie de l'objet, nous avons développé une structure de données, permettant de respecter les contraintes d'interactivité.
- *Remaillage local* Afin d'obtenir un profil de découpe en adéquation avec le déplacement de l'outil nous projetons les tétraèdres impliqués sur la trajectoire de découpe. Ceci entraîne une modification du volume original du tétraèdre et de sa matrice barycentrique<sup>11</sup>.
- *Retour haptique* Pour fournir à l'utilisateur une sensation beaucoup plus réaliste de toucher et de manipulation au cours de la simulation d'une opération de découpe, nous avons réalisé l'intégration du retour d'effort. Le modèle local utilisé pour le retour d'effort repose sur une modélisation de l'outil de découpe sous la forme d'un point. Afin d'améliorer le réalisme de la simulation, nous travaillons actuellement à intégrer au modèle local une représentation en volume de l'outil de découpe haptique.

La figure 15 montre quelques résultats pour la découpe de tissu biologique en utilisant des éléments finis explicites.

### 6.4.3. Simulateurs Médicaux

#### 6.4.3.1. Simulateur de Laparoscopie.

Dans le cadre du projet CAESARE, nous avons développé un simulateur de

laparoscopie. Les maillages des différents organes ont été obtenus grâce au projet Epidaure<sup>12</sup>. La tétraédricisation de ces maillages surfaciques a été faite en utilisant le mailleur volumique, GHS3D du projet Gamma<sup>13</sup>. Pour la déformation des organes, nous avons intégré ces modèles géométriques à notre simulateur Aladyn3D qui nous permet d'utiliser différents modèles physiques : masses-ressorts, éléments finis explicites avec un tenseur de Green-Lagrange et LEM et différents modes de résolution numérique (explicite et implicite). Nous avons aussi intégré un système à retour d'effort (PHANTOM).

La figure 16 montre les organes humains que nous avons simulés : la vésicule biliaire et le foie principalement. Chaque organe peut être simulé en utilisant différents types de modèle. Dans l'exemple de la figure, nous avons modélisé le foie en utilisant une approche à deux couches [46] : un maillage tétraédrique pour la simulation physique en utilisant des éléments finis explicites, avec un nombre très faible de tétraèdres (environ 45 tétraèdres), et un maillage surfacique du foie pour le rendu graphique, avec une très grande discrétisation (environ 3300 facettes). Nous avons connecté les deux maillages en utilisant une liaison rigide entre les

<sup>11</sup>La matrice barycentrique est une fonction linéaire qui lie le modèle continu au modèle discret.

<sup>12</sup><http://www-sop.inria.fr/epidaure>

<sup>13</sup><http://www-rocq.inria.fr/gamma/fra.htm>

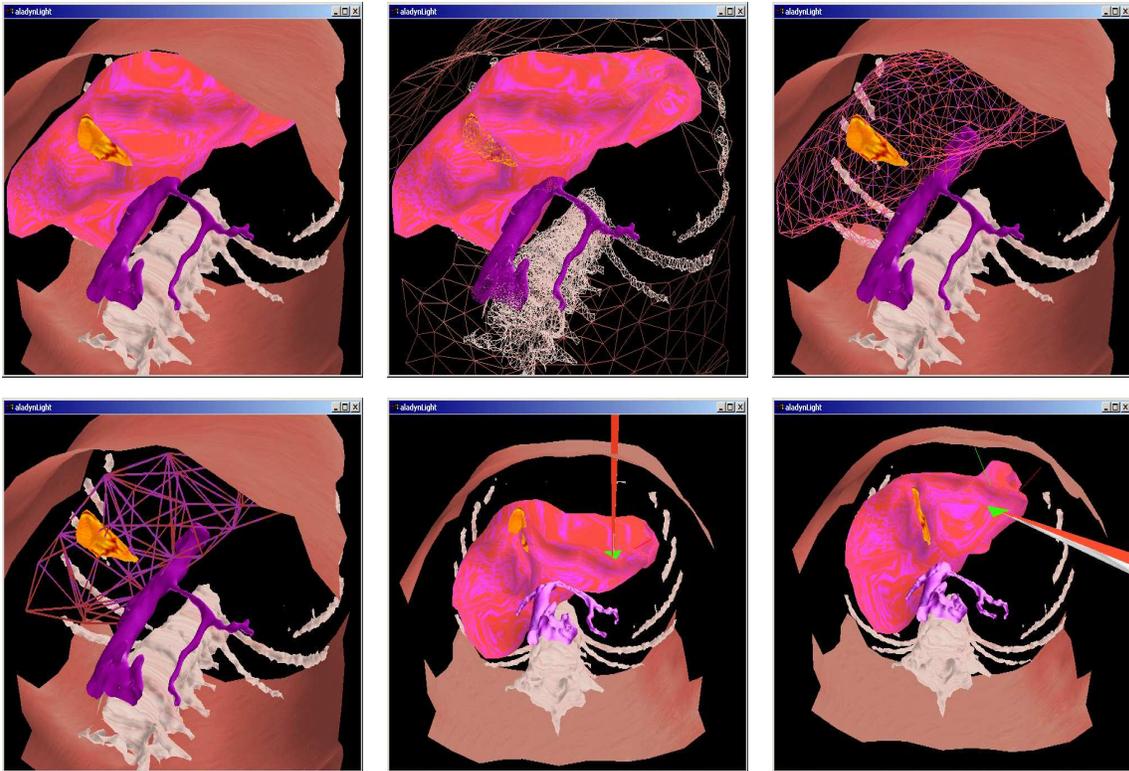


Figure 16. Exemple de la simulation d'un foie

sommets du maillage surfacique à l'extérieur et les points barycentriques des tétraèdres à l'intérieur. Cette approche nous a permis de simuler des déformations du foie à 50 Hz avec des maillages d'organes composés avec un nombre très importants d'éléments pour le rendu graphique, e.g. 20000 pour la colonne vertébrale, 1000 pour la peau, 3300 pour le foie, etc.

Enfin, nous avons couplé le simulateur au système à retour d'effort. L'interaction entre le système haptique et le simulateur donne des déformations du foie qualitativement satisfaisantes et des sensations de toucher des différents organes.

#### 6.4.3.2. Simulateur d'Arthroscopie

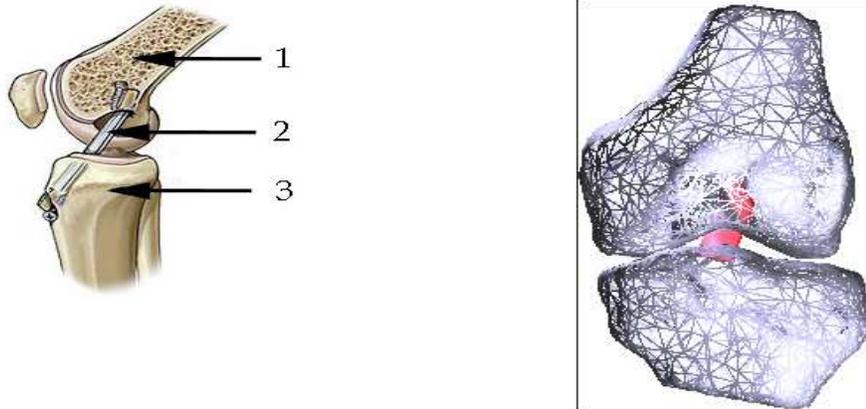


Figure 17. (à gauche) (1) Fémur (2) Greffon (3) Tibia (à droite) Le greffon virtuel

Cette année, nous avons commencé à développer un simulateur d'arthroscopie en collaboration avec la société Aesculap<sup>14</sup>. Le but du simulateur est d'aider le chirurgien pendant l'opération de remplacement du ligament ACL, voir figure 18. Le problème consiste à trouver le placement pour le



Figure 18. Modèle LEM du greffon

greffon en respectant des contraintes géométriques et physiques.

Afin de résoudre le problème physique du placement du ligament, nous avons modélisé le greffon du ACL en utilisant le modèle LEM. Cela nous permet de trouver, interactivement, la déformation du greffon en donnant des informations, *a priori*, sur le placement du greffon. Les premiers résultats sont visuellement satisfaisants, mais il reste encore un travail important d'identification des paramètres et d'expérimentation.

<sup>14</sup><http://www.aesculap.com>

#### 6.4.4. Synthèse du mouvement virtuel

**Participants :** Juan-Manuel Ahuactzin, Christian Laugier, Olivier Lebeltel, David Raulo, Ronan Lehy.

Il s'agit ici de définir des mécanismes aptes à engendrer automatiquement certaines classes de mouvements et de comportements pour un agent virtuel (ou « Robot virtuel »). On considère alors que l'agent virtuel est doté de capacités de mouvement autonome.

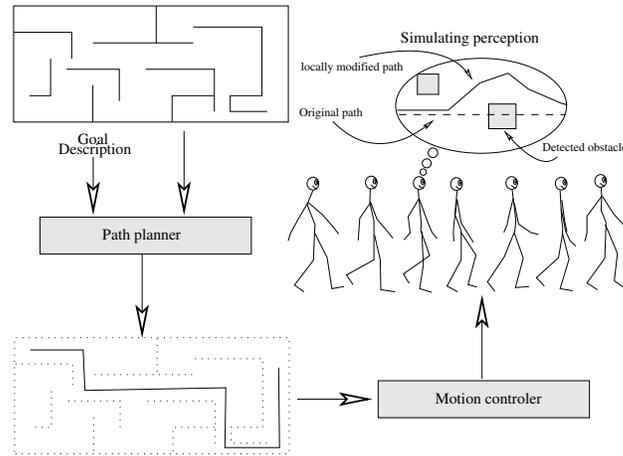


Figure 19. Structure décisionnelle pour un agent autonome virtuel.

Nous avons développé les principes de base d'une fonction de déplacement autonome en environnement virtuel. Cette fonction comporte un mécanisme de planification de trajectoires sans collision dérivé de la Robotique, un contrôleur de mouvement permettant de suivre de manière réactive une trajectoire donnée, des mécanismes de perception simplifiés, et enfin un système de programmation de comportements réactifs. Ce principe a été implanté dans notre plate-forme logicielle AlaDyn3D et utilisé pour donner à des personnages articulés certains comportements : évitement réactif d'obstacles mobiles, « footballeur », navigation dans un labyrinthe et dans une simulation de paysages naturel. Les personnages évoluent dans un environnement virtuel soumis aux lois de la Physique de Newton et pouvant comporter des corps déformables.

##### 6.4.4.1. Robot Virtuel.

Chaque personnage ou « Robot Virtuel » est représenté par une hiérarchie de chaînes cinématiques et de corps solides (cf. rapport d'activité 1999 : chaque segment correspond à un solide (e.g. l'avant-bras gauche), et chaque nœud à une articulation, pouvant comprendre chacune de 1 à 3 degrés de liberté (des rotations). A chaque instant, la *position* du personnage dans le monde virtuel est définie par le repère associé à la racine de l'arbre cinématique, et sa *posture* est décrite par l'ensemble des valeurs articulaires. Sa position relative par rapport aux obstacles et objets d'intérêt (e.g. le ballon) est alors calculée en fonction de sa perception de l'environnement (perception globale ou locale selon le cas). Afin d'accélérer le calcul des collisions, des boîtes englobantes ont été associées à cette structure (Fig. 20).

Le « contrôleur de mouvement » comporte deux fonctions complémentaires : la fonction de « locomotion » qui assure la coordination du mouvement des divers éléments du robot virtuel lors d'un déplacement réalisé dans un mode prédéfini, et la fonction de « suivi de chemin » qui génère les paramètres globaux (orientation, vitesse, accélération) du déplacement.

##### 6.4.4.2. Locomotion.

Pour la fonction de *locomotion* (marche ou course), nous avons utilisé une méthode classique consistant à coupler des données issues de capture du mouvement humain avec un algorithme d'interpolation par courbes de Bézier. Nous avons ajouté la possibilité d'importer des fichiers H-ANIM (format de description d'humanoïdes de synthèse dérivé du standard VRML97 pour le WEB), ceci permettant de doter le personnage

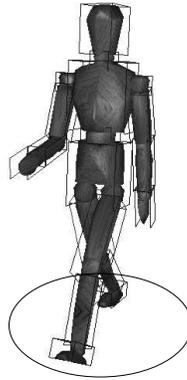


Figure 20. Modèle du Robot Virtuel (chaînes cinématiques, corps rigides articulés, boîtes englobantes pour le calcul des collisions.

de plusieurs « jeux » de locomotion. Les travaux des projets Bip et Imagis sur la simulation de locomotion bipède pourraient s'intégrer dans cette approche.

#### 6.4.4.3. Suivi de chemin.

En ce qui concerne le *suivi de chemin*, deux approches ont été abordées. Dans un premier temps nous avons utilisé un contrôleur de type PID auquel ont été ajoutées quelques contraintes simples sur les vitesses et accélérations linéaires et angulaires. Le chemin à suivre est fourni en temps réel par un planificateur de trajectoires (Fig. 19), un comportement de navigation dirigée vers un but en présence d'obstacles étant ainsi réalisé.

#### 6.4.4.4. Planification.

Dans l'implantation actuelle, le planificateur géométrique consiste en une version générique de l'algorithme ACA [65]. Ce planificateur est implanté orienté objet et permet de définir un problème de planification de trajectoires grâce à la dérivation de quatre classes appelées « *path driver* », « *goal predicate* », « *obstacle discriminator* » et « *local planner* ». Ces quatre classes contiennent des fonctions virtuelles à être écrites par l'utilisateur en fonction du problème de planification de trajectoires à résoudre. De cette façon, notre planificateur peut chercher une solution au problème sans s'occuper du type de trajectoires utilisées, du but à atteindre, des obstacles de l'environnement ou du planificateur local utilisé. L'idée générale du planificateur générique est de pouvoir écrire facilement et rapidement des planificateurs pour des tâches différentes ou similaires des robots virtuels. En plus, l'information obtenue lors de la résolution d'un problème de planification de trajectoires (*i.e.* un ensemble de balises dans l'espace de recherche du problème) pourrait être utilisée pour atteindre des buts de caractéristiques différentes.

#### 6.4.4.5. Programmation comportementale bayésienne.

Une seconde approche a été abordée cette année pour doter les « robots virtuels » de comportements autonomes : nous avons utilisé les méthodes de *Programmation Bayésienne* qui avaient été utilisées avec succès sur de petits robots mobiles (par O. Lebeltel dans le cadre de sa thèse [62]). Ces techniques permettent de décrire des comportements réactifs définis comme des *dépendances probabilistes* entre les variables sensorielles, motrices, plus éventuellement des variables internes des personnages. Il est ainsi possible de construire indépendamment plusieurs comportements simples (comme l'évitement réactif d'obstacles mobiles, le suivi de contour, etc...) puis de les *composer* hiérarchiquement ou les *séquence* en temps réel suivant ce qui est perçu de l'environnement. Les variables motrices peuvent être par exemple les vitesses linéaires et angulaires des personnages. Les *modes de locomotion* dont nous avons parlé précédemment peuvent être considérés comme des variables internes. Les variables sensorielles sont fournies par des « *capteurs virtuels* ». Nous avons notamment construit un capteur de proximité des obstacles comme un pré-traitement appliqué aux informations géométriques du monde virtuel, basé sur une variante de l'algorithme GJK de calcul de distances

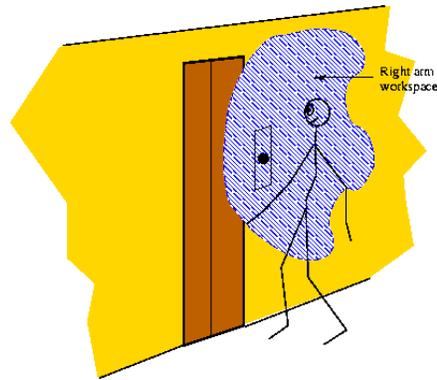


Figure 21. Espace de travail pour une tâche de cinématique inverse.

entre objets polyédriques. Cette méthode de spécification des comportements offre l'avantage de fonctionner dans des environnements dont la configuration n'est pas connue à priori (il n'est pas utile d'en construire préalablement une carte) ou lorsque celui-ci évolue dynamiquement au cours du temps.

#### 6.4.4.6. Résultats.

Notre première approche (planification + contrôleur de trajectoire sans l'aspect bayésien) a été validée expérimentalement avec un environnement dynamique simulé comportant des obstacles rigides (murs), des objets déformables (drapeaux que l'on peut déformer et/ou déchirer), un ballon dont le comportement est soumis aux lois de la physique de Newton, et un personnage articulé autonome ayant la capacité de « jouer au ballon » de manière automatique (et en temps réel). Ainsi, la version générique de l'ACA a déjà permis de planifier des trajectoires pour un personnage virtuel autonome qui cherche à atteindre des buts fixes ou mobiles, de même que pour un véhicule holonome. L'implantation réalisée est prévue pour le développement de classes dérivées, permettant résoudre en particulier les problèmes suivants : (1) amener une extrémité du personnage virtuel (*e.g.* sa main) vers une position déterminée (cinématique inverse, Fig. 21) (2) déplacer un objet d'une position à une autre (manipulation).

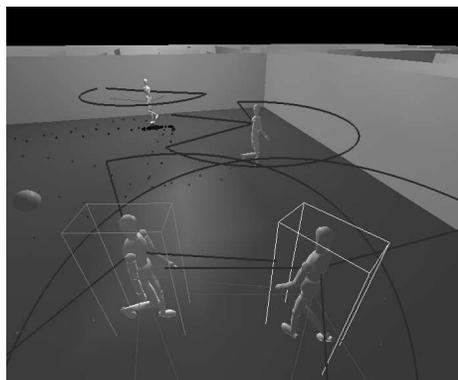


Figure 22. Comportement d'évitement réactif et capteurs virtuels.

D'autres expérimentations ont été menées pour tester l'approche Bayésienne. Dans le même type d'environnement que précédemment, nous avons pu contrôler simultanément une dizaine de personnages avec un même comportement réactif d'évitement d'obstacles (Fig. 22). Celui-ci s'est avéré peu coûteux en calculs et très robuste aux changements rapides de trajectoires des obstacles (par exemple lorsque l'expérimentateur prend le contrôle de l'un des personnages et tente d'intercepter un des autres). La même approche a également

été testée avec succès dans les environnements moins structurés que sont les simulations de paysage naturel (Fig. 23) ; ce travail a été fait en collaboration avec l'équipe Evasion du projet iMAGIS.



Figure 23. Population virtuelle dans un environnement naturel simulé.

Une méthode de combinaison des comportements à base de fusion probabiliste et de composition hiérarchique a été par la suite développée. Celle-ci réunit dans une architecture générique les aspects réactifs et délibératifs (planification) du contrôle de personnage et autorise la programmation de comportements complexes.

Parallèlement, l'approche de Programmation Comportementale Bayésienne a été appliquée au contrôle de « bots » de jeux vidéo (24) dans le cadre d'un sujet de DEA ([35]), intégrant en particulier une étude des avantages de la programmation inverse (consistant à donner les distributions de probabilités de variables sensorielles sachant les variables motrices) pour la spécification de comportements, et des notions d'apprentissage par l'exemple. Ce travail sera poursuivi en thèse, dans le cadre du projet ROBEA : « Modèles bayésiens pour la génération de mouvement », auquel participent aussi les équipes Evasion de iMAGIS et le projet Siames de l'IRISA.



Figure 24. Interface de pilotage d'un personnage de jeu vidéo.

## 6.5. Outils probabiliste pour la géométrie et la robotique

### 6.5.1. Programmation des robots

**Participants :** Pierre Bessière, Emmanuel Mazer, Kamel Mekhnacha, Olivier Lebeltel, Ruben Garcia, Carla Koike.

Ce travail complète nos premières études sur la programmation probabiliste des robots. Il vise à généraliser les résultats obtenus par notre équipe en la matière. De quels résultats s'agit-il ? Pierre Bessière propose un formalisme rigoureux pour l'inférence probabiliste, Olivier Lebeltel l'applique à la programmation d'un robot mobile et Kamel Mekhnacha à la réalisation d'un système de CAO probabiliste. De là trois questions principales : Le formalisme utilisé est-il applicable partout ? Peut-on étendre le concept de programmation probabiliste à un système complexe ? La modélisation probabiliste peut-elle être utilisée pour commander des bras manipulateurs ?

Sans pouvoir répondre de façon définitive à ces questions nous montrons que le raisonnement probabiliste peut être utilisé comme cadre général de programmation pour des bras manipulateurs équipés de capteurs de vision.

En effet le formalisme utilisé est suffisamment général : il peut s'étendre pour piloter des bras manipulateurs, il accommode aussi bien des capteurs de proximité que des systèmes de vision stéréoscopique, il permet de choisir entre des modèles empiriques et des modèles formels, il autorise plusieurs types de programmation : réactive planifiée et inverse.

Cette gamme de possibilités font de la programmation probabiliste un cadre potentiellement intéressant pour le développement de systèmes autonomes : c'est la thèse de Ruben Garcia.

Un programme de « pick and place » a été entièrement décrit de façon probabiliste. Le raisonnement probabiliste est utilisé pour résoudre les problèmes suivants :

1. autocalibration de caméras,
2. calibration de la paire stéréoscopique,
3. localisation d'un point par la paire stéréoscopique,
4. localisation d'un objet par la vision stéréoscopique,
5. calibration multirobot,
6. modèle de prise,
7. modèle de mouvement,
8. gestion de tâches,
9. intégration des modèles.

Ce programme a été implémenté sur la plate-forme « portique + bras » de l'Inria (Figure 26, 25).

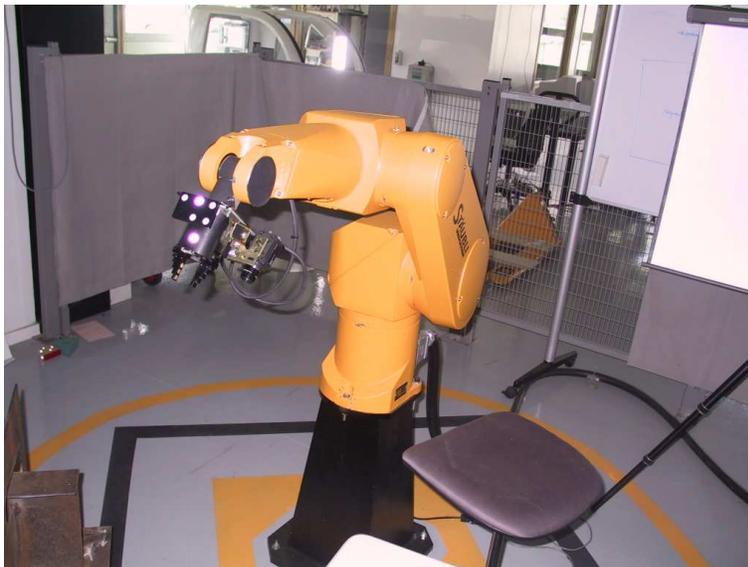


Figure 25. Bras robot stäubli RX90



Figure 26. Robot cartésien

### 6.5.2. Fusion de données sensorielles

**Participants :** Pierre Bessière, Christophe Coue, Thierry Fraichard.

Notre objectif est de construire une "carte" de l'environnement pour véhicule automobile. Il s'agit donc d'estimer la position et la vitesse des "obstacles" autour du véhicule, en se basant sur les mesures de différents capteurs (long-range radar, short-range radar, lidar). C'est donc un problème de tracking multi-cibles, multi-capteurs.

Nous avons utilisé une modélisation probabiliste des capteurs prenant en compte la probabilité de détection (voir [14][13]). Nos simulations illustrent l'importance de cette modélisation dans le cas de la fusion simple.

Notre contribution principale est le « tracking multi-cible » en utilisant la multi-modalité des distributions de probabilité. Le but est d'éviter la difficulté classique de l'association entre les pistes maintenues et les nouvelles mesures capteurs.

La Figure 27 montre une séquence extraite d'un tracking complet impliquant deux capteurs et quatre cibles. Elle illustre notamment l'importance de la phase de prédiction quand un capteur ne détecte pas une cible pourtant présente.

### 6.5.3. Construction de cartes sensori-motrices pour la mobilité

Nous souhaitons étendre le concept de carte utilisée en robotique. Ces nouvelles cartes ont pour ambition d'établir une relation entre les actions menées et les positions atteintes.

L'approche utilisée repose sur des méthodes probabilistes (e.g. localisation Markovienne), en les généralisant. Nous proposons un formalisme (thèse de Julien Diard), dans lequel une carte est représentée par une distribution de probabilités particulière (aspect déclaratif), qui permet de calculer des programmes pour se localiser, prédire, et générer du contrôle (aspect procédural). Le formalisme s'inspire également des théories biologiques (Berthoz, Franz, Kuipers) qui considèrent que les principes de construction incrémentale et de hiérarchie de cartes cognitives sont centraux.

Le formalisme inclut ainsi des opérateurs d'assemblage de modèles probabilistes, qui étendent nos travaux précédents sur la reconnaissance de comportements, et qui permettent la construction de hiérarchies de cartes. De premiers résultats expérimentaux ont été obtenus sur un robot Koala dans des situations de tâches simples ne nécessitant pas plus de deux niveaux de hiérarchie. Le travail actuel porte sur la généralisation du procédé, l'étude de l'articulation de ce type de modèle avec l'autre mode de représentation (de type « carte » obtenue

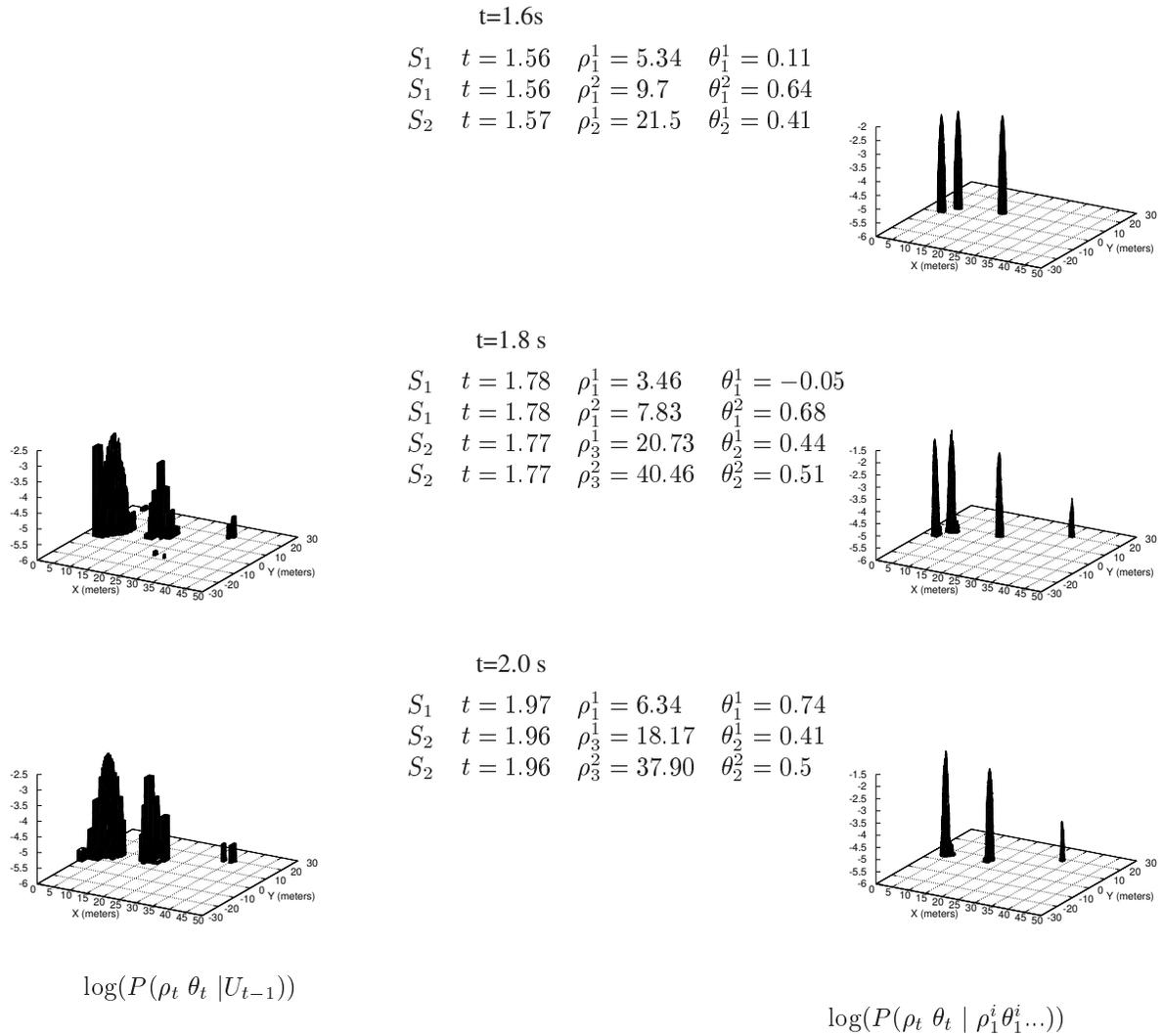


Figure 27. Séquence de suivi. A gauche, le résultat de la prédiction, à droite, le résultat de l'estimation.  
Au milieu, les mesures des différents capteurs.

par SMLAM), ainsi que l'étude de la plausibilité biologique de ce formalisme et des opérateurs d'assemblage proposés (collaboration INRIA/LPPA/NUS, projet de recherche post-doctoral de Julien Diard).

#### 6.5.4. Raisonnement probabiliste dans le domaine médical

OpenPL a été intégré comme composant d'un ancillaire informatisé d'aide à la pose de prothèse du genou (collaboration avec AESCULAP)(Fig. 28). L'intégration du moteur dans le système de navigation d'aide au chirurgien permet à celui-ci de tenir compte des incertitudes lors de son intervention. Cette intégration a donné lieu à des vérifications expérimentales basés sur des données post-opératoire mais n'est pour l'instant pas utilisée en conditions réelles.



Figure 28. Application de l'inférence probabiliste à la détermination de la cinématique du genou

Une deuxième collaboration pour la chirurgie de la prothèse de la hanche a été mise en place cette année. Des nombreux problèmes résultent d'un mauvais choix de paramètres dans cette chirurgie, par exemple, ceux liés au positionnement de la prothèse comme la luxation ou la perte de la prothèse et la diminution du degré de mouvement. Le but de ce projet est de reconstruire la cinématique de la hanche à partir des données mesurées sur le patient. L'inférence bayésienne dans ce cas permettra de déterminer les meilleurs paramètres de positionnement de la prothèse. Actuellement le projet est en étape d'acquisition de données obtenues à partir d'un logiciel qui récupère les points palpables du patient. À ces données s'ajouteront des mesures anatomiques de la hanche qui permettront de déterminer les paramètres cinématiques de positionnement.

## 7. Contrats industriels

### 7.1. Actions de développement Praxitèle et LaRA

Le projet Sharp a participé activement à l'action de développement Praxitèle (<http://www-rocq.inria.fr/praxitele/>) [1994-1997] portant sur le concept du transport urbain public et individuel. Nous participons maintenant à une nouvelle action de développement baptisée LaRA (<http://www.lara.prd.fr>) sur le thème de la « route automatisée », qui implique des organismes de recherche (Inrets, Inria, LCPC, ENCPC, ENSMP, ENST) et des partenaires industriels (Thomson-Detexsys, EDF, PSA, Renault, etc.). Dans LaRA, tout comme dans Praxitele, la contribution de Sharp porte sur l'aspect « conduite automatique ».

### 7.2. Projet Priamm « Studio virtuel »

Le projet Sharp participe activement au programme PRIAMM<sup>15</sup> du ministère de l'industrie dans le cadre du projet VISTÉO<sup>16</sup>. Ce projet a commencé en mai 2000 pour une durée de 24 mois. L'objectif de ce projet est de développer un ensemble d'outils logiciels permettant la mise en œuvre de studios virtuels physiquement réalistes incluant des interactions « physiques » entre le personnage réel et le studio virtuel dans lequel il est

<sup>15</sup>Programme pour l'Innovation dans l'Audiovisuel et le Multimédia.

<sup>16</sup>Système temps-réel intégrant vidéos et mondes virtuels

plongé. Ce projet, pour lequel la société Getris Image GDI est maître d'œuvre, implique les projets Sharp et Movi de l'Inria.

### 7.3. Robosoft

Dans le cadre de nos travaux dans le domaine du transport urbain public et individuel, nous avons initialisé un accord portant sur l'industrialisation de véhicules de type Cycab, dont trois exemplaires ont été vendus à NTU (Singapour). Depuis, Robosoft a été choisi par l'Inria, suite à un appel d'offre, pour industrialiser les véhicules Cycab.

### 7.4. CyberCars

Projet européen IST-2000-28487 CyberCars, "Cybernetic Cars for a New Transportation System in the Cities", (<http://www.cybercars.org>) [Août 2001-Juillet 2004]. Ce projet porte sur le développement et l'expérimentation de nouvelles techniques de transport. Ces techniques sont basées sur l'utilisation de véhicules individuels automatiques qui circulent dans les rues des villes ou dans des sites privés comme alternative à l'usage de la voiture privée et en complément avec les transports publics. Le consortium CyberCars comporte 14 partenaires venant de l'industrie et de la recherche publique. La contribution de Sharp à CyberCars porte sur les aspects automation de la conduite.

### 7.5. Carsense

Projet européen IST 1999-12224 CarSense, « Sensing of Car Environment at Low Speed Driving » (<http://www.carsense.org>) [janvier 2000-décembre 2002]. Ce projet européen porte sur la mise au point d'un système de perception multi-capteurs destiné à être embarqué sur un véhicule automobile. Il implique un certain nombre d'industriels européens du secteur automobile (Renault, BMW, Lucas Varsity, Thomson Detexys, Ibeo, etc.) et des instituts de recherche (Inria, Inrets, Livic). Dans CarSense, Sharp traite la partie fusion de données.

### 7.6. Arcos 2003

Action fédérative élaborée dans le cadre du PREDIT, le projet ARCOS 2003 a l'ambition de réussir une approche globale du système « véhicule-conducteur-infrastructure », en vue de contribuer à l'amélioration de la sécurité routière (avec un objectif de réduction de 30% des accidents !). L'équipe Sharp intervient dans le thème 3 de ce projet qui a pour titre la « Synthèse de l'information et élaboration de la commande ». Ce grand projet inclue de nombreux laboratoires français travaillant dans le domaine des véhicules et de la route (ENSMP, INRETS/LIVIC, SUPELEC, UTC...) ainsi que les plus grands constructeurs automobiles français (PSA et Renault).

### 7.7. Biba

Le projet BIBA pour Bayesian Inspired Brain and Artefacts (<http://www-biba.inrialpes.fr>) s'inscrit dans le programme IST (Information Society Technologies) de la Communauté Européenne. Les 2 objectifs du projet sont d'utiliser le raisonnement probabiliste pour, d'une part, reconsidérer la méthodologie, les modèles, les algorithmes et les techniques d'élaboration d'artefacts non-virtuels et, d'autre part, comprendre comment les systèmes biologiques utilisent la logique probabiliste dans leur environnement. Notre équipe est coordinatrice du projet. Nos partenaires sont University College of London (Département de physiologie), University College of Cambridge (Laboratoire de physiologie), Collège de France (Laboratoire de physiologie de la perception et de l'Action), l'Ecole polytechnique fédérale de Lausanne (Laboratoire des systèmes autonomes) et le Massachusetts Institute of Technology (Laboratoire des systèmes non-linéaires)

Le projet s'organise autour de 3 axes de recherche : (1) base neurale de l'inférence probabiliste, (2) nouveaux modèles et algorithmes probabilistes pour la perception et l'action et (3) nouvelles méthodologie et techniques pour la conception et le développement d'artefacts.

Au cours de cette première année du projet BIBA (nov 2001 - nov 2002), nous avons organisé 2 écoles internes qui ont permis de mettre en valeur la base pluridisciplinaire du projet. Une méthode d'évaluation des résultats a été définie, ainsi qu'une stratégie de diffusion de l'information. Cette première année a par ailleurs permis de définir les caractéristiques du premier robot expérimental. Enfin, la rédaction d'un état de l'art sur chaque axe de recherche a été réalisé.

## 8. Actions régionales, nationales et internationales

### 8.1. Actions régionales et nationales

#### 8.1.1. *Projet Gravir-Leibniz.*

Ce projet sur la prise en compte de l'incertitude (avec le laboratoire Leibniz) a reçu le label Imag pour quatre ans et obtenu pour cela un soutien financier de l'Imag jusqu'en 2002.

#### 8.1.2. *Pôles CNRS Structures et Machines Intelligentes.*

Nous sommes impliqués dans quatre de ces pôles : « Robots autonomes d'intervention sur site non-coopératif » (coordinateur : Laas à Toulouse), « Coopération homme-machine pour l'aide à la conduite automobile » (coordinateur : laboratoire Heudiasyc à Compiègne), « Aide au geste chirurgical » (coordinateur : laboratoire TIMC à Grenoble) et « Téléopération et Réalité Virtuelle » (coordinateur : Commissariat à l'énergie Atomique).

#### 8.1.3. *Projet Caesare.*

Ce projet d'une durée de deux ans a été soutenu par le ministère de la Recherche (programme « Télémédecine ») jusqu'en 2002. Outre Sharp, il impliquait cinq projets Inria (Epidaure, Sinus, Imagis, Mostra et M3n), l'Ircad<sup>17</sup> de Strasbourg et la société ESI. L'objectif de ce projet était de proposer des méthodes et algorithmes pour modéliser de la façon la plus précise possible, tout en garantissant une simulation interactive, des opérations de découpe et de déchirement sur des modèles dynamiques. L'implantation des méthodes est terminée pour les objets surfaciques et en cours de finition pour les corps volumiques.

#### 8.1.4. *Projet Robea ParkNav, « Interprétation de scènes dynamiques complexes et planification réactive de mouvement ».*

Ce projet Robea qui vient de démarrer [octobre 2002-septembre 2005], compte cinq partenaires : Movi et Sharp de l'Inria Rhône-Alpes, Prima de l'UMR Gravir, Vista de l'UMR Irisa et le groupe RIA du Laas-CNRS. Son objectif est l'automatisation de la conduite d'un véhicule évoluant au milieu d'obstacles mobiles (principalement d'autres véhicules en conduite manuelle ou automatique) dans un site propre équipé d'un système de perception à base de caméras. La réalisation de cet objectif passe par la résolution de deux problèmes complémentaires scientifiquement difficiles : (1) l'interprétation de scènes dynamiques complexes à l'aide de caméras, et (2) la planification réactive de mouvement en environnement dynamique.

#### 8.1.5. *Projet Robea « Commande de manipulateur mobiles non-holonomes ».*

Ce projet Robea d'une durée de 2 ans (prolongeable d'un an) a démarré en début 2002 et compte quatre partenaires : Icare de l'Inria Sophia, Sharp de l'Inria Rhône-Alpes, le groupe RIA du LAAS-CNRS et le LGP de l'ENI de Tarbes. Le thème de recherche sous-jacent de ce projet est celui de la coordination des mouvements respectifs d'un bras manipulateur (sous-système holonome) et d'une base mobile porteuse (sous-système non-holonome). L'objectif est l'utilisation de nouvelles techniques de commande par retour d'état non-linéaires pour la commande de robots non-holonomes.

#### 8.1.6. *Projet Robea « Un androïde doué de parole : Robotique Cognitive de la Parole et du Langage ».*

Ce projet Robea compte trois partenaires : ICP, INRIA et University of Texas at Austin. Il porte sur l'étude et la modélisation des mécanismes de perception, de production et d'apprentissage, afin de mieux percer les secrets de la parole et du langage, et d'ouvrir de nouvelles voies pour leur traitement automatique.

<sup>17</sup>Institut de la Recherche contre le Cancer de l'Appareil Digestif.

### 8.1.7. *Projet Robea « Modèles bayésiens pour la génération de mouvement ».*

La Réalité Virtuelle propose de nouvelles formes d'interaction entre l'homme et les systèmes informatiques. Les mondes synthétiques créés et gérés par ces systèmes peuvent être peuplés à la fois par des acteurs humains et par des acteurs de synthèse pilotés par des ordinateurs. L'approche que nous proposons d'étudier consiste à doter les entités virtuelles immergées dans ces environnements d'une autonomie de mouvement et d'action, ainsi que d'une réelle capacité de réaction et d'adaptation à certaines situations. Afin d'atteindre ces objectifs, nous réunissons dans ce projet des équipes spécialistes des différents domaines concernés : SHARP (INRIA Rhône-Alpes et GRAVIR) pour son expertise sur la Planification de Mouvement, l'Inférence Bayésienne et son expérience en Robotique ; SIAMES (IRISA) pour l'animation de personnages virtuels et la modélisation des environnements dans lesquels ils évoluent ; et EVASION (INRIA Rhône-Alpes et GRAVIR) pour son savoir-faire en synthèse de scènes naturelles animées et interactives.

## 8.2. Relations bilatérales internationales

### 8.2.1. *Russie et Asie Centrale*

#### 8.2.1.1. *Institut Franco-Russe Liapunov.*

L'institut Liapunov qui est un institut commun de l'Inria et de l'Académie des Sciences de Russie, nous a accordé son soutien pour une collaboration de deux ans avec l'université de St. Petersburg (équipe du Prof. V. A. Yakubovich) sur le thème *Controller design for autonomous car steering* [juillet 2000 - juin 2002]. Ce projet est consacré au problème de la conception d'un contrôleur qui assure le mouvement stable et robuste d'un véhicule le long d'un chemin imposé et inconnu d'avance.

### 8.2.2. *Pacifique et Asie du Sud*

#### 8.2.2.1. *Collaboration avec le Japon.*

Depuis octobre 1997, une collaboration dans le domaine des systèmes multi-robots a démarré avec l'institut Riken de Tokyo. Des visites croisées ont eu lieu au cours des trois dernières années. Cette année, Thierry Fraichard a passé 4 mois à l'institut Riken et Igor Paromtchik a passé 3 mois à l'INRIA Rhône-Alpes.

#### 8.2.2.2. *Collaboration avec Singapour.*

Le laboratoire commun, baptisé *Intelligent Vehicle Lab*, entre l'Université Technologique Nanyang de Singapour (NTU) et l'Inria a été inauguré en novembre 1998, dans le cadre de la collaboration scientifique sur le thème des véhicules autonomes signée par les présidents des deux institutions 2 mois plus tôt.

Depuis, cette collaboration s'est concrétisée : (a) par un nombre importants de visites et séjours croisés (d'une semaine à plusieurs mois) de chercheurs, (b) par l'accueil d'étudiants singapouriens à l'Inria (niveau undergraduate et graduate), et (c) par l'organisation en novembre 1999 à Singapour d'un symposium sur le transport intelligent, et d'un séminaire sur la robotique et l'imagerie médicale en décembre 2002.

### 8.2.3. *Amérique du Nord*

#### 8.2.3.1. *Collaboration avec l'université de Vancouver (Canada).*

La collaboration sur les aspects manipulation dextre, commencée lors de la visite du professeur K. Gupta à Grenoble en 1995, s'est poursuivie par plusieurs séjours longs de Moëz Cherif, et par des visites croisées. K. Gupta était en visite en juin 2000 et plusieurs publications communes sont apparues en 2000 et 2001.

#### 8.2.3.2. *Collaboration avec l'université de Californie, Berkeley (USA).*

Dans le cadre du programme France-Berkeley, une collaboration d'un an sur le thème de la simulation dynamique avait été menée en 2000-2001. Cette collaboration s'est poursuivie cette année dans le cadre d'un nouveau projet sur les simulateurs de chirurgie soutenu par le fonds France-Berkeley (avec l'équipe de S. Sastry). Plusieurs séjours croisés de chercheurs ont eu lieu.

#### 8.2.3.3. *Collaboration avec l'université de Stanford.*

L'étude de retour d'effort dans des environnements virtuels et les déformations élastiques non-linéaires ont été les sujets de collaboration avec le Centre de Technologie Avancée en Chirurgie de l'Université de Stanford. Plusieurs séjours croisés de chercheurs ont eu lieu.

## 8.2.4. Amérique du Sud et Amérique Centrale

### 8.2.4.1. Collaboration avec le Mexique.

Le réseau thématique Image & Robotique est un réseau qui a été mis en place à la suite des journées Franco-Mexicaines d'informatique et de d'automatique (JFMIA'99) qui ont eu lieu à Xalapa au Mexique en Mars 1999. Le principal objectif du réseaux Image & Robotique est de promouvoir et d'accroître les coopérations Franco-Mexicaines dans le domaine de l'Image et de la Robotique, et ceci au niveau des aspects scientifique, académique, et industriel. Ce réseau a été mis en place de manière effective en 2000. Il supporte depuis une école d'été annuelle, des échanges d'étudiants et des visites croisées.

Démarrage du projet NavDyn entre Sharp et le « Centro de Sistemas Inteligente », Itesm, campus Monterrey [Octobre 2002-Septembre 2004]. Ce projet soutenu par le Laboratoire Franco-Mexicain d'Informatique (LAFMI) porte sur la navigation de véhicules em environnement dynamique.

### 8.2.4.2. Coopération avec l'université de Brasilia (Brésil).

Dans le cadre du programme de coopération Inria-Brésil, nous avons collaboré jusqu'en 2001 avec l'université de Brasilia sur les thèmes robotique mobile et programmation des robots.

## 8.3. Accueils de chercheurs étrangers

En 2002, le projet Sharp a accueilli les chercheurs étrangers suivants :

- Prof. Zvi Shiller, Professeur à University of California, Los Angeles.
- Prof. Oussama Khatib, Professeur à l'Université de Stanford. Singapour.

## 9. Diffusion des résultats

### 9.1. Animation de la communauté scientifique

- Th. Fraichard a participé au comité de programme de la conférence internationale IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'2002).
- C. Laugier et J-M. Ahuactzin ont co-organisé avec le LAAS-CNRS à Toulouse, l'Ecole d'été en Image & Robotique en juillet 2002.
- C. Laugier a participé au comité de programme des conférences internationales suivantes : IEEE Int. Conf. on Robotics and Automation (ICRA'2002), IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'2002), Int. Conf. on Control, Automation, Robotics and Vision (ICARCV'2002), IEEE Int. Conf. on Intelligent Vehicles (IV'2002).
- C. Laugier est membre depuis 1997 du steering-advisory committee de la conférence IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), et depuis 2000 de l'advisory committee de la conférence internationale Int. Conf. on Control, Automation, Robotics and Vision (ICARCV).
- C. Laugier est membre du steering committee du réseau européen en robotique EURON.
- C. Laugier est membre du comité scientifique du programme national de robotique Robéa.
- S. Sekhavat a participé au comité de programme de la conférence internationale IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'2002).

### 9.2. Enseignements universitaires

Outre des interventions ponctuelles en second cycle universitaire, les membres du projet ont assuré les enseignements suivants :

- Cours « Motion planning » : Ecole d'été en Image & Robotique, Toulouse (FR) [juillet 2002].  
*Enseignants : J-M. Ahuactzin et Th. Fraichard.*
- Cours « Robotics » : Ecole d'été "Automatisation des syst ?mes de production", Grenoble (FR) [juillet 2002]. *Enseignant : Th. Fraichard.*

- Cours « Techniques avancées en planification de mouvement : DEA « Imagerie, Vision, Robotique » de l'INPG, Grenoble, (FR). *Enseignants : Th. Fraichard et S. Sekhavat.*
- TD-TP de Robotique : DEA « Imagerie, Vision, Robotique » de l'INPG, Grenoble, (FR). *Enseignant : Th. Fraichard.*
- Cours « Introduction to robotics » : Ecole d'été en Image et Robotique, Toulouse [juillet 2002]. *Enseignant C. Laugier.*
- Cours « Bayesian robot programming » : Ecole d'été en Image et Robotique, Toulouse [juillet 2002]. *Enseignant : O. Aycard*
- Cours « Robotique et autonomie de mouvement » : DEA « Imagerie, Vision, Robotique » de l'INPG, Grenoble, (FR). *Enseignant : C. Laugier.*
- Cours « Robotique » : en valeur C du Cnam. *Enseignant : C. Laugier et J. Troccaz.*
- Cours « Robotique et planification de mouvement » : Ecole Int. des Sciences et du Traitement de l'Information, Cergy Pontoise (FR). *Enseignant : C. Laugier.*

### 9.3. Participation à des colloques, séminaires, invitations

La diffusion des résultats et la participation active à des manifestations scientifiques internationales représentent deux aspects essentiels de l'activité du projet. On se reportera à la bibliographie pour en avoir la liste. Concernant les invitations aux différentes manifestations scientifiques :

- C. Laugier a fait une conférence sur invitation à la conférence ITSC'2002 à Singapour : "Towards automated roads : some projects and new technologies ».

## 10. Bibliographie

### Bibliographie de référence

- [1] C. BARD, C. LAUGIER, C. MILÉSI-BELLIER, J. TROCCAZ, B. TRIGGS, G. VERCELLI. *Achieving dextrous grasping by integrating planning and vision based sensing.* in « Int. Journal of Robotics Research », numéro 5, volume 14, octobre, 1995, pages 445-464.
- [2] M. CHERIF. *Motion planning for all-terrain vehicles : a physical modeling approach for coping with dynamic and contact interaction constraints.* in « IEEE Trans. Robotics and Automation », numéro 2, volume 15, 1999, pages 202-218.
- [3] A. DEGUET, A. JOUKHADAR, C. LAUGIER. *Models and Algorithms for the Collision of Rigid and Deformable Bodies.* éditeurs P. K. AGARWAL, L. E. KAVRAKI, M. T. MASON., in « Robotics : the algorithmic perspective », A K Peters, 1998, pages 327-338, Proc. of the Workshop on the Algorithmic Foundations of Robotics. Houston, TX (US). March 1998..
- [4] T. FRAICHARD. *Trajectory planning in a dynamic workspace : a 'state-time' approach.* in « Advanced Robotics », numéro 1, volume 13, 1999, pages 75-94.
- [5] A. JOUKHADAR, C. LAUGIER. *Dynamic Simulation : Model, Basic Algorithms, and optimization.* in « Proc. of the Workshop on the Algorithmic Foundations of Robotics », Toulouse (FR), juillet, 1996.
- [6] C. LAUGIER, P. GARNIER, T. FRAICHARD, I. PAROMTCHIK, A. SCHEUER. *Motion Planning and Sensor-Guided Manœuvre Generation for an Autonomous Vehicle.* éditeurs A. ZELINSKY., in « Field and Service Robotics », Springer, 1998, pages 60-67.

- [7] E. MAZER, P. BESSIÈRE, J.-M. AHUACTZIN. *The Ariadne's Clew Algorithm*. in « Journ. of Artificial Intelligence Research », volume 9, July-December, 1998, pages 295-316.
- [8] A. SCHEUER, T. FRAICHARD. *Continuous-Curvature Path Planning for Car-Like Vehicles*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », volume 2, pages 997-1003, Grenoble (FR), septembre, 1997.
- [9] B. TRIGGS, C. LAUGIER. *Automatic Task Planning for Robot Vision*. in « Proc. of the Int. Symp. on Robotics Research », éditeurs G. GIRALT, G. HIRZINGER., Munchen (DE), octobre, 1995.

### Articles et chapitres de livre

- [10] P. BESSIÈRE, E. MAZER. *Pour décider, mieux vaut parier*. in « La Recherche », numéro 350, février, 2002, pages 33-35, No. spécial : les nouveaux robots.
- [11] C. LAUGIER, M. C.. *L'immatériel au bout des doigts*. in « La Recherche », numéro 350, février, 2002, pages 52-55, No. spécial : les nouveaux robots.
- [12] S. SEKHAVAT, J. HERMOSILLO. *Cycab Bi-Steerable Cars - A New Family of Differentially Flat Systems*. in « Advanced Robotics », numéro 2, volume 5, 2002, pages 445.

### Communications à des congrès, colloques, etc.

- [13] C. COUÉ, T. FRAICHARD, P. BESSIÈRE, E. MAZER. *Multi-Sensor Data Fusion Using Bayesian Programming : an Automotive Application*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Lausanne (CH), septembre-octobre, 2002.
- [14] C. COUÉ, T. FRAICHARD, P. BESSIÈRE, E. MAZER. *Using Bayesian Programming for Multi-Sensor Data Fusion in Automotive Applications*. in « Proc. of the IEEE Intelligent Vehicle Symp. », Versailles (FR), juin, 2002, Poster session.
- [15] J. HERMOSILLO, C. PRADALIER, S. SEKHAVAT. *Modelling Odometry and Uncertainty Propagation for a Bi-steerable Car*. in « Proc. of the IEEE Intelligent Vehicle Symp. », Versailles (FR), juin, 2002, Poster session.
- [16] F. LARGE, S. SEKHAVAT, Z. SHILLER, C. LAUGIER. *Towards Real-Time Global Motion Planning in a Dynamic Environment Using the NLVO Concept*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Lausanne (CH), septembre-octobre, 2002.
- [17] F. LARGE, S. SEKHAVAT, Z. SHILLER, C. LAUGIER. *Using Non-Linear Velocity Obstacles to Plan Motions in a Dynamic Environment*. in « Proc. of the Int. Conf. on Control, Automation, Robotics and Vision », Singapore (SG), décembre, 2002.
- [18] P. PEK, O. LEBELTEL, C. LAUGIER. *Assisted and Automated Parallel Parking*. in « Proc. of the Int. Conf. on Control, Automation, Robotics and Vision », Singapore (SG), décembre, 2002.
- [19] C. PRADALIER, S. SEKHAVAT. *Concurrent matching, localization and map building using invariant features*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Lausanne (CH), septembre-octobre,

2002.

- [20] C. PRADALIER, S. SEKHAVAT. « *Localization Space* » : a Framework for Localization and Planning, for Systems Using a Sensor/Landmarks Module. in « Proc. of the IEEE Int. Conf. on Robotics and Automation », Washington, DC (US), mai, 2002.
- [21] K. SUNDARAJ, C. LAUGIER. *Physically Realistic Simulation of Large Deformations using LEM for Interactive Applications*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Lausanne (CH), septembre-octobre, 2002.
- [22] K. SUNDARAJ, C. MENDOZA, C. LAUGIER. *A Fast Method to Simulate Virtual Deformable Objects with Force Feedback*. in « Proc. of the Int. Conf. on Control, Automation, Robotics and Vision », Singapore (SG), décembre, 2002.

## Rapports de recherche et publications internes

- [23] S. BLONDIN. *Planification de mouvements pour véhicule automatisé en environnement partiellement connu*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2002.
- [24] H. BOUAZIZ VIALLET. *CATS : Calculateur Autonome de Tirages et Simulations*. Mémoire de fin d'études, Conservatoire Nat. des Arts et Métiers, Grenoble (FR), mars, 2002.
- [25] F. COLAS. *Modélisation analytique et bayésienne des neurones d'orientation de la tête*. Mémoire de Diplôme d'Etudes Approfondies, Univ. Joseph Fourier, Grenoble (FR), juin, 2002.
- [26] M. DUFOURNEAUD. *Bayesian Robot Evolution of a DogBot*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2002.
- [27] P. PEK. *Parking a Car Using Bayesian Programming*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2002.
- [28] F. RASPAIL. *Apprentissage bayésien par imitation*. Mémoire de Diplôme d'Etudes Approfondies, Univ. Joseph Fourier, Grenoble (FR), juin, 2002.
- [29] M. SDIKA. *Calcul de la distribution de maximum d'entropie-Application à la robotique*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2002.
- [30] A. TAPUS. *Utilisation de processus de décision markoviens pour la planification et l'exécution d'actions par un robot mobile*. Mémoire de Diplôme d'Etudes Approfondies, Univ. Joseph Fourier, Grenoble (FR), juin, 2002.
- [31] C. TAY MENG KEAT. *Design of a portable 2D simulator for mobile robots*. Industrial Attachment Program Report, Nanyang Technological University, Singapore, juin, 2002.

- [32] Y. USHANOFF. *Développement du concept d'agent autonome évoluant dans un monde virtuel*. Mémoire de fin d'études, Conservatoire Nat. des Arts et Métiers, Grenoble (FR), janvier, 2002.
- [33] A. VARGAS HERNÁNDEZ. *Iterative Trajectory Planning*. Master's internship report, Inst. Nat. de Recherche en Informatique et en Automatique, Montbonnot (FR), février, 1992.
- [34] S. VIEIRA. *Intégration de Modèles Déformables avec un Générateur d'images échographiques*. Mémoire de fin d'études, Conservatoire Nat. des Arts et Métiers, Grenoble (FR), janvier, 2002.
- [35] R. LE HY. *Programmation bayésienne de personnages de jeux vidéos*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2002.

## Bibliographie générale

- [36] R. ALAMI, R. CHATILA, S. FLEURY, M. GHALLAB, F. INGRAND. *An architecture for autonomy*. in « Int. Journal of Robotics Research », numéro 4, volume 17, avril, 1998, pages 315-337, Special issue on integrated architectures for robot control and programming.
- [37] D. AULIGNAC. *Modelisation de l'interaction avec des objets déformables en temps-réel pour des simulateurs médicaux*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble, France, décembre, 2001.
- [38] R. BALANIUK. *Using Fast Local Modeling to Buffer Haptic Data*. in « Proc. of the PHANToM Users Group Workshop », Boston, MA (US), octobre, 1999.
- [39] D. BARAFF. *Dynamic Simulation of Non-Penetrating Rigid bodies*. thèse de doctorat, Cornell University, mars, 1992.
- [40] D. BARAFF, A. WITKIN. *Dynamic Simulation of Non-penetrating Flexible Bodies*. in « Computer Graphics », numéro 2, volume 26, juillet, 1992.
- [41] F. BOUX DE CASSON. *Simulation dynamique de corps biologiques et changements de topologie interactifs*. Thèse de doctorat, Université de Savoie, Chambéry (FR), décembre, 2000.
- [42] R. A. BROOKS. *A Robust Layered Control System for a Mobile Robot*. éditeurs G. SHAFER, J. PERL., in « Readings in Uncertain Reasoning », Morgan Kaufmann, 1990, pages 204-213.
- [43] M. CHERIF, C. LAUGIER. *Motion Planning of Autonomous Off-Road Vehicles Under Physical Interaction Constraints*. in « Proc. of the IEEE Int. Conf. on Robotics and Automation », Nagoya (JP), mai, 1995.
- [44] S. COTIN. *Modèles anatomiques déformables en temps-réel..* Thèse de doctorat, INRIA Sophia Antipolis - Université de Nice, Sophia Antipolis, 1997.
- [45] F. DE LA ROSA, C. LAUGIER, J. NÁJERA. *Robust Path Planning in the Plane*. in « IEEE Trans. Robotics and Automation », numéro 2, volume 12, avril, 1996, pages 347-352.

- [46] G. DEBUNNE, M. DESBRUN, M.-P. CANI, A. BARR. *Dynamic Real-Time Deformations using Space and Time Adaptive Sampling*. in « Computer Graphics Proceedings », août, 2001.
- [47] A. DEGUET, A. JOUKHADAR, C. LAUGIER. *A Collision Model for Deformable Bodies*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Victoria, BC (CA), octobre, 1998.
- [48] M. FLIESS, J. LÉVINE, P. MARTIN, P. ROUCHON. *Flatness and Defect of Nonlinear Systems : Introductory Theory and Examples*. in « Int. Journal of Control », numéro 6, volume 61, 1995, pages 1327-1361.
- [49] T. FRAICHARD. *Planification de mouvement pour mobile non-holonome en espace de travail dynamique*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble (F), avril, 1992.
- [50] T. FRAICHARD. *Trajectory planning in a dynamic workspace : a 'state-time' approach*. in « Advanced Robotics », numéro 1, volume 13, 1999, pages 75-94.
- [51] T. FRAICHARD, R. MERMOND. *Path planning with uncertainty for car-like robots*. in « Proc. of the IEEE Int. Conf. on Robotics and Automation », volume 1, pages 27-32, Leuven (BE), mai, 1998.
- [52] T. FRAICHARD, A. SCHEUER, R. DESVIGNE. *From Reeds and Shepp's to continuous-curvature paths*. in « Proc. of the IEEE Int. Conf. on Advanced Robotics », pages 585-590, Tokyo (JP), octobre, 1999.
- [53] E. FRAZZOLI, M. A. DAHLEH, E. FERON. *Real-Time Motion Planning for Agile Autonomous Vehicles*. in « AIAA Journal of Guidance, Control and Dynamics », numéro 1, volume 25, 2002, pages 116-129.
- [54] Y. FUNG. *A First Course in Continuum Mechanics*. Prentice-Hall, Englewood Cliffs, N.J., 1994.
- [55] D. HSU, R. KINDEL, J.-C. LATOMBE, S. ROCK. *Randomized kinodynamic motion planning with moving obstacles*. in « Proc. of the Workshop on the Algorithmic Foundations of Robotics », Hanover, NH (US), mars, 2000.
- [56] E. JAYNES. *Probability theory - The logic of science*. 1995, <http://bayes.wustl.edu>.
- [57] J.-C. LATOMBE. *Robot motion planning*. Kluwer Academic Press, 1990.
- [58] C. LAUGIER, P. GARNIER, T. FRAICHARD, I. PAROMTCHIK, A. SCHEUER. *Motion Planning and Sensor-Guided Manœuvre Generation for an Autonomous Vehicle*. éditeurs A. ZELINSKY., in « Field and Service Robotics », Springer, 1998, pages 60-67.
- [59] C. LAUGIER, A. JIMENEZ, A. LUCIANI. *Simulating Physical Interactions Between an Articulated Mobile Vehicle and a Terrain*. in « Robotics and Autonomous Systems », numéro 11, volume 2, Elsevier, 1994, chapitre Special Issue on Telerobotics.
- [60] J.-P. LAUMOND. *Feasible trajectories for mobile robots with kinematic and environment constraints*. in « Proc. of the Int. Conf. on Intelligent Autonomous Systems », pages 346-354, Amsterdam (NL), 1986.

- [61] S. M. LAVALLE, J. J. KUFFNER. *Randomized kinodynamic planning*. in « Proc. of the IEEE Int. Conf. on Robotics and Automation », volume 1, pages 473-479, Detroit, MI (US), mai, 1999.
- [62] O. LEBELTEL. *Programmation bayésienne des robots*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), octobre, 1999.
- [63] L. LISOWSKI, G. BAILLE. *Specifications of a small electric vehicle : modular and distributed approach*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », Grenoble (FR), septembre, 1997.
- [64] T. LOZANO-PEREZ, M. T. MASON, R. H. TAYLOR. *Automatic synthesis of fine motion strategies for robots*. in « Int. Journal of Robotics Research », numéro 1, volume 3, 1984, pages 3-24.
- [65] E. MAZER, J.-M. AHUACTZIN, P. BESSIÈRE. *The Ariadne's Clew Algorithm*. in « Journ. of Artificial Intelligence Research », volume 9, July-December, 1998, pages 295-316.
- [66] K. MEKHNACHA. *Méthodes probabilistes bayésiennes pour la prise en compte des incertitudes géométriques : application à la CAO-robotique*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juillet, 1999.
- [67] C. MENDOZA SERRANO, C. LAUGIER. *Realistic Haptic Rendering for Highly Deformable Virtual Objects*. in « Proc. of the Int. Conf. on Virtual Reality », Yokohama (JP), mars, 2001.
- [68] B. V. MIRTICH, J. F. CANNY. *Impulse-based Dynamic Simulation*. in « Proc. of the Workshop on the Algorithmic Foundations of Robotics », San Francisco, CA (US), février, 1994.
- [69] N. J. NILSSON. *Shakey the Robot*. rapport technique, numéro 347, Artificial Intelligence Center, SRI International, Memlo Park (CA), 1984.
- [70] J. O'BRIEN, J. HODGINS. *Graphical Modeling and Animation of Brittle Fracture..* in « Computer Graphics », 1999, Proceedings of SIGGRAPH'99.
- [71] C. PRADALIER. *Conception d'un système de localisation pour un robot mobile : utilisation d'un télémètre laser et placement d'amers dans l'environnement*. Mémoire de Diplôme d'Etudes Approfondies, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), juin, 2001.
- [72] D. C. RUSPINI, K. KOLAROV, O. KHATIB. *Haptic Interaction in Virtual Environments*. in « Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems », pages 128-133, Grenoble (FR), septembre, 1997.
- [73] S. THRUN, M. BENNEWITZ, W. BURGARD, A. CREMERS, F. DELLAERT, D. FOX, D. HÄHNEL, C. ROSENBER, N. ROU, J. SCHULTE, D. SHULTZ. *MIMERVA : A second generation mobile tour guide robot*. in « Proc. of the IEEE Int. Conf. on Robotics and Automation », San Francisco, CA, 1999.
- [74] S. THRUN. *Probabilistic Algorithm in Robotics*. in « AI Magazine (Tech. Report CMU-CS-00-126) », 2001, à paraître.
- [75] J. TROCCAZ, P. PUGET. *Dealing with uncertainties in robot planning using program proving techniques*.

éditeurs R. C.BOLLES, B. ROTH., in « Proc. of the Int. Symp. on Robotics Research », MIT Press, 1988, pages 455-466.

- [76] D. D'AULIGNAC, R. BALANIUK, C. LAUGIER. *A Haptic Interface for a Virtual Exam of the Human Thigh.* in « Proc. of the IEEE Int. Conf. on Robotics and Automation », volume 3, pages 2452-2457, San Francisco, CA (US), avril, 2000.
- [77] D. D'AULIGNAC, M. C. CAVUSOGLU, C. LAUGIER. *Modelling the dynamics of a human thigh for a realistic echographic simulator with force feedback.* in « Proc. of the Int. Conf. on Medical Image Computer-Assisted Intervention », Cambridge (BG), septembre, 1999.