

Projet BIP

Contrôle/commande de robots marcheurs et applications

Rhône-Alpes

THÈME 4A

R *apport*
d'Activité

1999

Table des matières

1	Composition de l'équipe	3
2	Présentation et objectifs généraux	4
3	Fondements scientifiques	4
3.1	Les robots bipèdes	4
3.1.1	Motivations	5
3.1.2	Problématique scientifique	5
3.2	Contrôle/commande de systèmes robotiques complexes	6
4	Domaines d'applications	9
5	Logiciels	9
6	Résultats nouveaux	10
6.1	Modélisation et commande	10
6.1.1	Contrôle du centre de masse d'un robot bipède pendant la montée et la descente d'escaliers	11
6.1.2	Poursuite de trajectoires pour un double pendule sous-actionné	12
6.1.3	Manipulateurs à contraintes et bipèdes	13
6.1.4	Stabilité des systèmes non-linéaires en cascade	13
6.1.5	Commande par fonction de tâche s'adaptant aux déséquilibres du robot	13
6.1.6	Activités autour des prototypes de bipèdes	14
6.2	Perception et simulation	15
6.2.1	Asservissement visuel	15
6.2.2	Simulation graphique 3D d'un robot bipède	18
6.3	Contrôle/commande de systèmes robotiques complexes	19
6.3.1	Spécification et implantation d'une commande sûre d'un bras manipulateur	19
6.3.2	Etude de la stabilité d'une loi de commande pour véhicules autonomes	20
6.4	Contrôle/commande: validation et mise en œuvre	20
6.4.1	Génération de code ESTEREL pour ORCCAD	21
6.4.2	Sur le couplage d'ORCCAD avec SYNDEX	22
6.4.3	Tolérance aux pannes	22
6.4.4	Analyse de systèmes temps-réels préemptifs par l'algèbre $(\max,+)$	23
6.4.5	Vérification de propriétés de programmes ORCCAD	23
6.5	Langages réactifs	24
6.5.1	Formalisation des machines d'exécution pour programmes synchrones	24
6.5.2	Langages de programmation d'automates	25

7 Contrats industriels (nationaux, européens et internationaux)	25
7.1 VIGOR	25
7.2 TELEDIMOS	26
7.3 Transfert industriel	26
8 Actions régionales, nationales et internationales	26
8.1 Actions nationales	26
8.1.1 GdR-PRC "Commande de robots à pattes"	26
8.1.2 GdR Automatique	26
8.1.3 PSLC: Pôle Systèmes et Logiciels Critiques	26
8.1.4 Groupe GRAFCET	26
8.1.5 Collaborations internes à l'Inria	26
8.1.6 Collaborations avec d'autres laboratoires	27
8.2 Actions européennes	27
8.3 Actions internationales	28
9 Diffusion de résultats	28
9.1 Animation de la Communauté scientifique	28
9.2 Exposition Universelle	29
9.3 Enseignement	29
10 Bibliographie	29

1 Composition de l'équipe

Responsable scientifique

Bernard Espiau [DR Inria]

Assistante de projet

Françoise de Coninck [TR, à temps partiel dans le projet]

Personnel Inria

Alain Girault [CR]

Ambarish Goswami [CR, en délégation à l'université de Pennsylvanie, jusqu'à sa démission au 1er mai 1999]

Eric Rutten [CR]

Philippe Sardain [en délégation de l'IUT de Poitiers]

Daniel Simon [CR]

Chercheurs doctorants

Christine Azevedo [Boursier MENRT, depuis le 1er octobre 1999]

Nicolas Andreff [Boursier MENRT, en commun avec le projet MOVI]

Bilal El-Ali [Boursier MENRT]

Laure France [ATER jusqu'en août 1999, en commun avec le projet IMAGIS, jusqu'en octobre 1999]

Pierre-Brice Wieber [Boursier DGA]

Chercheurs post-doctorants

Nathalie Cislo [Post Doctorant, en coopération avec les Moyens Robotiques, depuis le 1er septembre 1999]

Cătălin Dima [Post Doctorant TOLERE, depuis le 2 novembre 1999]

Elena Panteley [Post Doctorant, depuis le 5 janvier 1999]

Mihaela Sighireanu [Post Doctorant TOLERE, jusqu'au 1er octobre 1999]

Stagiaires

Christine Azevedo [mars-septembre 1999, DEA]

Jean-Jacques Parmentier [mars-juillet 1999, DEA]

Adriana Vilchis [mars-septembre 1999, DEA]

Collaborateurs extérieurs

François Baccelli [ENS, Paris]

Carlos Canudas de Wit [LAG]

Antonio Loria [LAG]

Sergio Yovine [VERIMAG]

2 Présentation et objectifs généraux

Le projet a pour objectif l'étude générique des divers aspects intervenant dans le contrôle/commande des systèmes robotiques complexes. Il s'intéresse en particulier aux robots marcheurs de type bipède. L'intérêt des systèmes marcheurs réside dans leur capacité d'adaptation à des terrains ou des sols variés, leur permettant de se déplacer dans des milieux très contraints en dimensions, voire mal structurés. Les robots anthropomorphes sont donc particulièrement aptes à évoluer dans nos environnements courants, privés ou industriels, essentiellement conçus pour la bipédie. Ainsi, les domaines d'application visés sont-ils en priorité les robotiques personnelle, de service et d'intervention. Parallèlement, le projet s'attache à développer des activités de modélisation dans certains domaines de la biomécanique. Enfin, le projet ne souhaite pas que les applications de ses recherches se limitent au domaine des seuls robots marcheurs, dont le marché industriel est actuellement marginal. C'est pourquoi les techniques étudiées sont voulues suffisamment génériques pour faire l'objet de mises en œuvre dans d'autres domaines (cf section 4).

En résumé, les activités du projet se déclinent selon deux grands thèmes scientifiques : la modélisation et la commande de systèmes marcheurs d'une part ; l'étude de méthodes et d'outils pour le contrôle/commande temps-réel d'autre part.

3 Fondements scientifiques

3.1 Les robots bipèdes

Mots clés : commande, commande référencée capteur, mécanique, mécanique des solides, modélisation, robotique, robotique mobile, simulation de système mécanique, temps réel, asservissement visuel.

3.1.1 Motivations

Dans la classe des systèmes mobiles, les robots marcheurs, par exemple hexapodes, présentent des avantages certains sur leurs homologues à roues dès que le sol n'est plus plan ou libre : le franchissement des obstacles est plus aisé, l'emprise au sol plus faible, l'adaptabilité meilleure. Ceci concerne les grands domaines de la robotique non-manufacturière : exploration, maintenance, intervention, service. Cependant, dès lors que l'environnement de travail du système est conçu pour l'homme, la technologie multipode doit en général laisser place à la bipédie si l'on désire conserver de bonnes possibilités de déplacement et d'action sans modifier l'environnement. D'où l'intérêt assez récent que porte la communauté mondiale de Recherche et Développement en robotique aux systèmes dits humanoïdes, destinés à accompagner l'homme dans certaines de ses activités personnelles ou professionnelles. Par exemple, une certaine forme d'assistance à domicile de personnes à mobilité réduite, pour des tâches très routinières, pourrait être assurée par des robots bipèdes, car ceux-ci ont la faculté de pouvoir se déplacer sans adaptation particulière de l'environnement.

Si la faisabilité de tels systèmes reste largement hors d'atteinte pour ce qui est de robots aux capacités d'autonomie décisionnelle élevées, le niveau actuel de la technologie permet par contre d'envisager la réalisation de machines capables de se déplacer en marchant dans des conditions bien déterminées et d'exécuter quelques actions très simples.

3.1.2 Problématique scientifique

Conception générale La difficulté de conception d'un robot marcheur bipède résulte, d'une part de la nouveauté du concept, qui fait qu'il n'existe pas encore de solutions éprouvées, et d'autre part de contraintes spécifiques au principe même de la marche bipédique : tout d'abord, le système doit comporter un nombre élevé d'articulations dans un espace de volume réduit. De plus, la localisation de celles-ci est elle-même fort contrainte par des raisons de position des articulations, d'encombrement et de répartition des masses, ce qui nécessite par exemple l'étude de groupes moto-transmetteurs spécifiques. Par ailleurs, le système doit être en permanence en équilibre, soit statique, soit dynamique, ce qui peut nécessiter des couples instantanés élevés. Ceux-ci se répartissent d'ailleurs au cours du temps sur l'ensemble des articulations, posant ainsi le problème d'une gestion globale et dynamique de la puissance nécessaire. Ce dernier point est également lié aux choix technologiques concernant l'autonomie : la place disponible est faible et la masse embarquable limitée (elle se situe en général en hauteur (tronc) et a de ce fait une grande influence sur la dynamique du système).

Ce problème reste donc largement ouvert, l'état de l'art en la matière étant constitué par les très belles réalisations de Honda, les robots humanoïdes P2 et P3 présentés en 1997.

Modélisation Un robot bipède peut être modélisé comme une chaîne articulée arborescente de corps rigides dans l'espace tridimensionnel. La marche est caractérisée par différentes phases^[VDO92,Win90], principalement (pour une jambe donnée) le balancement (35 % du cycle) et le support (65 % du cycle). Il existe de plus une phase de double support qui dure environ

[VDO92] C. L. VAUGHAN, B. L. DAVIS, J. C. O'CONNOR, *Dynamics of Human Gait*, Human Kinetic, 1992.

[Win90] D. A. WINTER, *Biomechanics and Motor Control of Human Movement*, Wiley-Interscience, 1990.

12 % du cycle et qui est absente dans la course. Ces phases se décomposent elles-mêmes plus finement en fonction d'événements liés au mouvement du centre de gravité et, surtout, du pied. Ces différentes phases se caractérisent par des contacts divers entre le système et le sol.

Ainsi, le modèle mécanique d'un tel système comporte-t-il trois aspects :

- une dynamique de corps rigides représentable par des équations lagrangiennes ;
- un ensemble, variable selon les phases, de contraintes inégalités traduisant l'existence de contacts sans pénétration et sans glissement que *doit* respecter le système. Chaque ensemble définit un mode de fonctionnement ;
- un choix de loi d'impact modélisant les transitions (supposées instantanées) entre modes.

Nous sommes donc en présence d'un système dynamique hybride complexe, dont l'étude reste encore peu explorée.

Commande Un marcheur humain adopte naturellement une allure régulière et symétrique dite *de confort* dans laquelle sa consommation d'énergie est minimale pour une vitesse de progression raisonnable. Par ailleurs, certaines classes de systèmes mécaniques hybrides, par exemple un robot sauteur ou une transmission avec jeu, présentent souvent des comportements cycliques limites correspondant à des équilibres dynamiques, parfois mêmes naturellement stables dans un certain domaine. Dans le cas du robot marcheur le plus simple, le compas sur une pente, ces cycles correspondent à des trajectoires périodiques passives (sans actionnement extérieur), dans lesquelles la transition entre énergie cinétique et énergie potentielle est entièrement équilibrée par l'absorption d'énergie à l'impact [GTE98].

Tout ceci nous incite donc à rechercher une approche de la commande dans laquelle les trajectoires sont cycliques et énergétiquement optimales, en un sens qui reste à préciser. Un autre point est que ce qui importe vraiment est la progression globale du robot tout en préservant une stabilité mécanique particulière qui est de nature dynamique. Les approches classiques, comme le suivi précis de trajectoires articulaires figées sont donc inadaptées, sauf si l'on s'intéresse principalement au contrôle de la posture. Le domaine n'étant pas lui-même stabilisé (et loin s'en faut), le chercheur explorera donc des techniques de commande variées : optimisation non-différentiable, commande prédictive, commande par apprentissage itératif, commande par fonction de tâches [6]... On n'oubliera pas enfin la commande référencée capteurs [3], qui permettra de prendre en compte des mesures locales de distance, de proximité, de réaction au sol...

3.2 Contrôle/commande de systèmes robotiques complexes

Mots clés : architecture de contrôle, contrôle/commande, système embarqué, programmation synchrone, vérification formelle, environnement de programmation, temps réel, système hybride.

[GTE98] A. GOSWANI, B. THUILOT, B. ESPIAU, « A Study of the Passive Gait of a Compass-like Biped Robot: Symmetry and Chaos », *Int. Journal of Robotics Research* 17, 12, décembre 1998.

Un des problèmes principaux de la programmation des systèmes robotiques est d'aborder dans un cadre cohérent la description de l'évolution dynamique des deux composantes -continue et discrète- de ces systèmes hybrides. Ceci nécessite d'intégrer différents formalismes, trop souvent spécialisés dans un domaine précis et déconnectés les uns des autres. Ainsi, l'étude des *architectures de contrôle* est primordiale pour développer concepts et méthodes permettant de gérer la complexité de la spécification et de l'analyse de la globalité du système.

Un contrôleur de robots doit être capable de traiter une grande diversité d'actions, allant par exemple de la commande d'une cellule flexible d'assemblage en milieu industriel au contrôle d'un véhicule autonome opérant en milieu hostile. La diversité des applications potentielles en robotique et le désir de mettre en œuvre des algorithmes de commande sophistiqués impliquent une ouverture du système de programmation. D'un point de vue industriel, les contrôleurs actuels restent le plus souvent des machines fermées, dotées d'un langage limité ne permettant pas d'exécuter des actions complexes telles que des commandes référencées capteurs. Outre leurs limitations fonctionnelles, ils ne permettent pas de traiter correctement les aspects temps-réel sous-jacents à toute application robotique, même simple.

L'architecture de contrôle ORCCAD [7, 1] propose actuellement deux niveaux : le niveau *fonctionnel*, où les aspects commande en temps discrétisé sont prépondérants, et le niveau *contrôle*, qui gère principalement les aspects réactifs. Au niveau fonctionnel, les actions élémentaires sont définies sous forme de Tâches-Robot : ce concept clé spécifie dans la même entité une partie algorithmique (loi de commande) et une partie logique réactive. Il s'agit d'un objet informatique hybride, à l'interface des aspects temps continu et discret. En l'absence de modèles unifiés et opérationnels des systèmes hybrides, cette approche permet de cantonner l'interface temps continu/discret dans la coquille réactive de la Tâche-Robot. Les actions élémentaires exécutées par le robot sont réalisées par l'exécution périodique d'une loi de commande en boucle fermée dont la conception repose sur les techniques de l'automatique classique comme la spécification par fonction de tâches [6], la commande référencée capteurs ou encore la commande instationnaire pour systèmes non holonomes. Cette loi de commande est *encapsulée* dans un comportement logique réactif rythmant son exécution à l'aide de signaux typés (pré-conditions, exceptions et post-conditions) constituant sa *vue externe*. Ce comportement réactif est codé à l'aide du langage synchrone ESTEREL, ce qui permet de bénéficier des outils de vérification formelle associés FC2TOOLS/ATG.

Au niveau contrôle, les actions complexes sont spécifiées sous la forme de Procédures-Robot par composition structurée et hiérarchique de Tâches-Robot, manipulées par l'intermédiaire de leur vue externe, et de traitements d'exception. À la différence de ces dernières, les Procédures-robot n'ont pas de partie commande, autre que celle des Tâches-robot qui les composent. Nous pouvons ainsi définir de façon cohérente et dans un même formalisme des actions de complexité croissante, gérant les modes nominaux d'une mission robotique complexe ainsi que les modes dégradés dépendant des exceptions spécifiées et du contexte d'exécution. Tous ces aspects logiques sont également spécifiés en ESTEREL du niveau le plus haut (mission) jusqu'aux appels systèmes gérant l'exécution des lois de commande, un soin tout particulier étant pris quant à la gestion des transitions entre commandes successives. Ceci nous permet d'effectuer *a posteriori* une vérification formelle des programmes produits à un grand niveau de détail.

La formalisation des structures manipulées présente de nombreux avantages permettant la

définition d'un environnement de programmation cohérent et convivial [SKE98] :

- ceci permet de concevoir des interfaces de programmation adaptées à chaque type d'utilisateur du système. Ainsi, l'automaticien chargé du développement d'une Tâche-Robot peut en coder le comportement réactif sans rien connaître à ESTEREL, la vue externe de la Tâche-Robot étant générée automatiquement à partir d'une fenêtre graphique. D'autre part, le concepteur de mission ne code que la partie spécifique de son application, l'essentiel du code source, et en particulier celui permettant la connexion avec le système d'exploitation sous-jacent, étant généré automatiquement de façon totalement transparente ;
- l'identification de certaines propriétés génériques d'un système ainsi structuré (vivacité, sûreté) permet d'automatiser partiellement la génération de critères abstraits facilitant ainsi l'utilisation des outils de vérification ;
- le code temps-réel téléchargeable est automatiquement généré à partir des spécifications de haut niveau.

De nombreux travaux restent cependant à réaliser et certains problèmes restent ouverts :

- visant la conception de systèmes embarqués temps-réel à sûreté critique, la génération de code doit être améliorée afin de produire du code réparti optimisé et tolérant aux pannes ;
- l'utilisation des méthodes de vérification formelle actuelles reste difficile pour un non spécialiste du domaine, notamment en ce qui concerne le diagnostic et l'interprétation des résultats. Nous nous heurtons également rapidement à des problèmes de taille d'automates à vérifier, même si la méthodologie d'ORCCAD permet d'effectuer une spécification et des vérifications incrémentales. La prise en compte de contraintes temporelles n'est pas actuellement réalisée ;
- enfin, il manque au niveau supérieur une couche *décisionnelle* de type planification réactive, permettant une éventuelle reprogrammation partielle de la mission en cas de défaillance non spécifiée ou encore de spécifier des missions complexes de façon conviviale en évitant des fautes ou oublis de spécification. Nous fondons de sérieux espoirs sur l'utilisation de techniques de synthèse de contrôleurs pour systèmes à événements discrets, initialisés par Ramadge et Wonham [RW87], pour construire ce niveau sur un formalisme mathématique solide.

Nous proposons ainsi, dans un environnement cohérent, un ensemble d'outils intégrés permettant de programmer, de valider et de générer du code, de manière transparente pour l'utilisateur, pour ces méthodes et concepts bien formalisés.

[SKE98] D. SIMON, K. KAPellos, B. ESPIAU, «Formalization of hybrid structures in robot controllers: the Orccad approach», in : 9th Symposium on information control in manufacturing (INCOM'98), IFAC, Nancy, France, juin 1998.

[RW87] P. RAMADGE, W. WONHAM, «Supervisory control of a class of discrete event processes», *SIAM Journal of Control and Optimization* 25, janvier 1987, p. 206–223.

4 Domaines d'applications

Mots clés : robotique, système embarqué, robot marcheur.

Les domaines d'applications des robots marcheurs sont principalement ceux de la robotique non-manufacturière. Plus précisément, le projet viserait idéalement la robotique d'assistance (personnes âgées, handicapés) et d'intervention en milieu hostile terrestre, comme le nucléaire, en tant que secteurs applicatifs. Le premier domaine est également concerné par nos activités en biomécanique ; celles-ci ont aussi des applications dans le domaine du matériel à usage sportif. Les techniques de commande référencée vision sans étalonnage étudiées dans le projet trouvent quant à elles des applications dans d'autres domaines, dont la robotique manufacturière (manipulation, soudage, inspection) et les robots mobiles.

Enfin, les travaux concernant la programmation sûre sont d'une portée très large puisque s'appliquant à une large classe de systèmes critiques. Sont visés en particulier les systèmes présentant une forte interaction avec l'environnement, par le biais de nombreux capteurs et actionneurs, comme les véhicules automatiques [16] ou les engins sous-marins autonomes. Le domaine d'application peut s'élargir aux systèmes de contrôle/commande pour la production, et aux automates programmables, comme le suggère le transfert industriel lié à ORCCAD.

5 Logiciels

Participants : Daniel Simon, Alain Girault, Bernard Espiau, Roger Pissard-Gibollet [Moyens robotiques, correspondant].

Mots clés : robotique, contrôle/commande, temps réel, spécification, vérification.

Résumé : *ORCCAD (Open Robot Controller Computer Aided Design) est un environnement logiciel permettant de concevoir et de mettre en œuvre le contrôle et la commande d'un système robotique complexe. Il permet également la spécification et la validation des missions à réaliser par ce système.*

ORCCAD est principalement destiné aux applications temps réel critiques en robotique, dans lesquelles les aspects relevant de l'automatique (les asservissements, les commandes) sont amenés à interagir étroitement avec ceux manipulant des événements discrets. De tels systèmes sont souvent qualifiés d'hybrides. ORCCAD est conçu et développé en commun entre les projets BIP et ICARE (Sophia Antipolis) et le service des Moyens Robotiques de l'UR Rhône-Alpes. Ses principaux constituants sont :

- un éditeur de composants logiciels élémentaires (Modules)
- un éditeur graphique pour la spécification de Tâches-Robot (figure 1)
- un éditeur de Procédures-Robot, générant automatiquement la plus grande partie du code de contrôle ESTEREL
- un langage de spécification de missions robotiques (MAESTRO)

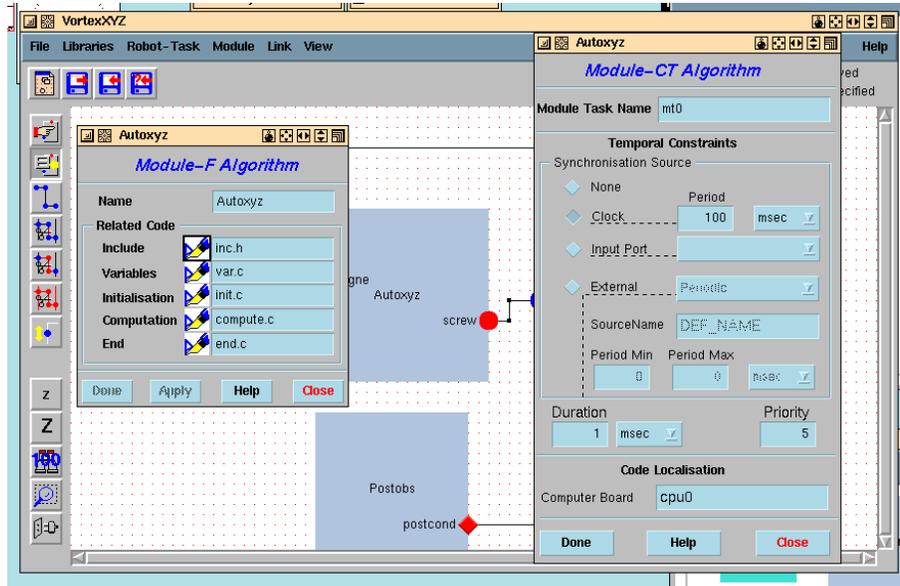


FIG. 1 – Éditeur de Tâches-Robot

- une interface de vérification générant des critères abstraits pour les outils FC2TOOLS/-ATG
- une connexion avec des simulateurs logiques (XES) ou hybrides (SIMPARC)
- un générateur de code temps-réel pour VXWORKS ou SOLARIS

ORCCADV3.0 β est distribuée sur une demi-douzaine de sites académiques à des fins d'évaluation. Une présentation du logiciel est disponible à l'URL :

<http://www.inrialpes.fr/iramr/pub/Orccad/>

Les fonctionnalités améliorées dans cette nouvelle version sont :

- un générateur de code pour le simulateur SIMPARC (Jean-Jacques Borrelly, INRIA-SOPHIA-ANTIPOLIS)
- un exécutif pour Posix/Solaris 2.6
- des modifications mineures de l'IHM.

6 Résultats nouveaux

6.1 Modélisation et commande

Participants : C. Azevedo, N. Cislo, B. El Ali, B. Espiau, A. Loria [LAG], E. Panteley, J.-J. Parmentier, R. Pissard-Gibollet [M.R.], P. Sardain [LMS], P.-B. Wieber.

6.1.1 Contrôle du centre de masse d'un robot bipède pendant la montée et la descente d'escaliers

Dans le cadre de notre travaux de recherche sur le contrôle de la marche d'un robot bipède, nous nous sommes intéressées à la marche statiquement stable. Elle est caractérisée par le maintien de la projection verticale du centre de masse (CM) à l'intérieur de la surface de sustentation. De plus, ce mode de marche est caractérisé par une vitesse très lente afin de réduire les effets dynamiques du mouvement. la marche peut donc être considérée comme une succession de postures statiquement stables. La marche humaine est, en général, une marche dynamique où l'être humain profite de l'inertie et de la force de gravité pour effectuer des mouvements balistiques afin d'optimiser la consommation d'énergie. Cependant, on observe la marche statiquement stable pendant la marche sur une surface accidentée ou pendant la montée d'escaliers.

Par analogie à la marche humaine, nous considérons que la marche humaine est une marche périodique où un cycle de marche comprend une phase de simple support et une autre phase de double support. La méthode de commande est basée sur l'approche de contrôle par fonction de tâche [6] où la tâche globale comprend une tâche principale de suivi de trajectoires de référence et une tâche secondaire de contrôle du CM. Une sous-tâche supplémentaire est aussi introduite pour prendre en compte les butées articulaires mécaniques. En général, les trajectoires de référence sont décrites par des polynômes temporels dont l'ordre dépend des contraintes imposées sur le comportement du modèle du robot.

Après avoir appliqué cette approche sur la marche sur une surface plane, nous l'avons étendue pour la montée et la descente d'escaliers. Nous considérons qu'on dispose d'une description détaillée de l'environnement extérieur, en particulier de la géométrie des marches d'escaliers. Dans le cas réel, ces informations peuvent être obtenues par reconstruction de l'environnement extérieur grâce aux informations fournies par des capteurs extéroceptifs [10].

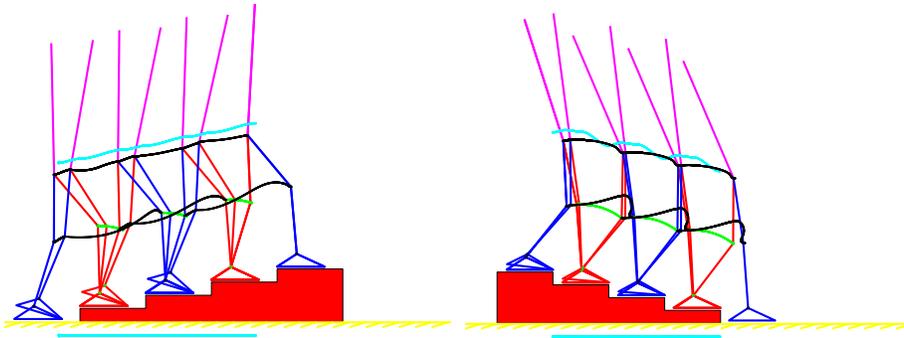


FIG. 2 – *Montée et descente d'escaliers*

La figure 2 illustre les résultats de notre approche pour la montée et la descente d'escaliers. Les animations de ces simulations sont visibles à l'adresse <http://www.inrialpes.fr/bip/>. En ce qui concerne la marche sur une surface plane ou la montée d'escalier, les résultats sont

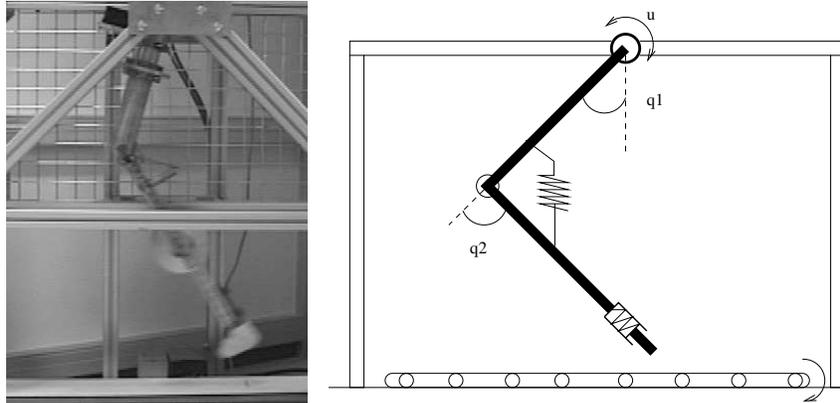


FIG. 3 – *Le double pendule sous-actionné*

très satisfaisants tandis que pour la descente d'escaliers ils le sont moins. En effet, on observe une inclinaison excessive du tronc vers l'arrière afin de conserver l'équilibre statique : on ne retrouve pas ce phénomène dans la réalité parce que l'être humain effectue un mouvement balistique dans ce mode de marche.

Afin de mettre en pratique ces résultats, nous avons adapté nos algorithmes et nos programmes de simulation pour correspondre au robot BIP en utilisant les caractéristiques géométriques, cinématiques et massiques de ce robot réel.

6.1.2 Poursuite de trajectoires pour un double pendule sous-actionné

L'INRIA dispose d'un dispositif mécanique destiné à étudier expérimentalement la marche balistique (voir Figure 3 et rapport d'activité 1998). Il s'agit d'un double pendule qui représente une jambe humaine. Le système est sous-actionné, seule la hanche est équipée d'un actionneur, l'articulation passive du genou étant munie d'un ressort de rappel. Le pied télescopique, muni d'un piston, peut rencontrer un tapis roulant qui permet de simuler le déplacement longitudinal. L'objectif du stage de DEA de C. Azevedo était de simuler la marche en réalisant des pas successifs (voir Figure 4).

Nous avons défini des trajectoires de référence réalisables dynamiquement pour chacune des deux articulations. L'ensemble de ces trajectoires définit une allure de marche pour le système. Une loi de commande de type PD, compensation de gravité et terme d'avance a été conçue pour assurer la poursuite de ces trajectoires. Des simulations sur MAPLE ont été réalisées et ont permis de contrôler les erreurs de poursuite avec des couples réalistes par rapport au moteur dont nous disposons. La stabilité a été partiellement analysée du fait des difficultés survenues lors des résolutions numériques. Malgré la simplicité apparente du système, le sous-actionnement engendre nombre de problèmes. Cette étude a mis en évidence les difficultés quant à appliquer les outils d'automatique existants [38].

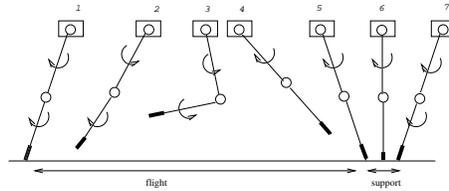


FIG. 4 – Un pas du double pendule

6.1.3 Manipulateurs à contraintes et bipèdes

Nous avons étudié le problème de la commande en position/force de manipulateurs robotiques interagissant avec un environnement rigide [17], ce qui est très lié au problème de la commande de la locomotion bipède.

Nous remarquons que tous les problèmes mentionnés ci-dessus ont été résolus sous l'hypothèse que l'équation de contraintes *n'est pas* soluble globalement. Cette hypothèse est importante car dans la plupart des cas pratiques, l'équation des contraintes est soluble uniquement pour des sous-ensembles des coordonnées. Cependant, bien que l'hypothèse de globalité soit très restrictive, elle est utilisée par la plupart des auteurs. Ainsi, nos résultats garantissent, dans un premier lieu, que les trajectoires n'échappent pas à l'ensemble où la Jacobienne de l'équation de contraintes est à rang plein.

6.1.4 Stabilité des systèmes non-linéaires en cascade

Poursuivant des travaux entrepris à l'Université de Santa Barbara (É.-U.), nous étudions le problème de stabilité de systèmes non-linéaires variants dans le temps en cascade, en collaboration avec A. Loria du LAG (UMR-CNRS 5528). Nous recherchons les conditions suffisantes pour garantir que l'interconnexion en cascade (boucle ouverte) de deux systèmes globalement uniformément asymptotiquement stables (UGAS), reste UGAS. Comme nous sommes motivés par des problèmes de suivi de trajectoire, et que nous nous intéressons en premier lieu aux systèmes non-autonomes, ces conditions suffisantes ont été établies ^[PL98] sous la forme d'une propriété d'intégrabilité de l'état du système "perturbateur".

Il faut remarquer que ces résultats ont fondamentalement permis la résolution, dans des travaux antérieurs, des problèmes pratiques suivants :

- principe de séparation pour systèmes de type Euler-Lagrange [34].
- commande de navires [12],
- commande d'un moteur turbo-diesel [19].

6.1.5 Commande par fonction de tâche s'adaptant aux déséquilibres du robot

La principale spécificité des robots marcheurs est le contact intermittent avec le sol, permettant une grande flexibilité de déplacements, mais imposant du même coup une instabilité

[PL98] E. PANTELEY, A. LORIA, «Global Uniform Asymptotic Stability of Time-varying Nonlinear Systems in Cascade», *System and Control Letters* 33, 2, 1998, p. 131–138.

structurelle du robot. Ce contact prend la forme de contraintes unilatérales, aboutissant à un Problème de Complémentarité (CP). Etudiant l'interaction d'un modèle de robot à base flottante avec ce CP, nous avons pu mettre en évidence la nécessité de contrôler en priorité ce contact du robot avec le sol, ainsi que les mouvements du centre de masse du robot. Le découplage nécessaire entre ces variables nous amène tout naturellement à envisager une loi de commande sous forme de fonction de tâche.

Or, sur la base du modèle dynamique précédemment développé, il apparaît que les contraintes sur les mouvements réalisables par le robot limitent fortement la stabilité que l'on peut obtenir [36]. En s'inspirant de notions issues de la théorie de la viabilité [Aub91], il apparaît que le défaut de stabilité prend son origine dans le choix de stabiliser une simple trajectoire. Nous avons donc proposé de modifier l'objectif de la loi de commande en ne considérant plus une seule trajectoire à stabiliser, mais en prenant en compte la totalité des trajectoires se manifestant dans un comportement de marche. Cette définition plus générale de la marche aboutit à considérer une famille paramétrée de trajectoires variant selon la longueur du pas, sa hauteur, sa vitesse d'exécution...

Afin de stabiliser cette famille de trajectoires prise dans son ensemble, il a été proposé de faire varier les paramètres de marche en fonction de l'état dynamique du robot. Il est alors apparu qu'une variation deux fois différentiable des paramètres avec le temps permet de faire apparaître le lien direct entre les contraintes imposées par le contact avec le sol, le déséquilibre du robot et les variations nécessaires des paramètres de la marche. On obtient ainsi une loi de commande qui essaie de suivre un comportement désiré dans les limites imposées par la dynamique, et qui permet d'éviter les chutes par le biais de modifications de ce comportement.

Enfin, une étude du système d'actionnement du robot BIP-2000 a été réalisée lors de l'encadrement du stage de fin d'études de l'Ecole Polytechnique de Jean-Jacques Parmentier [39].

6.1.6 Activités autour des prototypes de bipèdes

Deux prototypes du robot bipède ont été réalisés, avec le laboratoire de mécanique des solides de Poitiers, et donnent lieu à un travail de mise en place, lié aux expérimentations des résultats du projet et à l'Exposition Universelle de Hanovre en 2000 (voir Section 9.2).

Un scénario a été défini, par l'étude d'un compromis entre, d'une part, les postures et mouvements statiques ou dynamiques commandables, et, d'autre part, les contraintes liées aux aspects sûreté de fonctionnement du robot. Le scénario global est centré autour d'une chorégraphie tirant parti de la grande mobilité du robot (15 degrés de liberté pour l'arborescence tronc + pelvis + jambes). La marche continue sera filmée en laboratoire et projetée parallèlement.

Les tâches suivantes sont en cours de réalisation :

- modélisation progressive de la dynamique du bipède, en vue de son contrôle/commande;
- mise en œuvre du scénario retenu, par l'intégration incrémentale des résultats du contrôle/commande du robot bipède, avec prise en compte des modes dégradés;

[Aub91] J.-P. AUBIN, *Viability Theory*, Birkhäuser, Boston, Basel, Berlin, 1991.

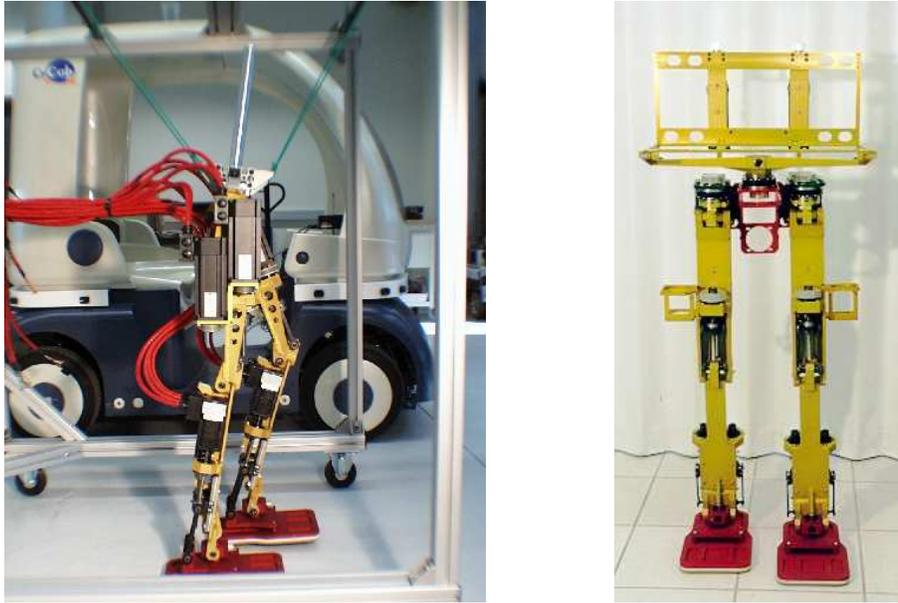


FIG. 5 – Prototype de robot bipède : motorisé, à huit degrés de liberté (à gauche), et squelette vu de face (à droite)

- définition de protocoles de maintenance et de sécurité;
- conception et mise en oeuvre d'un système de sécurité, à la fois logiciel (sous ORCCAD) et matériel, de ce système robotique complexe.

6.2 Perception et simulation

Participants : N. Andreff, L. France, B. Espiau, B. Lamiroy [MOVI].

6.2.1 Asservissement visuel

Dans le cadre de l'asservissement visuel (Action de Recherche Coopérative (ARC) AVEC), nous nous sommes penchés sur la modélisation des droites de l'espace [9]. Nous avons montré que les modélisations présentes dans la littérature n'étaient pas idéales. En nous inspirant des travaux de la communauté de vision par ordinateur, nous avons dérivé, à partir des coordonnées euclidiennes de Plücker d'une droite, une notion de coordonnées euclidiennes de Plücker normées. Elle permet de définir une nouvelle notion d'alignement des droites aussi bien dans l'image, comme dans les approches précédentes, qu'en orientation dans l'espace. À partir de cet alignement mixte (2D-3D), nous avons pu bâtir une loi de commande explicite et partiellement découplée en rotation et translation. Nous avons ainsi pu faire une analyse globale de la convergence de cette commande et en exhiber les singularités.

Cette loi de commande a été appliquée, dans le cadre du projet VIGOR, au positionnement d'une caméra par rapport à un trièdre orthogonal (Figure 6). Cette configuration de droites



FIG. 6 – *Un trièdre orthogonal, pièce de navire à souder.*



FIG. 7 – *La tâche engendrée par un pointeur laser rigidement lié à une caméra est contrainte à se déplacer sur une droite épipolaire laser.*

étant invariante par translation le long de son centre, nous avons proposé une méthode originale d'observation de la profondeur. Elle se fait, par similarité avec la stéréovision, par adjonction d'un pointeur laser non étalonné. Ce pointeur étant fixe par rapport à la caméra, la projection du point laser dans l'image est contrainte à se déplacer le long d'une droite épipolaire (Figure 7). L'étude de ces déplacements permet de définir la loi de commande en profondeur.

Nous avons par ailleurs étudié, toujours dans le cadre du projet VIGOR, en collaboration avec Bart Lamiroy du projet MOVI, l'asservissement visuel en stéréovision [32]. Nous avons mis en évidence les conditions dans lesquelles la simple juxtaposition des signaux des deux images est équivalente à la projection de ces mêmes informations sur la contrainte épipolaire. Nous avons également évalué la précision de la méthode proposée par rapport à un asservissement monoculaire classique (voir Figure 8).

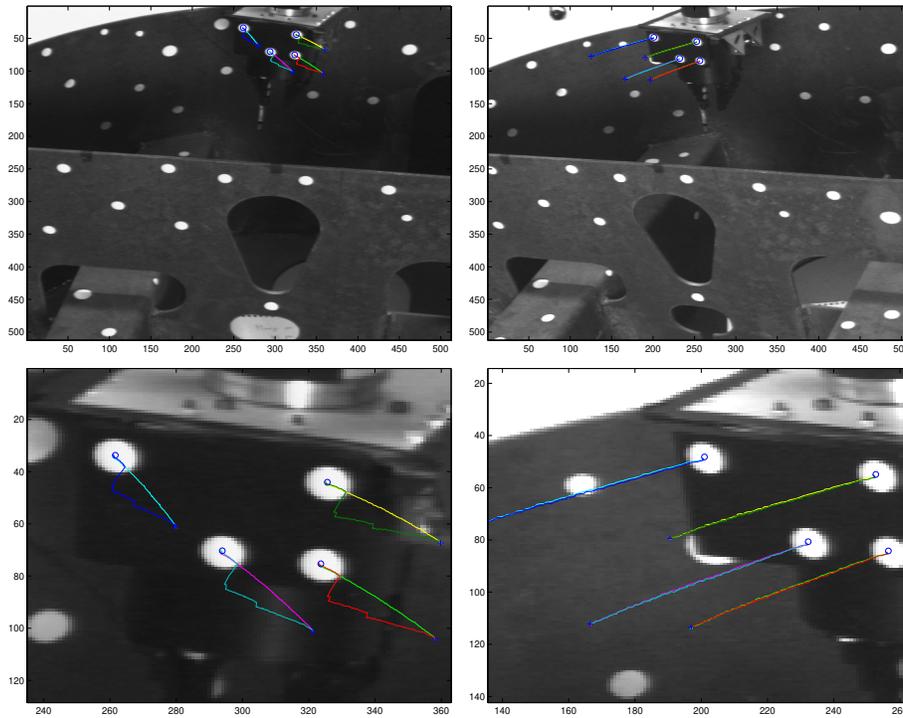


FIG. 8 – *Comparaison de l'asservissement visuel stéréoscopique et de l'asservissement visuel monoculaire (image de droite) : la trajectoire dans l'image de droite est identique ; en revanche, la trajectoire dans l'image de gauche est rectiligne dans le cas stéréoscopique (traduisant une trajectoire rectiligne dans l'espace) mais pas dans le cas monoculaire. En haut : les images complètes avec les trajectoires dans l'image obtenues par chacun des deux asservissements ; en bas : un zoom sur les zones d'intérêt.*



FIG. 9 – *Cycle de marche pour le robot bipède.*

6.2.2 Simulation graphique 3D d'un robot bipède

Un des thèmes de l'équipe porte sur la réalisation d'un simulateur pour un robot anthropomorphe bipède dans un environnement. Les roboticiens ont besoin d'un tel simulateur pour valider leurs lois de contrôle développées pour un robot bipède réel. Un modèle de robot a été construit puis animé avec des cycles de marche issus de la capture de mouvements (Figure 9). Ces mouvements capturés étant constitués uniquement de cycles de marche, nous les avons complétés en calculant les mouvements transitoires de départ et d'arrêt du robot. Pour cela, nous avons utilisé une méthode basée sur l'interpolation de polynômes cubiques et respectant des contraintes liées aux mouvements des pieds du robot (cohérence du mouvement et non pénétration dans le sol). Nous avons obtenu ainsi des mouvements complets, qui peuvent servir de référence pour le robot, et pourront être exploités par les roboticiens lors du calcul de la commande du robot. Dans notre cas, nous les avons utilisés pour faire marcher le robot dans son environnement.

Comme cet environnement est inconnu du robot, nous avons modélisé sa perception au moyen de capteurs proximétriques, qui donnent la distance entre le robot et les objets locaux. Nous avons utilisé des techniques de synthèse d'images afin de calculer cette distance aux obstacles. Nous avons modélisé également les imperfections des vrais capteurs pour tenir compte de l'orientation des objets et du bruit des capteurs. Cette perception locale est utile pour la détection d'obstacles, afin d'éviter toute collision du robot avec son environnement. Plus précisément, le robot doit modifier son comportement en fonction de l'environnement. Par exemple, lorsqu'il rencontre des escaliers, il doit pouvoir les monter ou les descendre. Il lui faut donc une certaine reconnaissance des objets (Figure 10). L'interprétation directe des sorties capteurs qui donne la distance aux objets est insuffisante. Nous avons calculé alors le profil de l'environnement perçu à partir des distances détectées par les capteurs, et nous avons proposé un algorithme de reconnaissance d'objets basé sur ce profil.

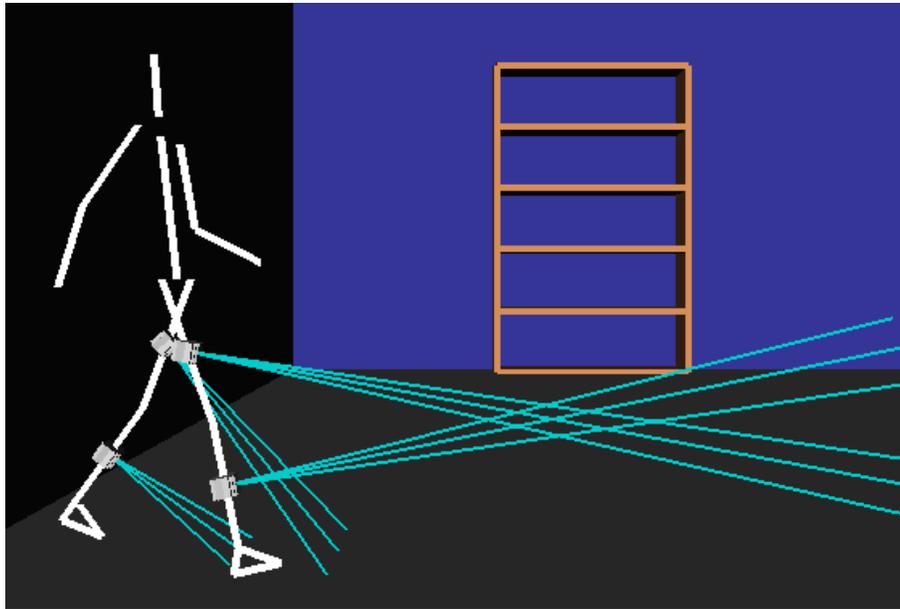


FIG. 10 – Configuration de capteurs sur le robot, avec 3 capteurs par support.

Une fois que ces fonctions de marche, de perception et de reconnaissance sont fournies au robot, nous avons étudié le placement des capteurs sur le robot, sous certaines hypothèses, afin de déterminer la meilleure configuration pour laquelle le nombre de capteurs est minimal pour une détection maximale des objets dans l'environnement. Puis, nous avons réalisé une simulation complète du robot bipède évoluant dans un environnement, dont le mouvement de marche est contrôlé en fonction de ce qui est perçu par les capteurs.

6.3 Contrôle/commande de systèmes robotiques complexes

Participants : A. Girault, R. Pissard-Gibollet [M.R.], D. Simon, A. Vilchis, S. Yovine [VERIMAG].

6.3.1 Spécification et implantation d'une commande sûre d'un bras manipulateur

Dans la commande des bras manipulateurs, il est bien connu que, si on ne prend pas de précautions particulières au voisinage des singularités dans l'utilisation du modèle différentiel inverse, les vitesses et les accélérations articulaires demandées tendent vers l'infini, ce qui implique un risque de détérioration du mécanisme et de son environnement.

Il apparaît que le concept de singularité peut être appréhendé dans un sens plus large que celui de singularité géométrique. Les méthodes de régulation associées peuvent également être considérées d'un point de vue plus général, par la minimisation d'une fonction de coût. Une telle approche est étudiée dans [6] en utilisant le formalisme de la fonction de tâche.

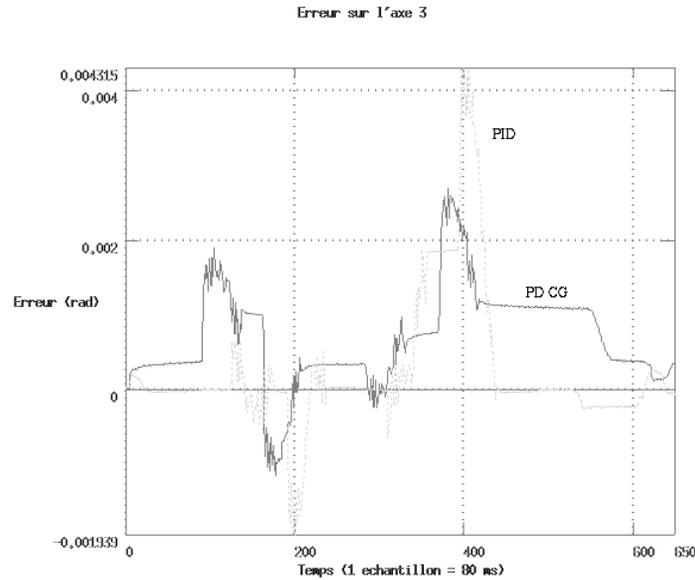


FIG. 11 – Comparaison de l'erreur sur l'axe 3 (PID vs PD+CG) pour le robot Rx90.

En utilisant cette approche, plusieurs lois de commande ont été développées dans l'environnement ORCCAD pour le bras manipulateur Staubli Rx90 des Moyens Robotiques [40]. En particulier, dans le cadre du stage de DEA d'A. Vilchis, une loi de commande proportionnelle/dérivée avec compensation de gravité dans l'espace articulaire (utilisant une modélisation partielle du modèle dynamique du bras) a été testée en montrant des erreurs de poursuite plus faibles qu'une commande PID classique (figure 11). La même loi de commande sans l'espace opérationnel n'a pu être testée qu'en simulation. Dans tous les cas de figure, l'accent a été mis sur la sûreté de fonctionnement en utilisant le cadre des traitements d'exception fourni par ORCCAD.

6.3.2 Etude de la stabilité d'une loi de commande pour véhicules autonomes

Cette activité se fait en coopération avec S. Yovine (VERIMAG). Nous avons étudié en détail la loi de commande pour véhicules autonomes. Le système que nous avons étudié est constitué de deux véhicules, un véhicule leader et un véhicule suiveur que nous commandons en accélération. La nouveauté par rapport à l'année dernière a consisté à prouver la stabilité de notre loi de commande sous trois conditions : que l'accélération du véhicule leader soit *toujours* bornée par $[a_{min}, a_{max}]$, que $a_{min} > -\lambda$, où λ est le gain de notre loi de commande, et que l'accélération du véhicule suiveur soit *initialement* bornée par $[a_{min}, a_{max}]$. Ce résultat amène à choisir $\lambda > -a_{min}$! [28].

6.4 Contrôle/commande : validation et mise en œuvre

Participants : F. Baccelli [ENS], B. Gaujal [Loria], A. Girault, C. Lavarenne [SOSSO],

R. Mateescu [VASY], R. Pissard-Gibollet [M.R.], E. Rutten, M. Sighireanu, D. Simon, Y. Sorel [SOSSO].

On peut noter ici que les Sections 6.4.2, 6.4.3 et 6.4.5 décrivent des travaux liés à l'action TOLERE (code réparti tolérant aux pannes pour systèmes embarqués), menée en collaboration entre les projets BIP et SOSSO. Son objet est l'étude de techniques permettant d'améliorer la fiabilité de code temps-réel réparti. Les principaux points abordés concernent la coopération des aspects flot de données et flot de contrôle, l'automatisation de la génération fiable de spécifications de haut niveau et l'étude de mécanismes d'exécution tolérants aux pannes dans un contexte temps-réel distribué. L'objectif à terme est le couplage d'ORCCAD comme outil de spécification d'application et de SYNDEX comme outil de génération de code réparti et optimisé.

6.4.1 Génération de code ESTEREL pour ORCCAD

Rappelons que l'environnement ORCCAD offre deux niveaux de conception de systèmes robotiques :

- La Tâche-Robot (TR) met en œuvre une loi de commande en combinant plusieurs modules algorithmiques, une ou plusieurs ressources physiques et un automate réactif décrivant son comportement temporel. L'ensemble est spécifié sous la forme d'un réseau flot de données. Enfin, l'automate de contrôle est généré automatiquement à partir de la description des pré-conditions, post-conditions et exceptions de la TR. Le formalisme utilisé pour cet automate est le langage synchrone ESTEREL [BG92].
- La Procédure-Robot (PR) combine logiquement et temporellement plusieurs TRs et PRs existantes pour réaliser une mission robotique complexe. Les PRs sont programmées en ESTEREL. Cela permet de spécifier leur comportement logique et temporel de façon claire et intuitive, tout en bénéficiant des bonnes propriétés sémantiques du langage dues à l'hypothèse de synchronisme.

Dans la version actuelle d'ORCCAD, le code généré est mixte (ESTEREL et C++). Pour une application ayant n TRs, $2n + 2$ tâches temps-réel C++ sont générées :

- La tâche `horloge` est l'horloge temps-réel.
- La tâche `manager` est écrite en C++. Elle est réveillée dès que la file d'attente n'est plus vide. Elle invoque alors les fonctions d'entrée de l'automate AA et lui fait faire une transition. Cet automate est la combinaison du code ESTEREL de la PR principale du programme ORCCAD et du code ESTEREL de toutes les autres PRs et de toutes les TRs.
- La Tâche-Module (TM) est écrite en C/C++. Elle exécute le code flot de données de la TR. Par défaut il y en a une par TR mais dans le cas du multi-cadence, il y en aurait plusieurs.

[BG92] G. BERRY, G. GONTHIER, «The ESTEREL Synchronous Programming Language: Design, Semantics, Implementation», *Science of Computer Programming* 19, 2, 1992, p. 87–152.

- La tâche `controller` est écrite en C++. Elle est chargée de la création et destruction de la ou des TMs de la TR.

Ces tâches se synchronisent au moyen de sémaphores. Ce schéma est assez complexe et c'est pourquoi nous avons proposé une nouvelle génération de code entièrement en ESTEREL. Après tout, ESTEREL est censé être le langage idéal pour programmer ce type d'applications temps-réel. L'idée est donc de remplacer toutes les tâches C++ en parallèle par des modules ESTEREL, et de remplacer les synchronisation par sémaphores par des synchronisations par diffusion synchrone. La diffusion synchrone a l'avantage d'être instantanée et de donner des primitives de synchronisation très simples à comprendre et à mettre en œuvre.

6.4.2 Sur le couplage d'ORCCAD avec SYNDEX

Mihaela Sighireanu a réalisé un traducteur du format DC vers le format d'entrée de SYNDEX. Ce traducteur utilise le méta-compilateur SYNTAX-FNC2 et s'appelle DC2SDX¹. Par ailleurs, Alain Girault réalise actuellement un nouveau générateur de code pour ORCCAD : il produira uniquement du code ESTEREL au lieu d'un code mixte ESTEREL et C++ actuellement. Ceci permettra ainsi de générer du code DC à partir d'un programme ORCCAD.

6.4.3 Tolérance aux pannes

Mihaela Sighireanu a établi un état de l'art sur les problèmes et les méthodes de la tolérance aux pannes. A partir de ce document, nous avons fait avec Yves Sorel et Christophe Lavarenne les choix de conception suivants. Ces choix sont dictés par notre domaine d'application, à savoir les systèmes réactifs embarqués :

- Les pannes que l'on veut traiter sont accidentelles, physiques, internes, opérationnelles et permanentes.
- La tolérance aux pannes est réalisée par compensation des erreurs, c'est-à-dire qu'une fois détectée, une erreur doit être masquée grâce à la redondance interne du système.
- Le type de redondance utilisé est logicielle et temporelle.

Notre but étant de produire *automatiquement* du code réparti tolérant aux pannes, nous avons fait les choix de conceptions supplémentaires suivants :

- Les pilotes d'entrée/sortie ne sont jamais répliqués et ont chacun un processeur pré-assigné pour la répartition.
- Seules les fautes permanentes des liens de communication sont prises en compte. Ces fautes sont détectées au moyen de « chiens de garde » dont la durée est calculée en fonction des paramètres des blocs d'instructions (durée d'exécution ...).

En fin de compte, nous avons mis au point deux nouvelles heuristiques de répartition sous SYNDEX, afin de produire du code redondant. La première variante utilise la redondance logicielle des blocs d'instructions et des communications. La seconde utilise la redondance temporelle des communications.

1. DC2SDX : <http://www-rocq.inria.fr/who/Mihaela.Sighireanu/DC2SDX>

6.4.4 Analyse de systèmes temps-réels préemptifs par l'algèbre $(\max, +)$

Cette activité se fait en coopération avec F. Baccelli (ENS) et B. Gaujal (Loria). Les modules communicants et synchronisés de façon plus ou moins lâche utilisés pour construire les Tâches-Commande (TC) de programmes ORCCAD s'implantent sous forme de tâches temps-réel généralement périodiques, gérées par un système d'exploitation à priorités fixes. Cet ensemble de tâches périodiques peut être modélisé sous forme d'un Graphe d'Événements Temporisés (GET), ayant un modèle linéaire dans l'algèbre $(\max, +)$. L'ordonnanceur temps-réel est représenté par une fonction de contraction du temps appliquée aux tâches préemptées, ce qui permet de garder une structure de GET et donc d'appliquer les résultats classiques de $(\max, +)$.

Un certain nombre de problèmes relatifs au comportement temporel de l'ensemble de modules peuvent alors être résolus de façon analytique : nous pouvons prouver que le système atteint bien un régime périodique après une période transitoire, dont nous pouvons calculer la durée. La durée du cycle en régime permanent est également calculable. Enfin, le respect des échéances temporelles de chacune des tâches peut aussi être prouvé.

Certaines hypothèses demandent encore à être relâchées (pour l'instant, la méthode s'applique à une implantation monoprocesseur et à des tâches à requête sur échéance). Les problèmes de partage de ressources communes pouvant aboutir à une inversion de priorité ne sont pas traités pour l'instant. Cependant, la méthode prend d'emblée en compte les problèmes de synchronisation et de précedence, et n'est pas limitée à une politique d'ordonnement particulière [37].

6.4.5 Vérification de propriétés de programmes ORCCAD

Un programme ORCCAD s'articule autour d'un contrôleur à événements discrets écrit en ESTEREL et d'une ou plusieurs loi de commande. La vérification formelle porte sur le contrôleur. Celui-ci est modélisé par un automate d'états fini, dont les étiquettes des transitions portent les actions de test des signaux d'entrée et d'émission des signaux de sortie. Les propriétés que l'on désire vérifier font intervenir le séquençement, à la fois des étiquettes dans l'automate, et des actions à l'intérieur des étiquettes. Par exemple, on veut vérifier que l'enchaînement des lois de commande est tel que le robot est toujours commandé par une et une seule loi. Cela revient à vérifier que les actions !"Activate_ROBOT" et ?"CmdStopOK_ROBOT" alternent pendant toute la durée du programme, et que toute séquence commence par un !"Activate_ROBOT" et finit par un ?"CmdStopOK_ROBOT". Or ces deux actions peuvent apparaître aussi bien ensembles dans une seule étiquette de l'automate, que séparées dans des étiquettes différentes.

Pour cela, Mihaela Sighireanu et Radu Mateescu ont défini une logique temporelle spécifique, appelée « μ -calcul régulier sans alternance ». C'est une extension du « μ -calcul modal sans alternance » [EL86], étendu avec des prédicats sur les étiquettes et des expressions régulières sur les séquences d'étiquettes. L'avantage de cette logique temporelle est de permettre l'écriture des formules qui nous intéressent, à savoir les formules portant sur le séquençement des actions, aussi bien dans les étiquettes que dans l'automate. Mihaela et Radu ont réalisé un évaluateur qui prend en entrée un automate et d'une formule du μ -calcul régulier sans alternance, en fait

[EL86] E. A. EMERSON, C.-L. LEI, « Efficient Model Checking in Fragments of the Propositional Mu-Calculus », in : *Proceedings of the 1st LICS*, p. 267-278, 1986.

le produit synchrone et résout «à la volée» le système d'équations booléennes obtenu par le produit. L'outil correspondant, appelé `evaluator` version 3.0, met en œuvre un algorithme particulièrement efficace, linéaire par rapport à la taille de la formule (nombre d'opérateurs) et la taille de l'automate (nombre d'états et de transitions). L'outil `evaluator` est distribué avec CADP (Cæsar Aldébaran Development Package²) et est prévu pour fonctionner avec.

Enfin, nous avons défini plusieurs propriétés intéressantes à vérifier, et nous les avons écrites en μ -calcul régulier sans alternance. Il restera à les intégrer à l'interface utilisateur d'ORCCAD.

En parallèle est menée une réflexion sur une structuration des propriétés qu'il est intéressant de vérifier, selon deux points de vue : d'une part en termes de techniques disponibles pour les évaluer (analyse statique, analyse d'invariants par compilation synchrone, ou analyse comportementale) ; d'autre part en termes de leur signification vis-à-vis des structures manipulées dans notre environnement : propriétés génériques, à vérifier pour toute mission, et propriétés spécifiques. On souhaite pouvoir les formuler en termes des entités manipulées (lois de commandes, exceptions, ...) plutôt qu'en termes intrinsèques au modèle sous-jacent (logique temporelle ou μ -calcul). Ce point mène vers la définition d'un langage spécialisé pour les propriétés qui soit en correspondance avec le codage des missions en ESTEREL, et aussi vers une étude de ce dernier pour favoriser leur caractère «vérifiable».

Dans la même perspective, nous nous intéressons aux techniques de synthèse de contrôleurs, notamment celles liées aux modèles synchrones, et les méthodes de calcul et d'exécution des contrôleurs.

6.5 Langages réactifs

Participants : C. André [Université de Nice], F. Boulanger [Ecole Supérieure d'Electricité], A. Girault, F. Jiménez-Fraustro [EP-ATR, INRIA-RENNES], P. Le Guernic [EP-ATR, INRIA-RENNES], E. Rutten.

6.5.1 Formalisation des machines d'exécution pour programmes synchrones

Ce travail est réalisé dans le but de rédiger un chapitre du livre «Langages synchrones» qui sera publié par Hermès, sous la direction de Paul Caspi³ et Nicolas Halbwachs³.

Exécuter un programme synchrone est une tâche délicate en raison de la différence fondamentale existant entre le programme synchrone et l'environnement asynchrone dans lequel il est exécuté. C'est le rôle de la machine d'exécution, qui doit observer l'état courant du système (acquisition), décider (exécution du programme synchrone) et agir sur l'environnement (action). La formalisation des machines d'exécution nous a amenés à définir deux modèles pour la boucle d'exécution qui en constitue le cœur. Dans le modèle général, chaque événement d'entrée déclenche une nouvelle réaction du programme. Dans le modèle périodique, les réactions du programme sont déclenchées à chaque «top» d'une horloge temps réel périodique («tick» en anglais). Pour chacun de ces deux modèles, nous avons spécifié les diverses tâches devant être exécutées en parallèle ainsi que leur mode de synchronisation avec la tâche de la

2. CADP: <http://www.inrialpes.fr/vasy/cadp>

3. CNRS, laboratoire VERIMAG

boucle d'exécution. Deux problèmes se posent pour lesquels nous avons établi des conditions de correction : la cohérence des entrées (possibilité pour une entrée donnée d'être remise à jour *pendant* la transition du programme) et la validité de l'hypothèse de synchronisme (le programme est-il *plus rapide* que l'environnement?). Enfin, nous avons recensé diverses mises en œuvre de machines d'exécution, à la fois centralisées et réparties.

6.5.2 Langages de programmation d'automates

La programmation des systèmes de contrôle/commande dans les automatismes industriels relève des possibilités d'élargissement des domaines d'applications d'ORCCAD. La problématique y fait intervenir la spécification de systèmes hybrides (commande continue et contrôle discret), la gestion de la complexité des contrôleurs, la sécurité critique nécessitant des méthodes et outils de validation et vérification, et les plateformes de mise en œuvre diverses et éventuellement distribuées. Dans ce domaine, la programmation se fait souvent sur la base d'automates programmables industriels (API). Il existe des standards internationaux la concernant, et particulièrement la norme CEI 1131 ^[CEI93] de la Commission Électrotechnique Internationale. Ils proposent des langages de programmation, parmi lesquels les *Ladder Diagrams* (LD), les Diagrammes de Fonction en Séquence (*Sequential Function Charts* SFC, proche du grafcet), et le Texte Structuré (ST), un langage impératif séquentiel, de la famille de Pascal.

Nous menons avec F. Jiménez-Fraustro et Paul Le Guernic (projet EP-ATR, INRIA-RENNES) une activité visant l'intégration de ce standard de spécification et de l'environnement de programmation synchrone SIGNAL : il s'agit d'établir un modèle de ces langages, visant à offrir à de telles spécifications l'accès aux outils d'analyse et mise en œuvre des techniques synchrones. On a construit un modèle en SIGNAL du langage ST [31, 30], réutilisant des résultats de l'étude sur l'interopérabilité entre Statemate et SIGNAL^[Bea99] [21, 11]. Le travail en cours comprend la modélisation en Signal des éléments du langage SFC, et l'analyse formelle et automatisée de propriétés d'un diagramme SFC, telles que : la stabilité du diagramme SFC, la possibilité de réactivation d'une étape ou la possibilité de blocage.

7 Contrats industriels (nationaux, européens et internationaux)

7.1 VIGOR

Participants : B. Espiau, N. Andreff.

BIP participe avec MOVI au projet LTR VIGOR (*Visually Guided Robots Using Uncalibrated Cameras*), dont les partenaires sont : l'INRIA, l'université Hébraïque de Jérusalem, l'université de Karlsruhe, l'université de Cambridge, la société SINTERS et les chantiers navals d'Odense (Danemark).

[CEI93] CEI, « Norme Internationale – Automates programmables : Langages de programmation », *rapport de recherche n° IEC 1131 partie 3*, CEI/IEC (Commission Électrotechnique Internationale/International Electrotechnical Commission), 1993.

[Bea99] J.-R. BEAUVAIS, *Modélisation de STATECHARTS en SIGNAL pour la conception de systèmes critiques temps-réel*, Thèse de Doctorat en Informatique, IFSIC, Université de Rennes 1, février 1999.

7.2 TELEDIMOS

BIPet MOVifont partie du consortium IST TELEDIMOS (*Telepresence simulation platform supporting civil works machines in remote dismantling waste disposal and large scale demining operations*). Ce projet de 2 ans est prévu pour démarrer début 2000. Les partenaires sont les laboratoires ou associations *Hellenic Technology of Robotics*, BNFL, CRIIF, EARLR, et l'INRIA ; et les entreprises de construction *Hellenic technodomiki* et *Techniki*, et JCB.

7.3 Transfert industriel

Le projet participe au transfert du savoir-faire de l'INRIA en contrôle/commande temps-réel, en particulier autour d'ORCCAD, vers la société ATHYS en cours de création.

8 Actions régionales, nationales et internationales

8.1 Actions nationales

8.1.1 GdR-PRC "Commande de robots à pattes"

B. Espiau est co-responsable avec Carlos Canudas de Wit du projet GdR-PRC du CNRS "Commande de robots à pattes", et Elena Panteley y participe.

8.1.2 GdR Automatique

Elena Panteley participe au projet "De la commande adaptative à la stabilisation des systèmes variants dans le temps et de retour".

8.1.3 PSLC : Pôle Systèmes et Logiciels Critiques

Le PSLC est un projet qui a été déposé auprès du Pôle Numérique de la Région Rhône-Alpes, sous l'impulsion du laboratoire CRNS VERIMAG, en regroupant plusieurs équipes de recherche ainsi que plusieurs entreprises de la Région Rhône-Alpes. Il comporte des projets pilotes ayant pour finalité la création d'activité économique.

8.1.4 Groupe GRAFCET

E. Rutten participe à un groupe de travail de l'EEA consacré au Grafcet, et plus largement aux formalismes liés aux automatismes industriels⁴.

8.1.5 Collaborations internes à l'Inria

- Avec le service des Moyens Robotiques de l'UR, sur ORCCAD et sur la réalisation du bipède.

4. <http://www.lurpa.ens-cachan.fr/grafcet/>

- Avec le projet MOVI, sur l’asservissement visuel non calibré (doctorant : N. Andreff). Le projet LTR VIGOR, à l’initiative de MOVI, est également une opération commune sur ce sujet.
- Avec le projet IMAGIS sur le thème de l’animation (doctorante : L. France)
- Avec le projet VASY, sur la vérification des propriétés temporelles des tâches robotiques en ORCAD.
- Avec le projet SOSSO dans le cadre de l’action incitative TOLERE.
- Avec le projet EP-ATR, INRIA-RENNES, sur la modélisation synchrone des langages de programmation d’automates pour le contrôle/commande.
- Avec le projet MACSI, INRIA-LORRAINE, ainsi qu’avec l’ENS, sur l’utilisation des algèbres (max,+).

8.1.6 Collaborations avec d’autres laboratoires

- L’équipe BIP est à l’origine du projet multi-laboratoires BIP ayant pour objectif la réalisation d’un robot bipède mécanique et de sa commande [4]. Les partenaires en sont le LAG (Carlos Canudas de Wit), le laboratoire de mécanique des solides de Poitiers (Guy Bessonnet et Philippe Sardain) et le laboratoire des matériaux à usage sportif de Poitiers (Alain Junqua). Deux prototypes de robots à 15 degrés de liberté ont été réalisés cette année, aboutissement de cinq années de collaboration (figure 5). Avec le LAG, la collaboration s’élargit à des domaines plus amont et est très active : doctorants en commun, groupes de travail, séminaires.
- Le projet participe à l’Action de Recherche Coopérative (ARC) AVEC (Asservissement visuel en environnement complexe) qui voit coopérer les projets BIP et MOVI (INRIA-RHÔNE-ALPES), ICARE (INRIA-SOPHIA-ANTIPOLIS) et VISTA (INRIA-RENNES).
- Le projet participe, Alain Girault en étant le coordinateur, à l’Action de Recherche Coopérative (ARC) TOLERE (Code réparti tolérant aux pannes pour systèmes embarqués), dont les partenaires sont les projets BIP et SOSSO (INRIA-ROCQUENCOURT).

8.2 Actions européennes

- Le projet LTR VIGOR a débuté en février 1998.
- Le projet BIP est membre depuis l’automne 1998 du réseau européen CLAWAR (Climbing and Walking Robots).
- B. Espiau travaille avec R. Boulic de l’École Polytechnique Fédérale de Lausanne sur la commande posturale des robots.

8.3 Actions internationales

- A. Goswami était en séjour de longue durée aux USA (Université de Pennsylvanie) jusqu'en mai 1999.
- F. Génot était en séjour post-doctoral au MIT (*Mech. Eng. Lab., Prof S. Dubowsky*) jusqu'en février 1999.
- B. Espiau est l'un des cinq membres français de la mission européenne au Japon de l'IARP "Human Friendly Robotics" en novembre 1999.

9 Diffusion de résultats

9.1 Animation de la Communauté scientifique

B. Espiau est membre des comités de programme pour ICAR 1999 et IEEE ICRA 2000 ; rapporteur de l'habilitation à diriger des recherches de Philippe Martinet ; rapporteur des thèses de doctorat de Benoît Perrin, Eric Collevet et Mostafa Rostami ; membre des jurys de thèse de doctorat de Laurent Royer, Nicolas Turro, et Laure France. Sont prévus avant fin décembre les jurys de thèse de Soraya Arias, Nicolas Andreff, Bilal El-ali, d'habilitation à diriger des recherches de Eric Rutten.

E. Rutten a été membre des comités de programme pour *Euromicro Conference on Real-Time Systems, ERTS'99* et *Salon Real-Time Systems RTS 2000*.

En dehors des conférences citées en bibliographie, les chercheurs du projet ont participé ou présenté leurs travaux dans les manifestations suivantes :

- Alain Girault et Eric Rutten ont participé au *Workshop on synchronous languages*, à Gandia (Espagne), en décembre 1998, et à Hyères, en novembre-décembre 1999.
- Daniel Simon a participé à l'École d'été ETR'99 "Applications, Réseaux et Systèmes", Poitiers, septembre 1999.
- Nicolas Andreff a donné un séminaire à la Réunion du GT5 du Gdr-PRC ISIS sur le thème 'Commande référencée vision'.
- Elena Panteley a donné un séminaire invité au LSS, Supélec, Gif s/Yvette en octobre 1999.
- Bernard Espiau a donné une conférence sur les robots marcheurs aux Journée Nationales de la Recherche en Robotique, en octobre 1999.

Le projet a invité à donner un séminaire : Edward Lee (UC Berkeley, USA), Florence Maraninchi (VERIMAG, Grenoble), Marco Antoniotti (PARADES, Rome, Italie), Nicolas Turro (INRIA-SOPHIA-ANTIPOLIS), Mark Spong (University of Illinois at Urbana-Champaign).

9.2 Exposition Universelle

Le projet BIP participe, avec le Laboratoire de Mécanique des Solides (LMS, Poitiers), au pavillon français à l'Exposition Universelle 2000, en présentant le robot bipède conçu et développé conjointement. L'exposition a lieu à Hanovre du 1er juin au 31 octobre 2000, dans le cadre de la thématique : Homme -Nature - Technologie. La démonstration quotidienne du robot bipède dans le Pavillon Français de l'exposition, pendant cinq mois, nécessite un travail préparatoire conséquent.

9.3 Enseignement

- B. Espiau : commande des robots (École des Mines de Paris) ; commande référencée capteurs (DEA IVR, Grenoble).
- N. Andreff : monitorat en informatique à l'INPG.
- L. France bénéficie d'un demi poste d'ATER à l'université Joseph Fourier.
- A. Girault : co-encadrement (avec R. Pissard-Gibollet (M. R.) et S. Sekhavat (SHARP)) d'un stage de magistère d'Informatique : réalisation d'un simulateur fonctionnel de robot.
- A. Girault : théorie des automates en 1^{re} année du département des télécommunications de l'INPG.
- B. Espiau, A. Girault et D. Simon donnent une série de trois conférences de deux heures en robotique à l'ENS de Lyon.

10 Bibliographie

Ouvrages et articles de référence de l'équipe

- [1] J. BORRELLY, E. C. MANIÈRE, B. ESPIAU, K. KAPellos, R. PISSARD-GIBOLLET, D. SIMON, N. TURRO, «The ORCCAD Architecture», *International Journal of Robotics Research* 17, 4, 1998, p. 338–359.
- [2] P. CASPI, J.-C. FERNANDEZ, A. GIRAULT, «An Algorithm for Reducing Binary Branchings», in : *Fifteenth Conference on the Foundations of Software Technology and Theoretical Computer Science, FST&TCS'95*, P. Thiagarajan (éditeur), LNCS, 1026, Springer-Verlag, Bangalore, India, décembre 1995.
- [3] B. ESPIAU, F. CHAUMETTE, P. RIVES, «A New Approach to Visual Servoing in Robotics», *IEEE Trans. on Robotics and Automation* 8, 3, 1992.
- [4] B. ESPIAU, THE BIP-TEAM, «Bip: a joint project for the Development of an Anthropomorphic Biped Robot», in : *Int. Conf. on Advanced Robotics*, Monterey, CA, USA, juillet 1997.
- [5] A. GOSWAMI, B. THUILOT, B. ESPIAU, «Compass-Like Biped Robot Part I: Stability and Bifurcation of Passive Gaits», *Rapport de Recherche n° 2996*, INRIA, octobre 1996, à paraître dans *Int. Journal of Robotics Research*.

- [6] C. SAMSON, B. ESPIAU, M. L. BORGNE, *Robot Control: The Task Function Approach*, Oxford Science Publications, 1991.
- [7] D. SIMON, B. ESPIAU, E. CASTILLO, K. KAPELLOS, « Computer-Aided Design of a Generic Robot Controller Handling Reactivity and Real-Time Control Issues », *IEEE Transactions on Control Systems Technology* 1, 4, décembre 1993, <http://www.inrialpes.fr/iramr/pub/0rccad>.

Thèses et habilitations à diriger des recherches

- [8] B. E. ALI, *Contribution à la Commande du Centre de Masse d'un Robot Marcheur*, thèse de doctorat, Institut National Polytechnique de Grenoble, 1999, à paraître.
- [9] N. ANDREFF, *Asservissement visuel à partir de droites et auto-étalonnage pince-caméra*, thèse de doctorat, Institut National Polytechnique de Grenoble, 1999, à paraître.
- [10] L. FRANCE, *Simulation graphique d'un robot bipède dans un environnement structuré*, Thèse de doctorat en informatique, Université Joseph Fourier - Grenoble I, octobre 1999.
- [11] E. RUTTEN, *Programmation sûre des systèmes de contrôle/commande : le séquençement de tâches flot de données dans les langages réactifs*, Habilitation à diriger des recherches, Université de Rennes 1, décembre 1999, à paraître.

Articles et chapitres de livre

- [12] A. LORIA, T. FOSSEN, E. PANTELEY, « Global output-feedback dynamic positioning of ships: A cascaded approach », *IEEE Transactions on Control Systems Technology*, 1999, à paraître.
- [13] P. CASPI, A. GIRAULT, D. PILAUD, « Automatic distribution of reactive systems for asynchronous networks of processors », *IEEE Transactions on Software Engineering* 25, 3, may-june 1999, p. 416–427.
- [14] A. GIRAULT, B. LEE, E. LEE, « Hierarchical finite state machines with multiple concurrency models », *IEEE Transactions on Computer-Aided Design* 18, 6, juin 1999, p. 742–760.
- [15] A. GOSWAMI, « Postural stability of biped robots and the foot-rotation indicator (FRI) point », *International Journal of Robotics Research* 18, 6, juin 1999, p. 523–533.
- [16] K. KAPELLOS, D. SIMON, M. JOURDAN, B. ESPIAU, « Task Level Specification and Formal Verification of Robotics Control Systems: State of the Art and Case Study », *Int. Journal of Systems Science* 30, 11, november 1999, p. 1227–1245.
- [17] A. LORIA, E. PANTELEY, « Force/tracking control of constrained manipulators without velocity measurements », *IEEE Transactions on Automatic Control* 44, 1999, p. 1407–1412.
- [18] F. MULTON, L. FRANCE, M. CANI-GASCUEL, G. DEBUNNE, « Computer Animation of Human Walking: a Survey », *Journal of Visualization and Computer Animation* 10, 1999, p. 39–54.
- [19] E. PANTELEY, A. LORIA, A. SOKOLOV, « Global uniform asymptotic stability of nonlinear nonautonomous systems: Application to a turbo-diesel engine », *European Journal of Control*, 1, 1999, à paraître.

Communications à des congrès, colloques, etc.

- [20] N. ANDREFF, R. HORAUD, B. ESPIAU, « On-line Hand-Eye Calibration », *in: Proceedings of the Second International Conference on 3-D Digital Imaging and Modeling (3DIM'99)*, Ottawa, Canada, octobre 1999.
- [21] J.-R. BEAUVAIS, R. HOUEBINE, P. LE GUERNIC, E. RUTTEN, T. GAUTIER, « Une modélisation de STATECHARTS et ACTIVITYCHARTS en SIGNAL », *in: Actes du 2^{ème} Congrès sur la Modélisation des Systèmes Réactifs, MSR'99*, Cachan, France, 24 – 25 Mars, Hermès, p. 213–222, 1999.
- [22] B. EL ALI, B. ESPIAU, « Controlling the Center of Mass of a Biped Robot when Walking Up and Down Stairs », *in: Proceedings of Biomechanics meets Robotics, Modelling and Simulation of Motion*, Heidelberg, Allemagne, novembre 1999.
- [23] B. ESPIAU, I. GUIGUES, R. PISSARD-GIBOLLET, « Can an underactuated leg with a passive spring at the knee achieve a ballistic step? », *in: Proceedings of the International Symposium on Experimental Robotics*, Sidney, Australia, 26–28 march, 1999.
- [24] B. ESPIAU, « The BIP project », *in: IARP-E.U. Japan Symposium on Human-Friendly Robotics*, Tokyo, Japan, 8–9 nov., 1999.
- [25] L. FRANCE, A. GIRAULT, B. ESPIAU, « Generation of Walk Transient Trajectories for a Biped Robot », *in: Proceedings of the International Conference on Advanced Robotics, ICAR'99, Japan, octobre*, 1999.
- [26] L. FRANCE, A. GIRAULT, J. GASCUEL, B. ESPIAU, « Sensor Modeling for a Walking Robot Simulation », *in: Proceedings of the Eurographics Workshop on Computer Animation and Simulation, Milano, Italy, september 1999*, p. 189–198, 1999.
- [27] A. GIRAULT, G. BERRY, « Circuit generation and verification of ESTEREL programs », *in: Proceedings of the International Symposium on Signals, Circuits, and Systems (SCS'99)*, Iasi, Roumanie, juillet, 1999.
- [28] A. GIRAULT, S. YOVINE, « Stability analysis of a longitudinal control law for autonomous vehicles », *in: Proceedings of the IEEE Conference on Decision and Control (CDC'99)*, Phoenix, USA, décembre, 1999.
- [29] F. GÉNOT, B. ESPIAU, « On the Control of the Mass Center of Legged Robots Under Unilateral Constraints », *in: Proc. of the First International Symposium on Mobile, Climbing and Walking Robots CLAWAR'98*, 1999.
- [30] F. JIMÉNEZ-FRAUSTRO, E. RUTTEN, « Modélisation synchrone de standards de programmation de systèmes de contrôle », *in: Actes de la Journée d'études sur les Nouvelles Percées dans les Langages pour l'Automatique*, Amiens, 25 novembre, 1999. à paraître.
- [31] F. JIMÉNEZ-FRAUSTRO, E. RUTTEN, « A synchronous model of the PLC programming language ST », *in: Proceedings of the Work In Progress session, 1st Euromicro Conference on Real Time Systems, ERTS'99*, York, England, June 9–11, p. 21–24, 1999. <http://www.idt.mdh.se/ecrts99/wip.htm>.
- [32] B. LAMIROY, B. ESPIAU, N. ANDREFF, R. HORAUD, « Controlling Robots With Two Cameras: How to Do it Properly. », *in: International conference on Robotics and Automation (ICRA2000)*, 2000. Soumission; réponse attendue fin novembre.

- [33] A. LORIA, E. PANTELEY, A. TEEL, « A new persistency-of-excitation condition for ugas of NTLV systems: application to stabilization of nonholonomic systems », *in: Proceedings of the 5th. European Control Conference, Karlsruhe, Germany, septembre 1999.*
- [34] A. LORIA, E. PANTELEY, « A separation principle for a class of Euler-Lagrange systems », *in: Proceedings of the Workshop on Nonlinear Observers and Control, Geiranger, Norway, juin 1999.*
- [35] D. SIMON, R. PISSARD-GIBOLLET, K. KAPellos, B. ESPIAU, « Synchronous composition of discretized control actions: design, verification and implementation with ORCCAD », *in: Real-Time Computing Systems and Applications RTCSA'99, Hong-Kong, december 1999. à paraître.*
- [36] P.-B. WIEBER, « Control Issues in Biped Walking », *in: Colloque EUROMECH 397, Impact in Mechanical Systems, 1999.*

Rapports de recherche et publications internes

- [37] F. BACCELLI, B. GAUJAL, D. SIMON, « Analysis of Preemptive Real Time Systems using the (max,plus)algebra with applications in robotics », *Rapport de Recherche n° 3778, INRIA, octobre 1999.*

Divers

- [38] C. AZEVEDO, *Poursuite de trajectoires pour un double pendule sous-actionné*, Rapport de DEA LAG, INPG, Grenoble, juin 1999.
- [39] J.-J. PARMENTIER, *Contribution à la commande d'un robot bipède*, Rapport de stage de fin d'études, Ecole Polytechnique, juin 1999.
- [40] A. VILCHIS-GONZALEZ, *Spécification et implémentation d'une commande sûre d'un bras manipulateur*, Rapport de DEA IVR, ENSIMAG, Grenoble, juin 1999.