

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

No. attribué par la bibliothèque

/ / / / / / / / / / /

T H E S E

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : Automatique Productive

préparée au laboratoire : INRIA Rhône-Alpes
dans le cadre de l'École Doctorale : EEATS

présentée et soutenue publiquement

par

Christine AZEVEDO

le 16 septembre 2002

Titre :

**Contrôle de la locomotion artificielle :
de l'homme aux robots**

Directeur de thèse : Bernard ESPIAU

JURY

M. Mazen ALAMIR	Président
M. Wisama KHALIL	Rapporteur
M. Guy BESSONNET	Rapporteur
M. Bernard ESPIAU	Directeur de thèse
M. Philippe POIGNET	Co-encadrant
M. Friedrich PFEIFFER	Examineur

Remerciements

De façon générale je voudrai faire part de ma reconnaissance à l'INRIA Rhône-Alpes et au projet BIP de m'avoir accueillie pendant ces trois années de thèse, ainsi qu'au LIRMM et au projet RDC du département robotique de m'avoir reçue régulièrement à Montpellier. Je remercie plus particulièrement mon directeur de thèse Bernard Espiau pour ses conseils éclairés, sa patience et sa confiance dans mon travail ainsi que Philippe Poignet qui s'est rallié à ma cause en cours de route et m'a consacré du temps et de l'énergie.

J'adresse également mes remerciements à Wisama Khalil et Guy Bessonnet qui m'ont fait l'honneur d'être les rapporteurs de ce travail, ainsi qu'à Mazen Alamir et Friedrich Pfeiffer qui ont accepté de faire partie de mon jury. Leurs remarques, corrections et conseils ont été riches et constructifs.

Merci à Christine Assaïante de m'avoir fait le plaisir de relire une partie de ce mémoire. La perspective d'un post-doc dans son équipe m'a redonné énergie et courage pour clore cette thèse.

Une mention spéciale au service des Moyens Robotiques de l'INRIA dont les compétences et la bonne humeur m'ont été d'un grand secours.

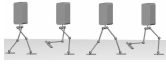
Je suis également très reconnaissante de l'amitié et des conseils de mes mentors : Roger, Gérard, Pascal, Soso, Niko.

Un clin d'oeil et un grand sourire à : Céline, Véro, Laurence, Jean-Matthieu, Olivier, Isa... Sans oublier Fabien qui aura été un stagiaire fort sympathique et attachant.

Une pensée émue à mes parents qui me vouent une confiance inconditionnelle et me soutiennent dans tous mes projets ... et bien sûr à Thomas sans qui rien ne serait.

Que sei eu do que serei, eu que não sei o que sou ? Ser o que penso ? Mas penso tanta coisa ! E há tantos que pensam ser a mesma coisa que não pode haver tantos ! Génio ? Neste momento Cem mil cérebros se concebem em sonho gênios como eu, E a história não marcará, quem sabe ?, nem um, Nem haverá senão estrume de tantas conquistas futuras. Não, não creio em mim. Em todos os manicômios há doidos malucos com tantas certezas ! Eu, que não tenho nenhuma certeza, sou mais certo ou menos certo ? Não, nem em mim... Em quantas mansardas e não-mansardas do mundo Não estão nesta hora gênios-para-si-mesmos sonhando ? Quantas aspirações altas e nobres e lúcidas - Sim, verdadeiramente altas e nobres e lúcidas -, E quem sabe se realizáveis, Nunca verão a luz do sol real nem acharão ouvidos de gente ?(...)

Fernando Pessoa - A Tabacaria

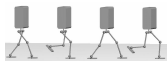


Avant-propos

Les robots mobiles à pattes (ou marcheurs) se distinguent des systèmes à roues ou à chenilles par la problématique scientifique qu'ils représentent et les applications qu'ils visent. Ils sont particulièrement adaptés aux environnements encombrés où le franchissement d'obstacles est plus aisé que le roulement : forêts, rochers, etc. Le déplacement est peu invasif car les pieds peuvent être positionnés de façon à préserver le milieu (végétaux, cultures...). Les robots marcheurs bipèdes de par leur nature s'adaptent parfaitement aux infrastructures conçues pour et par l'homme (couloirs, escaliers...).

Les applications et ambitions des robots bipèdes et autres humanoïdes construits actuellement sont souvent différentes. Les activités de service et assistance sont l'un des objectifs visés, car l'on imagine aisément que le gabarit anthropomorphe de ces machines puisse leur permettre d'aider l'homme dans ses tâches professionnelles ou privées (Bischoff, 2001). Du fait de leur aspect "sympathique", les humanoïdes ont évidemment des applications ludiques et publicitaires (Hirai et al., 1998). Enfin, les techniques mises au point pour les robots marcheurs à deux pattes permettent de progresser dans la réalisation des orthèses, prothèses et autres exosquelettes (Espiau et Génot, 2002) mais aussi dans la compréhension des mécanismes liés au contrôle locomoteur (VanDerKooij, 2000).

Les robots bipèdes posent des questions scientifiques difficiles : aux problèmes classiques des robots mobiles (perception et compréhension de l'environnement, décision, adaptation, autonomie) s'ajoutent d'autres difficultés. Un premier point délicat concerne la stabilité du système qui est sans cesse remise en cause au cours de la locomotion, mais qui doit néanmoins être garantie pour éviter toute situation de chute au robot. La ponctuation de la démarche par des contacts et impacts rend en effet l'équilibre précaire. La conception mécanique de ces structures est également complexe, la recherche de caractéristiques anthropomorphes nécessitant de solutionner les problèmes liés à l'encombrement, aux choix des capteurs, au câblage... Pour l'ensemble de ces raisons, la commande de telles machines est source de nouvelles problématiques.



La confrontation au problème de la locomotion conduit à réaliser combien notre connaissance de nous-même est limitée lorsque l'on tente d'insuffler nos capacités de déplacement dans un corps mécanique (Knoll, 2002). L'acquisition de la marche bipède nécessite de surmonter les contraintes d'équilibre dynamique qui accompagnent la locomotion. Il s'agit de parvenir à un ajustement des composantes d'équilibre et de rythmicité pour permettre une coordination harmonieuse entre posture et mouvement au moyen d'un contrôle anticipé (Assaiante, 2001).

La plupart des travaux concernant les robots bipèdes tendent à imiter les comportements humains. Nous nous sommes intéressés à certains des mécanismes intrinsèques qui régissent la locomotion anthropomorphe, c'est-à-dire à la recherche des causes et non pas des effets du contrôle de la marche.

Nous nous sommes ainsi attachés à respecter une cohérence dans la rédaction de ce mémoire, le fil conducteur qui guide les pages qui vont suivre étant la relation humain-machine et leurs apports mutuels.

Ce mémoire se décompose en quatre parties :

- ✓ La première donne l'orientation générale de ce travail et l'esprit qui l'anime : trouver l'inspiration dans l'observation de l'homme pour appréhender la marche bipède en robotique.
- ✓ La deuxième présente les machines bipèdes sur lesquelles nous avons validé notre travail, en simulation ou expérimentalement, et nous décrivons également les options retenues pour la modélisation dynamique des robots.
- ✓ La troisième propose notre approche expérimentale de la locomotion, depuis la génération de démarches jusqu'à la réalisation du contrôleur permettant de poursuivre ces trajectoires. Cette méthodologie est adaptée à l'analyse des effets des paramètres de la marche. Elle convient parfaitement aux mouvements posturaux intervenant dans des situations ou environnements délicats.
- ✓ La quatrième enfin, développe le cadre d'une stratégie de commande originale. Elle puise son inspiration dans les solutions choisies par la nature et qui se traduit en termes de commande prédictive sans suivi de trajectoire de référence. Cette approche favorise l'adaptabilité et la réactivité des déplacements.

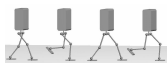
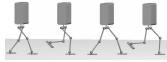


Table des matières

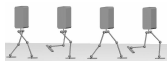
I	La marche bipède : de l'homme aux robots	11
	Problématique	12
1	La marche bipède humaine	13
1.1	Introduction	13
1.2	Structure de la marche	13
1.3	Contrôle de la marche	18
1.4	Conclusion	23
2	Principes de la marche bipède robotisée	24
2.1	Introduction	24
2.2	Structure de la marche	24
2.3	Contrôle de la marche	30
2.4	Synthèse	32
3	Analogies entre homme et robot	33
3.1	Introduction	33
3.2	De l'homme vers le robot : conception et contrôle de machines bipèdes	33
3.3	Du robot vers l'homme : la marche artificielle	36
3.4	Conclusion	36
	Conclusion	38
II	Des robots à leurs modèles	39
	Problématique	40
1	Plates-formes expérimentales	41
1.1	Introduction	41
1.2	Robot BIP dans sa configuration 3D	42
1.3	Robot BIP dans sa configuration 2D	45
1.4	Le robot RABBIT	46
1.5	Conclusion	47

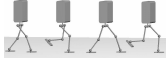


2	Modélisation dynamique	48
2.1	Introduction	48
2.2	Dynamique des robots	48
2.3	Application aux prototypes	54
2.4	Conclusion	57
	Conclusion	58
III	Approche expérimentale de la locomotion	59
	Problématique	60
1	Génération d'allures de marche	62
1.1	Introduction	62
1.2	Robot BIP 2D	63
1.3	Robot BIP 3D	67
1.4	Conclusion	72
2	Contrôleur pour la poursuite d'allures de marche	73
2.1	Introduction	73
2.2	Loi de commande	73
2.3	Réglage des paramètres	74
2.4	Implantation du contrôleur	75
2.5	Application au suivi de mouvements 3D	79
2.6	Application au suivi de mouvements 2D	80
2.7	Conclusion	80
3	Expérimentation de marches	81
3.1	Introduction	81
3.2	Présentation des marches	81
3.3	Analyse de la marche de référence	82
3.4	Comparaison des marches	85
3.5	Conclusion	87
	Conclusion	88
IV	Optimiser et prédire : vers l'autonomie	89
	Problématique	90



1	La commande prédictive	92
1.1	Introduction	92
1.2	Principes	93
1.3	Commande prédictive non linéaire	97
1.4	Conclusion	104
2	Stratégie de commande	105
2.1	Introduction	105
2.2	Stratégie	105
2.3	De la stabilité à la faisabilité	110
2.4	Étude systématique sur un exemple simple	112
2.5	Locomotion bipède	121
2.6	Conclusion	132
3	Vers la mise en œuvre de la commande	133
3.1	Introduction	133
3.2	Choix du problème d'optimisation	133
3.3	Auto-apprentissage	133
3.4	Limitation des itérations	134
3.5	Calcul sur des périodes supérieures à T_e	134
3.6	Simplification du modèle	134
3.7	Programmation en "dur"	134
	Conclusion	135
	Pour finir...	137
	Publications de l'auteur	139
	Bibliographie	140
	Annexes	151





Première partie

La marche bipède : de l'homme aux robots

Sommaire

- ★ **Problématique**

mettre en exergue les analogies entre locomotions humaine et artificielle

- ★ **La marche bipède humaine**

de la structure au contrôle de la locomotion

- ★ **Principes de la marche bipède robotisée**

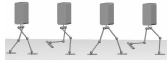
tour d'horizon de la technique bipède

- ★ **Analogies entre homme et robots**

quelles interactions entre robotique et sciences du vivant ?

- ★ **Conclusion**

s'inspirer des comportements humains pour contrôler les machines bipèdes

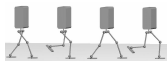


Problématique

Mettre en exergue les analogies entre locomotions humaine et artificielle.

L'une des ambitions de la robotique est de comprendre de quelles capacités peuvent ou doivent être dotées les machines en interaction avec leur environnement et de quelle manière ces machines peuvent s'adapter pour améliorer leurs performances et devenir plus autonomes. Depuis longtemps déjà, informatique et robotique se sont inspirées des sciences du vivant (Hollerbach, 1982 ; Loeb, 1983 ; Slotine, 2002). Pour l'étude de la locomotion artificielle, l'homme est une référence indiscutable, que l'on souhaite ou non l'imiter. La compréhension des solutions choisies par la nature pour résoudre le problème de la motricité bipède est une source d'informations importantes pour la robotique. Mais il est loin d'être évident que la transposition aveugle d'un domaine à l'autre soit efficace. De plus, elle peut s'avérer techniquement irréalisable. Cependant, des interactions sont possibles et souhaitables entre sciences de l'humain et sciences techniques.

Le premier chapitre rappelle les principes de base de la marche anthropomorphe. Le chapitre suivant est un état de l'art des solutions retenues classiquement en robotique bipède. La structure de la locomotion et les techniques de contrôle des déplacements sont abordées à la fois dans le cas de la marche humaine et dans celui de la marche robotisée. Notre objectif est de mettre en évidence les analogies qui existent entre les deux domaines ainsi que leurs limites. Cette analyse nous sera utile pour définir des allures de marche du robot BIP et nous verrons également comment ces observations nous ont conduit à élaborer une nouvelle approche pour la commande de robots bipèdes.



Chapitre 1

La marche bipède humaine

1.1 Introduction

Deux approches permettent d'appréhender la locomotion anthropomorphe : une approche *descriptive* de la marche basée sur l'observation de variables mesurables qui permet de mettre en évidence certains invariants (biomécanique) et une approche *explicative* qui s'intéresse aux causes du déplacement (neurosciences, physiologie). La première met en avant les aspects structurels de la marche, la seconde étudie les mécanismes de contrôle du mouvement. Nous proposons dans ce chapitre de récapituler les propriétés qui nous semblent importantes pour aborder la locomotion artificielle.

1.2 Structure de la marche

De nombreux travaux s'intéressent à la description de la marche humaine et de ses invariants. La présentation qui suit s'est inspirée des références suivantes :

(Gage, 1990 ; Espiau et Sardain, 2000 ; Zatsiorsky et al., 1994 ; Whittle, 1996 ; Cappozzo, 1984 ; Rose et Gamble, 1994 ; Allard et Blanchi, 1996 ; Donatelli, 1990 ; Winter et al., 1990).

Nous proposons tout d'abord un rappel des notions de base couramment employées dans le contexte de la locomotion bipède :

- on appelle *pas* le mouvement qui se produit entre le poser du talon d'une jambe et celui de la jambe opposée ;
- la *foulée* comprend deux pas et correspond au mouvement qui sépare deux posers successifs du talon d'une même jambe. L'enchaînement de foulées constitue le mouvement de *marche* (fig.1.1 page suivante) ;
- le mouvement humain est classiquement étudié par rapport à trois plans de référence : le *plan sagittal* (le plan dans lequel se manifeste principalement le mouvement), le *plan frontal* ou coronal, le *plan transversal* ou horizontal (fig.1.2), à partir desquels sont définies trois composantes de rotation : le tangage, le roulis et le lacet.

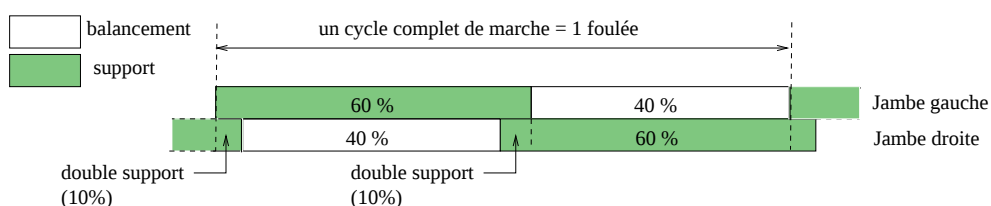
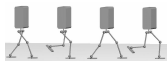


FIG. 1.1 – Cycle de marche d'un bipède

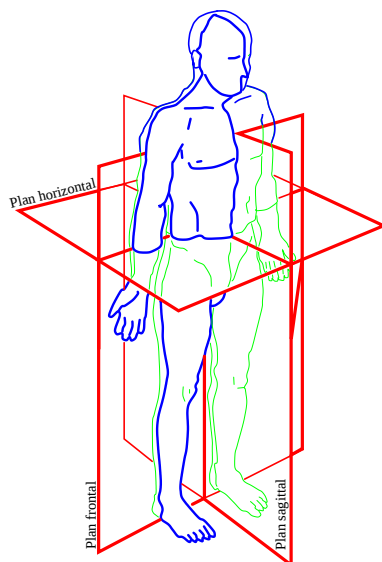
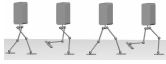


FIG. 1.2 – Les 3 plans de référence pour l'analyse du mouvement (Vaughan et al., 1992)

La marche normale est périodique et symétrique. Au repos, le centre de gravité du corps (cgc) se trouve légèrement au dessus de la mi-hauteur (55 %) de la taille totale de l'individu. Les conditions moyennes de marche pour un adulte mâle sain, en ligne droite sur sol plan horizontal en régime stationnaire sont : vitesse de $5,4 \text{ km.h}^{-1}$; cadence de 115 pas.mn^{-1} ; longueur de foulées de 155 cm. Ces valeurs évoluent en fonction de la taille et du poids de l'individu.

1.2.1 Le cycle de marche

La marche est une succession de mouvements répétitifs. La notion de *cycle de marche* se rapporte aux mouvements d'une jambe. Il correspond aux mouvements qui surviennent pendant une *foulée* (fig.1.1). Le cycle débute lorsqu'un pied entre en contact avec le sol et se termine lorsque ce même pied entre à nouveau en contact avec le sol. Le cycle comprend deux phases principales : l'*appui* (également appelé *support*) et le *balancement* (également appelé *oscillation*) (fig.1.3 page 16).



La phase d'appui débute avec le contact initial du talon sur le sol. Elle prend fin lorsque les orteils décollent, instant où commence la phase de balancement. L'appui représente environ 60 % du cycle et le balancement environ 40 % du cycle. On appelle *phase de double appui* la période au cours de laquelle les deux pieds sont simultanément au sol, elle survient deux fois au cours du cycle et représente à chaque fois 10 % du cycle. Plus la vitesse de déplacement augmente, plus le double appui est court par rapport à la phase d'oscillation.

- La phase d'appui d'une jambe donnée peut être décomposée en cinq sous-périodes (fig.1.3 page 16) :

0 % - **contact initial** : la phase d'appui débute avec le poser du talon du pied de la jambe considérée qui devient alors *jambe de support*. La jambe opposée étant la *jambe libre*.

0 à 10 % - **reprise en charge** : le premier double appui est une sous-phase de reprise en charge. À l'issue de cette période, le genou de support présente un premier maximum de flexion dû à l'absorption de l'impact. Le centre de gravité du corps (cgc) atteint son point le plus bas.

10 à 30 % - **mi-appui** : le simple appui commence, la jambe libre entre dans sa phase de balancement. Le pied passe par une période de pose à plat, la jambe atteint son extension maximale. La progression vers l'avant du cgc décélère, il atteint son maximum de hauteur tandis que sa projection sur le sol atteint la limite des orteils du pied d'appui.

30 à 50 % - **appui terminal** : la projection du cgc se trouve en dehors du pied d'appui, il accélère et retombe vers l'avant du côté non supporté. L'équilibre est alors assuré en ré-injectant une quantité d'énergie équivalente à celle perdue précédemment dans le cycle de marche.

50 à 60 % - **pré-balancement** : cette sous-phase est le second double appui durant lequel le poids est transféré sur le pied de la jambe qui était libre. Cette période s'achève par le décollage des orteils.

- La phase de balancement d'une jambe se décompose en trois sous-périodes :

60 à 70 % - **balancement initial** : la jambe peut être accélérée ou décélérée par l'intervention des muscles qui lient hanche et genou et qui agissent pour transférer l'énergie entre cuisse et tibia.

70 à 80 % - **mi-balancement** : le milieu de balancement est une période de transition au cours de laquelle les muscles du genou sont au repos.

80 à 100 % - **balancement terminal** : la jambe peut être accélérée ou décélérée, le balancement prend fin lorsque le pied libre heurte le sol.

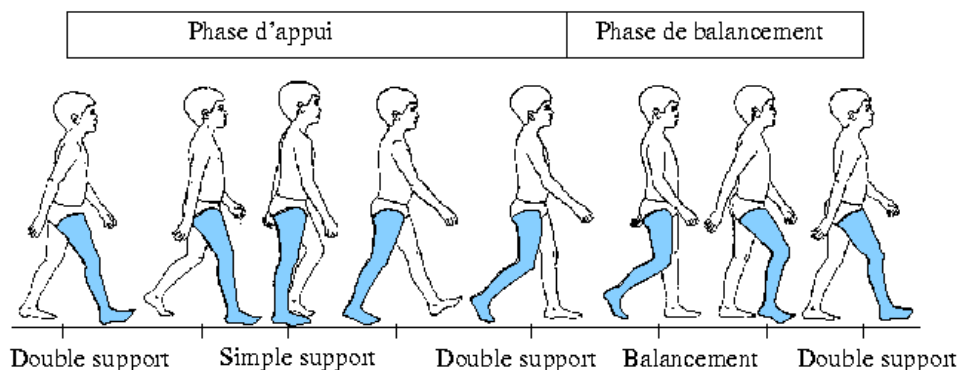
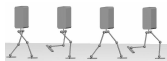


FIG. 1.3 – Phases d'un cycle de marche (Vaughan et al., 1992)

La structure de la marche n'est pas figée, elle peut être différente d'un individu à un autre, mais aussi chez un individu donné. Les *transitoires* correspondent à des facteurs de modification des patrons standards de la marche :

1. à court terme : démarrage du pas, arrêt du pas, virages, changements de la pente du sol, escaliers, supports étroits (poutres), port de charges...
2. à long terme : apprentissage, fatigue...

1.2.2 Équilibre

Le contrôle de l'équilibre est une composante commune à la posture et à la locomotion (Assaiante, 2001).

L'équilibre postural (statique) est assuré dès lors que la projection du centre de gravité (cgc) se trouve à l'intérieur de la *base de support*. La base de support (ou *polygone de sustentation*) est délimitée par les points de contact avec le sol, elle correspond à la surface de l'enveloppe qui relie les points d'appui (fig.1.4 page 17). Le centre de pression (cdp) est le point où le moment des efforts de pression est nul et celui où le moment des efforts de contact est normal au sol ¹. La projection du cgc et le cdp sont proches lorsque la projection du cgc se trouve à l'intérieur de la base de support.

L'équilibre locomoteur (dynamique) est plus complexe car il doit concilier la propulsion du corps vers l'avant, qui constitue un déséquilibre manifeste, avec le maintien de la stabilité latérale du corps. Lors de la marche, la projection du cgc n'est pas constamment située sur la base de support. Un individu qui marche normalement est

¹Le centre de pression est également le point du sol sur lequel est appliquée la force de réaction au sol. On peut montrer que son expression fait intervenir des termes dynamiques.

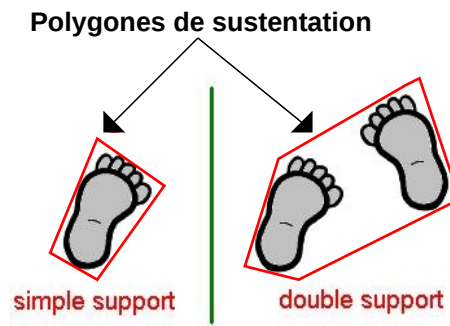
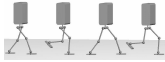


FIG. 1.4 – Le polygone de sustentation bipède

continuellement en état de déséquilibre avant. La seule façon d’empêcher la chute est de positionner le pied libre en avant du cgc (Winter, 1990). La progression résulte de la chute en avant du corps depuis sa hauteur maximale en milieu de support jusqu’à son point le plus bas pendant le double appui au cours duquel l’énergie potentielle est transformée en énergie cinétique. L’énergie cinétique nécessaire pour ramener le cgc à son point le plus haut au milieu de support suivant est fournie par l’inertie de la jambe de balancement.

1.2.3 Bilan énergétique

Les êtres humains ont une façon de se déplacer qui privilégie une vitesse de progression. Pour une vitesse donnée, ils choisissent une démarche particulière et une combinaison de paramètres de marche spécifiques. Ces variables “préférées” sont sélectionnées de manière à minimiser le coût énergétique (Nubar et Contini, 1961 ; Donelan et al., 2001).

Dans ce coût énergétique, on distingue deux types d’énergies : l’énergie mécanique et l’énergie métabolique. L’énergie métabolique correspond à l’énergie dépensée par un individu pour activer les processus métaboliques (ex : contraction musculaire) et pour activer les réactions chimiques cellulaires (ex : respiration). Cette énergie est obtenue par la nourriture ingérée et par les réserves en glucides et lipides. Elle est transformée en énergies thermique, chimique et mécanique.

Au repos, l’énergie est convertie en chaleur uniquement. L’énergie métabolique moyenne consommée est alors de $17,2 \text{ cal.kg}^{-1}.\text{mn}^{-1}$. En station debout immobile, l’énergie métabolique consommée en moyenne est de $24,5 \text{ cal.kg}^{-1}.\text{mn}^{-1}$. Lors d’une activité comme la marche, l’énergie est convertie en chaleur à 76 % et en énergie mécanique à 24 %. En marche normale stationnaire l’énergie consommée

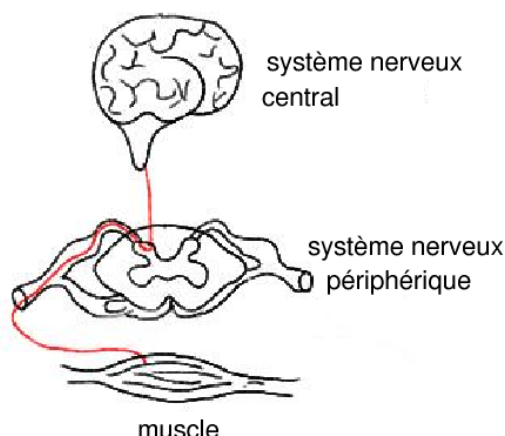
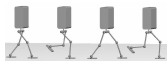


FIG. 1.5 – De l'idée du mouvement à la contraction musculaire

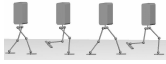
est de $72 \text{ cal.kg}^{-1}.\text{mn}^{-1}$ pour une vitesse de confort de 90 m.mn^{-1} ($5,4 \text{ km.h}^{-1}$) soit une consommation de $0,8 \text{ cal.kg}^{-1}.\text{m}^{-1}$. La *vitesse de confort* est celle qui minimise le taux d'énergie métabolique consommée par unité de distance, c'est la vitesse spontanément adoptée par le marcheur en régime permanent.

1.3 Contrôle de la marche

Parallèlement aux travaux qui décrivent les données spatio-temporelles des différents paramètres du pas, des recherches portent sur les processus volontaires impliqués dans la locomotion. La dynamique de marche peut être abordée par une approche descendante (Vaughan et al., 1992 ; Winter et al., 1990), commençant par une impulsion nerveuse dans le système nerveux central (SNC) et se terminant par la génération de forces de réaction au sol (fig.1.5).

La séquence des évènements nécessaires pour marcher se résume comme suit :

1. idée du déplacement au niveau du SNC ;
2. transmission des signaux correspondant à la marche au système nerveux périphérique (moelle épinière) ;
3. contraction des muscles correspondant à la marche souhaitée ;
4. génération de forces et de couples dans les articulations ;
5. régulation des forces et couples articulaires par les segments du squelette ;
6. déplacements des segments créant le mouvement de marche ;
7. génération de forces de réaction.



1.3.1 Perception et action

Les recherches actuelles tendent à démontrer qu'il existe dans la moelle épinière des centres nerveux capables d'induire une activité rythmique (générateurs de la marche) (Berthoz, 1997 ; Clarac, 1997 ; Giménez y Ribotta et al., 2000 ; Dyu-sens et al., 2002). Le rythme est entretenu par des boucles internes qui contrôlent intensité et cadence et des récepteurs sensoriels qui modifient et adaptent l'activité au milieu extérieur. Le corps humain est muni de nombreux capteurs dont les rôles sont différents et complémentaires (Berthoz, 1997 ; VanDerKooij, 2000) :

- **capteurs proprioceptifs :**

- *récepteurs musculaires* : les muscles contiennent, outre les fibres musculaires, des fibres appelées *fuseaux neuromusculaires* qui détectent l'allongement du muscle,
- *récepteurs articulaires* : ces capteurs mesurent les positions articulaires, ce sont des équivalents biologiques des capteurs de positions tels que les potentiomètres et les codeurs optiques,
- *récepteurs de Golgi* : ces capteurs sont placés sur les tendons en série avec le muscle, ils mesurent l'effort exercé par le muscle sur l'articulation.

Les capteurs proprioceptifs mesurent les mouvements relatifs des segments corporels.

- **capteurs extéroceptifs :**

- *récepteurs cutanés* : ces capteurs sont positionnés sur la peau avec des concentrations variables selon la zone du corps et mesurent les pressions et les frottements du contact des membres entre eux ou avec le monde extérieur,
- *récepteurs visuels* : ces capteurs mesurent le glissement de la scène observée sur la rétine, la position des objets dans l'espace, les formes, les couleurs...

- **autres capteurs** : les *récepteurs vestibulaires* sont placés dans l'oreille interne.

Ils mesurent le mouvement de la tête dans l'espace, ce sont des équivalents biologiques des accéléromètres. Cette information peut aussi être estimée en recoupant les informations visuelles et proprioceptives dans le cas où la base de support et la scène visuelle sont fixes. Ces capteurs sont de véritables centrales inertielles, ils perçoivent les forces d'inertie et informent des mouvements absolus de la tête et du corps dans l'espace. La sensibilité de ces capteurs leur permet de détecter des variations d'accélération de quelques dixièmes de degré par seconde au carré. Leur sensibilité dépend de la fréquence du mouvement, elle est faible à basse fréquence. Dans les travaux de T. Viéville, les informations inertielles et visuelles sont associées pour la navigation autonome d'un robot mobile (Viéville et al., 1993).

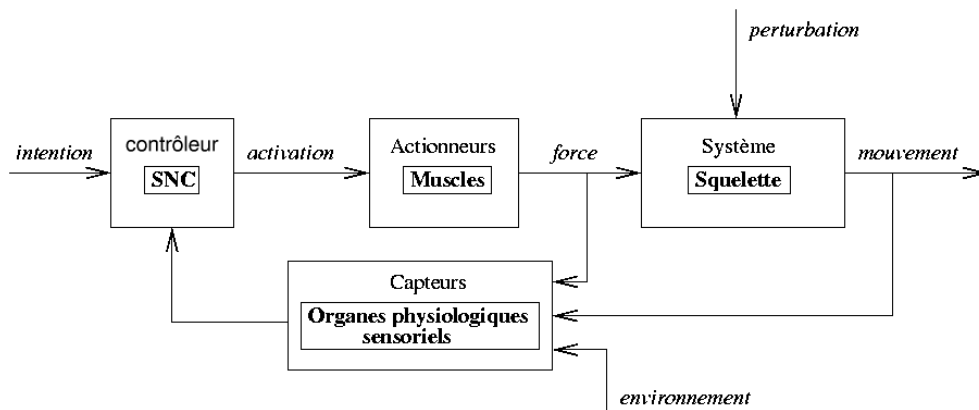
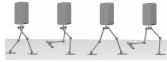


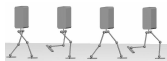
FIG. 1.6 – Schéma bloc du contrôle de l'équilibre chez l'homme (VanDerKooij, 2000)

La perception n'est pas uniquement une interprétation des informations sensorielles, elle est contrainte par l'action visée : ainsi les capteurs biologiques peuvent s'adapter et modifier la fréquence et l'amplitude de leurs informations en fonction de la tâche à réaliser (Berthoz, 1997).

1.3.2 Équilibre postural

Dans toutes les activités organisées dans l'espace, le contrôle de l'*équilibre postural* est un pré-requis d'une motricité efficace. L'homme est pourvu de nombreux degrés de liberté, cette redondance assure un grand choix de postures possibles pour une même tâche. La posture assure une fonction anti-gravitaire qui permet le maintien de l'équilibre à l'initialisation et au cours du mouvement et sert de cadre de référence pour organiser le mouvement (Assaiante, 2000). Le système de contrôle de l'équilibre peut être décomposé en sous-systèmes. Le fonctionnement de ces sous-systèmes peut être décrit de façon descendante (fig.1.6) :

1. le système nerveux central (SNC) initie et contrôle les mouvements dans le but d'effectuer une tâche particulière (marcher, attraper un objet...). On peut voir le SNC comme un contrôleur hiérarchique complexe (VanDerKooij, 2000) ;
2. les signaux issus du cerveau activent les muscles qui actionnent le squelette ;
3. des signaux issus des organes sensoriels dispensent des informations concernant la posture, les forces internes et externes et l'environnement. Ces informations sont utilisées par le SNC pour maîtriser la commande.



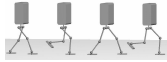
Les informations issues des capteurs physiologiques sont utilisées à deux niveaux :

- bas niveau : les signaux issus des fuseaux musculaires, des tendons et des capteurs de la peau sont envoyés dans le muscle (réflexes spinaux). Ces réflexes contribuent à l'interaction dynamique avec l'environnement. En effet, une propriété du muscle est de se contracter lorsqu'il est étiré. Ainsi lorsque le corps s'incline sous l'effet de la gravité, les capteurs le détectent et induisent une contraction musculaire qui redresse le corps.
- haut niveau : les autres informations sensorielles fournissent à l'individu des connaissances concernant les positions et orientations relatives des segments du corps les uns par rapport aux autres, la position du corps dans son environnement, les forces exercées sur cet environnement, la positions d'obstacles... Ces informations permettent de planifier et coordonner les actions motrices.

Le système vestibulaire présenté précédemment participe à la stabilisation posturale. Sa puissance anticipatrice est liée au fait qu'il mesure une accélération. L'accélération est maximale au début de la chute ou du déséquilibre, les réflexes de redressement de la tête et du corps sont donc très rapides car déclenchés dès le début de la perturbation. Cette vivacité de rattrapage est encore accentuée par le fait que le cerveau dispose d'un répertoire de réactions posturales toutes prêtes (Berthoz, 1997). Le contrôle des articulations implique aussi la maîtrise des paramètres temporels qui permet, soit de façon anticipée, soit de façon rétroactive, une coordination efficace entre posture et mouvement. Dans le cas d'une perturbation extérieure imprévue, seul le contrôle rétroactif permet de maintenir ou rétablir l'équilibre. Lors d'une perturbation prévisible, il est possible d'assurer la fonction d'équilibre grâce à un contrôle anticipé.

1.3.3 Anticipation et prédiction

La réalisation efficace d'une action implique la *prédiction* de ses conséquences et donc une *anticipation* (ex : anticipation de la trajectoire d'une balle pour l'attraper) (Assaiante, 2000). Le cerveau sert à prédire le futur et à anticiper les conséquences de sa propre action et de celle des autres pour s'adapter afin de "gagner du temps" (Berthoz, 1997). Des mécanismes biologiques dotent le cerveau de modèles internes du corps et de son environnement qui lui permettent de prédire le déroulement des événements induits par son acte. Le pouvoir d'anticipation est lié à la propriété des capteurs de mesurer les variations rapides (=dérivées). Lorsqu'il engage une action, le cerveau fait des hypothèses sur l'état que doivent prendre certains capteurs au cours de son déroulement. Le contrôle moteur compare prédiction et réalisation pour engager des corrections. Le contrôle du mouvement se fait de façon discrète. Le



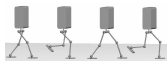
temps élémentaire de traitement des données est d'environ 25 ms, celui nécessaire pour une action élémentaire de contrôle de mouvement étant d'environ 100 ms. Les muscles se contractent lentement : ils atteignent leur maximum de force 80 ms après avoir reçu la commande nerveuse. Pour le contrôle de l'équilibre, le système nerveux reçoit des informations avec un certain délai. Il est donc impossible de connaître l'orientation exacte du corps par rapport à son environnement. Le système nerveux ne dispose donc que d'une estimation. Le cerveau est capable de stocker des structures de mouvements. Ces structures correspondent à des relations mémorisées (liens topologiques) entre plusieurs composantes sensorielles ou motrices de l'action. La prédiction des conséquences du mouvement est donc également liée à la mémoire des conséquences passées.

1.3.4 Stratégies

L'automatisme locomoteur est inconscient, mais les interventions des centres supérieurs peuvent se faire à tout moment lorsqu'il y a un problème de choix ou de stratégie locomotrice (Clarac, 1997). Le système nerveux n'est pas capable de contrôler tous les degrés de liberté qui caractérisent l'organisation anatomique et dynamique du squelette. Le cerveau dispose de modèles internes de la réalité physique qui permettent de simplifier le système à contrôler et de simuler le mouvement. Il dispose également d'un répertoire de mouvements naturels qui impliquent des groupes de muscles et de membres travaillant ensemble : c'est ce que A. Berthoz a appelé *synergies* (Berthoz, 1997). Les réseaux neuronaux manipulent ces mouvements simples pour constituer des séquences complexes orientées vers un objectif : ce sont des *stratégies*. Cette organisation simplifie le contrôle en évitant d'avoir à prendre en compte tous les muscles séparément.

1.3.5 Apprentissage

L'acquisition d'une habileté sensori-motrice au cours de l'enfance suit une progression similaire à celle de l'apprentissage d'une nouvelle habileté chez l'adulte (Newell, 1995). Dans un premier temps, les mouvements sont dominés par une dynamique non contrôlée qui conduit à un mouvement inefficace et à une faible stabilisation posturale. Puis, les individus apprennent à supprimer les effets néfastes de la dynamique passive par un contrôle musculaire actif. Ils apprennent à utiliser la dynamique passive inhérente au mouvement pour réduire la quantité de contrôle musculaire actif et d'énergie métabolique nécessaire pour réaliser l'action.



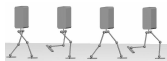
1.4 Conclusion

Si la marche humaine possède des invariants structurels (cyclicité, symétrie, phases...), les allures peuvent néanmoins différer d'un individu à l'autre, ainsi que chez une même personne en fonction de la finalité de l'action. Les déplacements correspondent à des mouvements qui permettent de minimiser un critère. Ce critère est lié aux objectifs du déplacement (performance ou endurance par exemple), il n'est donc pas nécessairement énergétique. Il faut également garder à l'esprit que la marche n'est symétrique et périodique que quelques pas après son initiation et quelques pas avant son freinage, c'est-à-dire en régime stationnaire. La marche stationnaire de confort n'est qu'un des aspects de la locomotion, elle est ponctuée de transitoires (arrêts, virage, escaliers...) dont les allures sont très diverses. La fatigue et l'apprentissage sont aussi des facteurs non négligeables de modification de la locomotion.

Le contrôle de la marche est lié à la réception et au traitement des informations sensorielles. Les actes induits par les données des capteurs physiologiques peuvent être réflexes ou contrôlés. Les capacités d'anticipation et de prédiction permettent de planifier les mouvements en utilisant des modèles simplifiés du corps et de son environnement. Ces capacités sont indispensables étant donné les délais de transmission des informations depuis les capteurs jusqu'au cerveau. La mémoire du système de contrôle permet de stocker des informations et de disposer de séquences de mouvements correspondant à des liens topologiques entre les composantes sensorielles et motrices de l'action. Ce répertoire est organisé en associant les séquences à des objectifs, correspondant ainsi à des stratégies.

Étant donnée la complexité du corps humain, les difficultés de modélisation et de simulation sont très importantes. Par ailleurs, certaines mesures peuvent être délicates, voire impossibles, car elles nécessiteraient, par exemple, une chirurgie invasive pour placer des capteurs. D'où l'un des intérêts de construire des plates-formes expérimentales bipèdes pour effectuer ces campagnes de mesures et valider des hypothèses. Par ailleurs, la modélisation d'un robot est moins problématique étant donné que les éléments de sa construction sont parfaitement connus.

Nous avons mis en évidence certains aspects très importants de la structure et du contrôle de la marche humaine. Ces éléments doivent également être pris en compte dans le cas des machines anthropomorphes. Nous reviendrons sur ce point dans le chapitre 3. Auparavant, le chapitre suivant propose un récapitulatif des principes de base de la robotique dans le domaine de la locomotion bipède.



Chapitre 2

Principes de la marche bipède robotisée

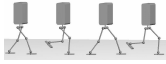
2.1 Introduction

De nos jours, les prototypes de robots bipèdes se multiplient. Leur mécanique devient de plus en plus complexe et aboutie (Gienger et Pfeiffer, 2001 ; Yamaguchi et al., 1999) (fig.2.1 page 25). Les industriels japonais s'orientent vers des machines humanoïdes aux rôles sociaux (divertissement, service). Les instituts universitaires sont eux davantage concernés par le développement de techniques avancées pour le contrôle des mouvements de leurs machines en interaction avec l'environnement. D'où l'intérêt grandissant des chercheurs et industriels pour l'analyse et la synthèse de la locomotion des robots bipèdes.

Ce chapitre est une présentation générale des principes classiquement retenus en robotique bipède. L'objectif est de pouvoir mettre en parallèle les choix faits par les scientifiques pour résoudre le problème de la locomotion anthropomorphe et ceux élaborés par la nature, présentés dans le chapitre précédent. Les détails concernant la modélisation et la commande seront développés plus loin dans ce mémoire. Nous présentons tout d'abord les propriétés liées à la structure des robots et de leur déplacement. Nous ferons ensuite un état de l'art de la commande des systèmes à deux pattes, ou comment obtenir un comportement désiré pour une machine soumise aux propriétés décrites.

2.2 Structure de la marche

Pour comprendre la marche robotique il est nécessaire de comprendre les phénomènes physiques mis en jeu par cette activité. Les robots bipèdes possèdent en effet, des propriétés et des caractéristiques spécifiques qu'il est important de garder à l'esprit.



WABIAN



Humanoid Robotics Inst.
Waseda University

M2



Leg Laboratory
Massachusetts Inst. of Technology

PINO



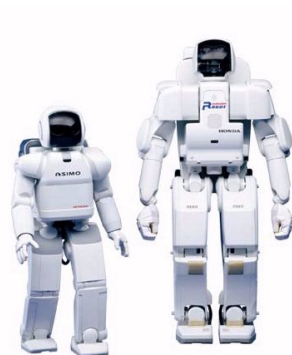
Hiroaki Kitano
Kitano Symbiotic Systems

SDR-4X



SONY

ASIMO et P3



Honda Motor Co

Johnnie



Institute for Applied Mechanics
Technical University of Munich

FIG. 2.1 – Quelques robots bipèdes parmi les plus aboutis.

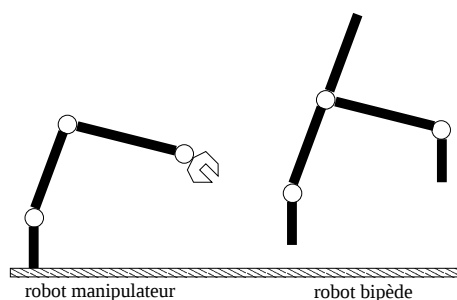
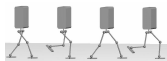


FIG. 2.2 – Comparaison entre robot manipulateur et robot bipède

2.2.1 Description des robots bipèdes marcheurs

Une machine bipède est composée de plusieurs segments interconnectés par des articulations généralement actives. On peut la voir comme un robot manipulateur arborescent dont la base serait libre de se déplacer (fig.2.2). La complexité de ces systèmes dépend du nombre de degrés de liberté dont ils sont pourvus. Les prototypes construits peuvent avoir une structure très simple (Garcia et al., 1997 ; McGeer, 1990) ou très complexe (Vukobratović et al., 1990 ; Hirai et al., 1998 ; Yamasaki et al., 2001 ; Yamaguchi et al., 1999 ; Gienger et Pfeiffer, 2001 ; Guihard et Gorce, 2001). Dans les deux cas, les robots sont capables de se déplacer en marchant. Les jambes du robot sont libres de décoller de la surface d'appui. La fréquence de ces décollements conduit à différents types de déplacements : marche, course, saut... Dans le cas de la marche, une jambe au moins est toujours en contact avec le sol. Comme nous l'avons vu page 16, le cycle de marche se décompose en phases de simple et de double appui. Le simple appui correspond à une configuration ouverte de la chaîne cinématique constituée par le squelette du robot. Le double appui correspond au contraire à une configuration fermée de la chaîne cinématique. Le nombre de degrés de liberté du robot est donc variable, il dépend de la phase d'appui ; le nombre d'articulations actives étant quant à lui constant, le système peut être sous-actionné ou sur-actionné.

2.2.2 Dynamique hybride

Un robot bipède peut être vu comme un système pendulaire qui interagit de façon intermittente avec son environnement : le sol. On peut donc représenter la progression du système bipède comme la séquence de phases continues et discontinues où les impacts avec le sol sont des événements qui ponctuent le déplacement (Hurmuzlu et al., 2001).

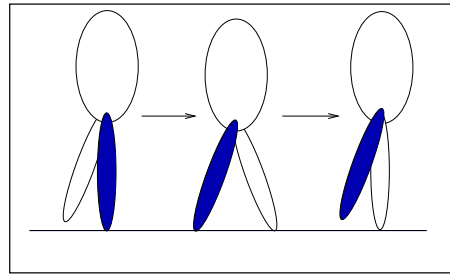
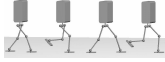


FIG. 2.3 – Exemple de marche bipède

Dans le cas où le transfert d'une phase d'appui à l'autre est supposé instantané (fig.2.3), lorsque la jambe de balancement entre en contact avec le sol, deux événements simultanés sont déclenchés :

1. l'impact, qui représente un changement instantané des vitesses généralisées,
2. la permutation des rôles des jambes qui correspond au transfert du point de pivot au point de contact.

La jambe qui servait de jambe d'appui devient la jambe libre et la jambe qui servait de jambe de balancement devient jambe de support. Dans le cas où une phase de double appui existe entre les deux simples appuis, un seul événement est déclenché par l'entrée en contact du pied libre sur le sol : l'impact.

2.2.3 Interaction avec le sol

La dynamique des robots bipèdes est caractérisée par l'existence de contraintes variables résultant du contact avec le sol. Une façon de modéliser ce phénomène est de considérer le robot comme une chaîne arborescente de corps rigides qui interagissent avec le sol au travers de liaisons unilatérales et de frottements. Des contraintes traduisent la non-pénétrabilité du pied dans le sol ; elles sont unilatérales, ce qui implique que les forces de contact normales sont semi-positives (absence de force attractive) (Hurmuzlu et al., 2001 ; Goswami, 1999). La condition de non glissement des points de contact sur le sol porte sur les forces de contact tangentielles. Cette propriété est assurée dès lors que la force de frottement est suffisante pour s'opposer au glissement des deux surfaces en contact.

2.2.4 Faisabilité des mouvements

La position q d'un robot bipède se compose d'une part des positions articulaires du robot (posture), et d'autre part de la position et de l'orientation du système dans l'espace. La dimension de q correspond aux degrés de liberté du robot. Un robot

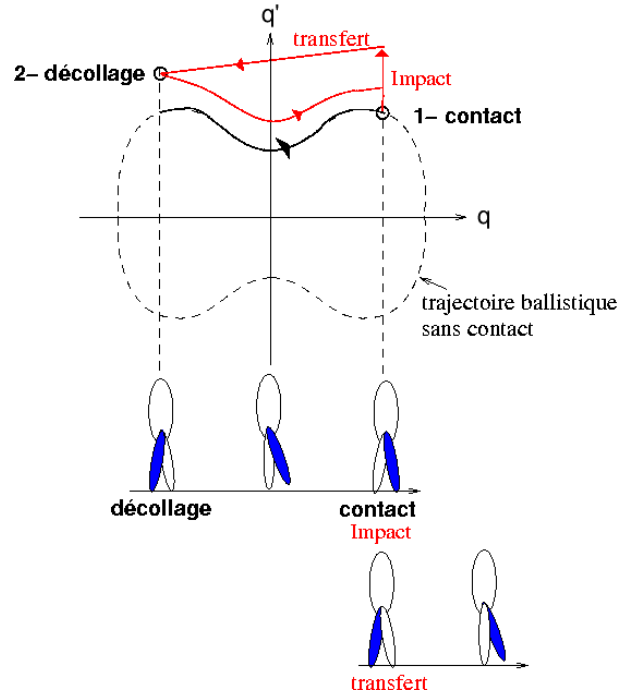
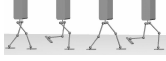


FIG. 2.4 – Plan de phase - q et q' sont respectivement les position et vitesse articulaires de la jambe considérée

comporte des actionneurs et des systèmes de transmission permettant de produire des couples articulaires. Les moteurs agissent sur une partie seulement de la position du robot, c'est-à-dire sur la posture. On peut montrer que les déplacements du robot sont indépendants des changements de posture du robot, ils sont uniquement influencés par la gravité et les forces de contact (Wieber, 2000). Ainsi, ce sont les forces de contact avec le sol (passives) qui permettent d'influer sur le déplacement.

2.2.5 Systèmes non linéaires et stabilité

Les robots bipèdes sont des systèmes dynamiques non linéaires du deuxième ordre, on peut donc les représenter par une équation différentielle. Dans le cas où l'on ne considère pas les forces de contact avec le sol, l'équation correspondant au système de la figure 2.3 page 27 s'écrit :

$$\dot{x} = f(x) + g(x)u$$

où $x \in \mathbb{R}^n$ est un vecteur d'état du système et $u \in \mathbb{R}^m$ le vecteur de commande des articulations.

L'étude dans le plan de phase de l'évolution du vecteur d'état (fig.2.4) permet d'appréhender le caractère hybride de la locomotion bipède en observant le saut de

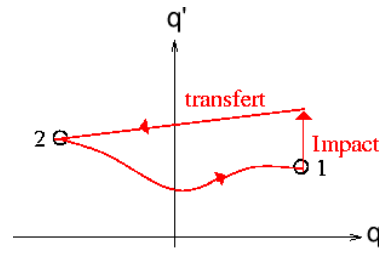
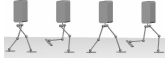
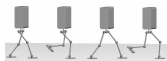


FIG. 2.5 – Cycle limite

vitesse à l'impact et le transfert instantané d'une phase d'appui à l'autre. Lorsque le pied de la jambe libre entre en contact avec le sol, un transfert se produit dans le plan de phase du point 1 au point 2 qui se trouve en général sur une trajectoire dynamique différente de la trajectoire d'origine. Si le point 2 se trouve sur la trajectoire d'origine (fig.2.5), alors le déplacement qui en résulte est périodique et correspond à un cycle limite.

On peut montrer que le déplacement régulier d'une machine bipède correspond à une variation quasi-périodique du vecteur d'état x du système (Manamani et al., 1998 ; Hurmuzlu et al., 2001 ; Grizzle et al., 1999). On a ainsi une trajectoire orbitale Ω dans le plan de phase. Une marche est dite *stable* s'il existe un voisinage de la trajectoire qui soit invariant. Si le système rejoint la trajectoire originale correspondant à Ω malgré des perturbations, alors le système est dit *asymptotiquement stable*. Pour de plus amples détails sur l'analyse de la stabilité des systèmes non-linéaires (Lyapunov, Poincaré), se référer à (Khalil, 1996). Les notions de stabilité des systèmes non-linéaires s'appliquent à l'étude de la marche stationnaire, mais les changements de trajectoires rentrent difficilement dans ce cadre d'étude. Or, pour un robot, il est primordial de pouvoir modifier son déplacement pour pouvoir s'adapter à son environnement. On considère alors plutôt la stabilisation d'un ensemble de trajectoires sur la base du critère de *viabilité*. Soit S l'ensemble des positions du robot qui correspondent à des situations de chute, c'est-à-dire l'ensemble des positions pour lesquelles une partie du robot autre que ses pieds est en contact avec le sol. On dira qu'un état x_i est *viable* si à partir de cet état, le robot est en mesure de réaliser un mouvement qui ne pénètre pas dans l'ensemble S (Wieber, 2000). Les états du robot sont non viables lorsque la chute est irrémédiable. On appelle *noyau de viabilité*, l'ensemble des états viables. Ainsi, les postures (équilibre statique) sont des positions viables, les trajectoires cycliques, ou les trajectoires qui se terminent sur un état viable restent également dans le noyau de viabilité. Si une trajectoire de référence reste dans les limites de ce noyau et à une certaine distance de sécurité de



ces limites, et si les mouvements réels du robot demeurent proches de cette trajectoire, alors les états du robot demeureront à l'intérieur du noyau. Une perturbation peut éloigner le robot de la trajectoire de référence et remettre en cause la viabilité des états suivants.

2.2.6 Critères de stabilité

Équilibre postural

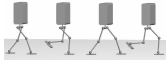
On considère un robot qui se trouve à l'arrêt en position debout. S'il subit une poussée dans le dos, il devra compenser le déplacement provoqué par la poussée pour maintenir son équilibre statique. c'est-à-dire pour maintenir la projection de son *cdg* (centre de gravité) à l'intérieur du polygone de sustentation. Pour réaliser cet effort, les forces de contact sont utilisées. Lorsque la poussée devient trop importante, les appuis deviennent insuffisants pour lutter contre le déplacement. La chute ne pourra alors être évitée que si le robot effectue un pas, c'est-à-dire s'il change d'appuis. La chute ne pourra donc pas toujours être évitée. La notion d'équilibre postural est primordiale en robotique bipède, en effet un robot ne sera utile que s'il est capable de s'arrêter pour effectuer une tâche donnée (avec ses mains par exemple) sans tomber. Cette notion est étroitement liée aux transitoires de marche.

Équilibre dynamique

Le *zmp* (zero moment point) correspond au *cdp* (centre de pression) des forces de contact (Goswami, 1999 ; Vukobratović et al., 1990). L'équilibre dynamique est garanti dès lors que le *zmp* demeure à l'intérieur du polygone de sustentation. Cette propriété n'est valide que dans le cas où tous les points de contact sont sur un même plan, ce qui réduit fortement ses applications. De plus, ce critère ne prend pas en compte la composante tangentielle des forces de contact. C'est néanmoins ce critère qui est le plus largement utilisé dans la littérature (Hirai et al., 1998). Nous le verrons, l'utilisation de contraintes sur les forces de contact permet de formuler plus efficacement le problème de l'équilibre dynamique (Wieber, 2000).

2.3 Contrôle de la marche

La coordination des mouvements du robot doit permettre d'assurer le bon déroulement des déplacements du robot en prenant en compte les caractéristiques présentées précédemment. Le problème du contrôle est de spécifier les couples des actionneurs des articulations U tels que le système se déplace sans chute.



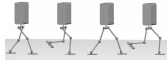
2.3.1 Poursuite d'une trajectoire pré-calculée

L'approche la plus couramment retenue dans la littérature pour le contrôle des robots marcheurs est la poursuite de trajectoires de référence pré-calculées qui décrivent le comportement désiré pour le robot (Hurmuzlu et al., 2001). La manière la plus simple de procéder est de spécifier les profils temporels des positions articulaires. Les trajectoires peuvent être définies par analogie avec le système anthropomorphe (Hemami et Farnsworth, 1977 ; Vukobratović et al., 1990) ou avec des systèmes mécaniques plus simples tels que les bipèdes passifs (McGeer, 1990 ; Thuilot et al., 1997). Il est également possible d'utiliser des oscillateurs (Van Der Pol's par exemple) (Kato et Mori, 1984). Les mouvements de référence peuvent également être décrits dans le domaine fréquentiel (Unuma et al., 1995). Les trajectoires de référence peuvent être liées aux couples articulaires (Sano et Furusho, 1990 ; Hurmuzlu, 1993a ; Goswami et al., 1996 ; Pratt et Pratt, 1998). Le recours à des techniques d'optimisation est également courant : les critères minimisés sont généralement de type énergétique ou quadratique en couples actionneurs (Cabodevilla et al., 1996 ; Chevallereau et al., 1997 ; Rostami et al., 1998 ; Roussel, 1998 ; Chessé et Bessonnet, 2001). La définition de fonctions objectifs associées aux paramètres des pas de la marche (vitesse de déplacement, longueur des pas...) permet également de définir des mouvements (Gri-shin et Formal'sky, 1990).

Dans le contexte de la poursuite de trajectoires, différentes lois de commande peuvent être utilisées (Hurmuzlu et al., 2001) : contrôleurs linéaires basés sur des correcteurs PID (Zheng et Hemami, 1984 ; Park et Kim, 1998 ; Furusho et Sano, 1990) ; commandes dynamiques (*computed torque*) (Hurmuzlu, 1993b ; Park et Kim, 1998) ; commandes de type mode glissant (*sliding mode control*) (Manamani et al., 1997) ; commandes impulsives (*impulse control*) (Chevallereau et al., 1997) ; commandes à structures variables (*variable structure control*) (Slotine et Li, 1991) ; commande optimale (Channon et al., 1996) ; commande adaptative (Yang, 1994 ; Pratt, 1998). Les méthodes sus-mentionnées se basent sur des comportements prédéterminés et ne permettent donc pas de modifier aisément les trajectoires pour adapter les mouvements à l'environnement (obstacles, sols inclinés...) ou aux perturbations (poussée, heurt, collision).

2.3.2 Adaptation en ligne de la marche

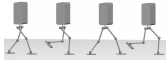
Le robot doit être capable de s'adapter à des environnements non connus a priori. En effet, le sol est rarement parfaitement plat et lisse, ce qui signifie que la position du centre de pression et les forces de contact ne peuvent pas être totalement prédites.



Certains chercheurs ont mis en place des techniques permettant d'adapter la marche en ligne en compensant les mouvements du *zmp* (par le mouvement du tronc par exemple) (Park et Chung, 1999 ; Takanishi et Lim, 1990). Les techniques de logique floue ont été utilisées pour réguler les forces de réaction du sol dans le cas de terrains irréguliers (Shih et al., 1991). Pour éviter les obstacles (escaliers, murs...), le robot doit pouvoir modifier sa trajectoire pour s'arrêter, contourner ou enjamber. L'une des techniques utilisées pour adapter la marche en temps réel est basée sur des ensembles de trajectoires prédéfinies parmi lesquelles le contrôleur choisit la meilleure candidate (Wagner et al., 1988). L'adaptation du mouvement aux contraintes liées aux changements de l'environnement ou à l'occurrence d'événements se fait en ligne en choisissant et en combinant des mouvements choisis parmi le répertoire de trajectoires (Wieber, 2000 ; Denk et Schmidt, 2001a ; Denk et Schmidt, 2001b). Des méthodes ont été proposées pour générer des trajectoires transitoires en temps réel (France et al., 1999 ; Rose et al., 1996). Des recherches portent également sur les techniques d'apprentissage par réseaux de neurones permettant de modifier les déplacements en ligne (Kun et Miller, 1996).

2.4 Synthèse

Les résultats théoriques concernant les robots bipèdes font généralement des hypothèses difficilement compatibles avec les conditions expérimentales. En particulier, les études traitent essentiellement de la marche stationnaire sur sol plat. Or ces machines n'auront d'utilité que si elles sont capables de démarrer, de s'arrêter à des instants arbitraires, et de se tenir en équilibre pendant qu'elles effectuent une tâche donnée. Les déplacements des systèmes doivent être robustes et s'adapter à un environnement inconnu et la poursuite d'une trajectoire de référence unique n'est donc pas viable. Bien que l'utilisation d'un ensemble de trajectoires paramétrées soit une voie d'exploration intéressante, il nous semble cependant irréaliste d'imaginer pouvoir générer tous les cas de figures qui pourront se présenter. Le critère de stabilité *zmp* est largement et parfois abusivement utilisé (Goswami, 1999 ; Wieber, 2000). Il est vrai que le cadre théorique de l'équilibre dynamique manque encore cruellement de résultats constructifs.



Chapitre 3

Analogies entre homme et robot

3.1 Introduction

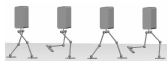
Les deux chapitres précédents nous ont permis de nous familiariser avec la notion de marche bipède. Comme il le sera répété tout au long de ce mémoire, ce travail de thèse s'est fortement inspiré des solutions mises en place par la nature pour résoudre le problème de la marche bipède. Nous présentons dans ce chapitre les thèmes sur lesquels nous pensons que des interactions entre sciences biologiques et sciences techniques sont possibles et souhaitables (Azevedo, 2001). Nous verrons dans un premier temps comment les connaissances de la biomécanique et des neurosciences peuvent avantageusement inspirer et aider le développement et le contrôle de systèmes marcheurs robotisés. Puis nous montrerons que le transfert de connaissances et d'expériences peut également se faire de la robotique vers la biomécanique dans le cadre de la marche artificielle humaine.

3.2 De l'homme vers le robot : conception et contrôle de machines bipèdes

Les connaissances issues de l'observation de la nature sont une source d'informations de première importance lorsqu'il s'agit de vouloir créer et animer un bipède. L'homme n'est-il pas finalement le robot bipède idéal (Duysens et al., 2002) ?

3.2.1 Conception de robots anthropomorphes

Il est évident que les prototypes de machines à deux pattes qui sont construits aujourd'hui tendent de plus en plus à copier l'humain. On distingue tout d'abord les robots humanoïdes japonais dont la ressemblance avec l'homme a essentiellement des objectifs de sociabilisation et de séduction (ASIMO, 2000 ; SDR-3X, 2000). On trouve ensuite les robots anthropomorphes dont les aspirations sont d'aider à mieux comprendre la marche humaine. Le caractère anthropomorphe porte généralement sur le squelette du robot (disposition des masses, dimensions...) et sur les actionneurs (ex : muscles artificiels pneumatiques) (Guihard et Gorce, 2001). Néanmoins,



les machines existantes sont loin d'égaliser les quelques trois cents degrés de liberté du corps humain et son système musculaire perfectionné. Quoi qu'il en soit, il n'est pas nécessaire d'atteindre un niveau de complexité très élevé pour étudier la marche humaine, les mouvements de base de la locomotion pouvant être réalisés avec quinze degrés de liberté (Sardain et al., 1997). Par ailleurs, la modélisation et la simulation sont également beaucoup plus accessibles dans le cas d'un prototype construit par l'homme et dont on connaît parfaitement le mécanisme. De plus, les comportements musculaires peuvent être simulés et ne nécessitent pas nécessairement de réalité mécanique. L'intérêt de tels robots est qu'ils peuvent servir de plate-forme expérimentale à des tests et mesures qui nécessiteraient une chirurgie invasive pour positionner les capteurs sur un individu. Le savoir-faire théorique de la robotique peut ainsi permettre de formaliser et valider des hypothèses sur la façon dont le cerveau réalise le contrôle des mouvements (Hanneton, 1999 ; VanDerKooij, 2000).

3.2.2 Contrôle de la marche robotisée

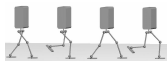
Les propriétés du contrôle de la locomotion humaine nous ont permis de mettre en évidence des principes de base dont la robotique ne peut à terme s'affranchir pour parvenir à des machines performantes, robustes et autonomes. Nous rappelons ici ces principes.

Allures de marche

La locomotion humaine comporte des marches stationnaires de confort et des transitoires aux allures diverses. Dans le cas des robots bipèdes, les deux aspects sont également indispensables pour évoluer dans un environnement inconnu et changeant.

Cycles optimaux

La marche d'un individu est cyclique et optimale par rapport à un critère variable en fonction du type d'allure recherchée (vitesse, endurance, précision...). La notion d'optimalité nous semble nécessaire dans le cas des machines, ne serait-ce que par rapport à leur autonomie et à leur durée de vie. Les transitoires de marche correspondent à des critères distincts à minimiser (énergie, temps, distance...). La transposition des patrons (allures) de marche biomécaniques ne nous semble pas être la meilleure façon d'aborder le problème. Une solution optimale chez l'homme ne l'est pas nécessairement pour la machine et réciproquement. Néanmoins, le recours à des méthodes d'optimisation basées sur des critères multiples semble être une voie intéressante pour la synthèse de la marche chez le robot.



Trajectoires de référence

On peut noter que, contrairement aux choix généralement faits en robotique, dans la nature, le contrôle de la marche ne correspond pas à un suivi de trajectoire. L'homme dispose de répertoires de séquences de référence, mais celles-ci ne sont pas des trajectoires articulaires. Ce répertoire est utilisé pour élaborer des stratégies en cas de nécessité, par exemple lors d'une perturbation.

Contrôleur

Si l'on ne peut pas comparer le cerveau à un ordinateur (Berthoz, 1997 ; Slotine, 2002), ils jouent tous deux le rôle de contrôleur de la marche. L'homme est capable de prédire et d'anticiper en utilisant des modèles : cela lui permet d'élaborer des stratégies de réaction et d'adaptation. Il mémorise et enregistre des séquences de mouvements qui lui permettent de trouver rapidement des solutions à certaines situations.

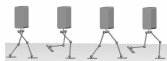
Les robots doivent être dotés de telles capacités afin d'éviter les situations bloquantes et les mouvements qui entraîneront une chute inévitable. On peut citer certains aspects pour lesquels la machine est plus performante que l'homme :

- calcul de relations mathématiques complexes,
- mémoire infailible,
- capacité de répétition parfaite,
- vitesse de transmission.

Le rôle des capteurs est essentiel dans cette boucle. Ils doivent être choisis et disposés sur la machine bipède de manière à fournir au contrôleur toute l'information nécessaire et pertinente. Rappelons que, chez l'humain, outre les capteurs de position articulaire, le contrôle de la marche repose essentiellement sur les capteurs d'effort de la plante de pieds, la vision et la centrale inertielle constituée par l'oreille interne (fusion multi-capteurs).

Équilibres postural et dynamique

Le contrôle de l'équilibre postural est indispensable pour que le bipède puisse effectuer des tâches à l'arrêt. Or, cet aspect est souvent négligé en robotique et il convient de s'en préoccuper. Par ailleurs, la marche dynamique exploite les impacts en transformant l'énergie potentielle due au déséquilibre avant en énergie cinétique. La maîtrise de ce phénomène tout en conservant l'équilibre dynamique est un point crucial, qui est souvent éludé en robotique en simplifiant le problème en limitant les impacts et en posant les pieds à plat.



3.3 Du robot vers l'homme : la marche artificielle

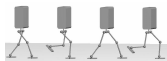
Si la nature est une source d'inspiration pour la technique bipède, la robotique peut également être d'un apport important pour la mobilité humaine. Et tout particulièrement dans le domaine de la réhabilitation de la marche (Menzel et D'Alusio, 2000 ; Rabishong, 1996) dont on peut citer deux exemples :

- l'électro-stimulation fonctionnelle (functional electrical stimulation : FES). Cette technique permet de stimuler artificiellement les muscles de membres paralysés, notamment pour la restauration de la locomotion chez des patients atteints de lésions médullaires¹. De nombreux programmes de recherche travaillent sur le sujet (Quintern, 1998 ; Riener, 1998). Le projet SUAW (stand up and walk) a développé un stimulateur qui est implanté sous la peau de l'abdomen et connecté par des électrodes aux nerfs et aux muscles des patients. Ce stimulateur est contrôlé de l'extérieur via une antenne. Cette antenne est liée à un programmeur portable qui délivre la bonne impulsion au bon moment comme le ferait normalement le système nerveux central. Les patients communiquent avec ce programmeur par des boutons placés sur le déambulateur (Guiraud et al., 2002 ; Guiraud et al., 2001). La nécessité du passage à une commande en boucle fermée ouvre des voies de collaboration avec la robotique (LIRMM et al., 2002 ; Popovic et al., 2001 ; Yamaguchi et Zajac, 1990). Les prototypes robotiques pourront alors servir de plate-forme pour les validations et les tests matériels et logiciels avant implantation du patient.
- les prothèses actives de membres inférieurs ainsi que les exosquelettes. Les exosquelettes sont des structures anthropomorphiques artificielles destinées à augmenter les capacités physiques (vitesse, force et endurance) de celui qui les utilise (Weiss, 2001 ; Vickers, 2000). Les orthèses actives peuvent être également utilisées dans les programmes de rééducation fonctionnelle. La commande de ces dispositifs entre parfaitement dans le savoir-faire de la robotique.

3.4 Conclusion

L'observation de la locomotion naturelle peut être une base de réflexion pour l'élaboration de solutions en robotique bipède. Encore une fois, il ne s'agit pas de "singer" l'homme, mais de s'en inspirer.

¹médullaire = en rapport avec la moelle (il s'agit généralement de la moelle épinière et non de la moelle osseuse)



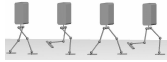
Les principes de base que l'on pourra retenir pour la marche bipède sont :

1. ne pas effectuer de suivi précis de trajectoires de référence ;
2. distinguer et traiter différemment les cas de la marche stationnaire et non stationnaire (dont l'équilibre postural) ;
3. être capable de passer d'un type de marche à un autre (transitoires) (Abernethy et al., 2002) ;
4. considérer des allures optimales selon des critères variables adaptés aux objectifs de la marche (énergétiques, temporels...) ;
5. anticiper et adapter les mouvements pendant le déplacement.

Les capteurs physiologiques sont nombreux et complémentaires. Ils jouent un rôle crucial dans la locomotion humaine. Le contrôle de la marche repose sur la réception et l'exploitation des informations qui en sont issues. De la même façon en robotique, l'utilisation et la disposition judicieuse de capteurs sont essentielles.

Si biomécanique et neurosciences peuvent être utiles lorsqu'il s'agit d'élaborer et contrôler des robots marcheurs, la robotique peut inversement collaborer à l'essor des sciences de l'humain. Les machines bipèdes sont plus aisément modélisables que le corps humain et leur instrumentation beaucoup plus simple et sûre. Les théories concernant la locomotion peuvent ainsi être évaluées sur ces plates-formes. De plus, les besoins actuels des techniques liées à la marche artificielle font clairement appel à des outils issus de l'automatique.

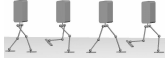
Comme nous le verrons dans la deuxième partie, les conceptions des prototypes bipèdes BIP (INRIA, LMS) et RABBIT (GdR CNRS, ROBEA) se sont largement appuyées sur les aspects anthropomorphes.



Conclusion

La nature a largement inspiré les scientifiques et les ingénieurs depuis fort longtemps. L'utilisation de modèles biologiques a notamment conduit aux notions d'*intelligence artificielle* (AI) et de *vie artificielle* (Alife) (Brooks, 2001). La première s'intéresse aux propriétés du vivant concernant la perception, la reconnaissance et la génération de l'action. À titre d'exemple, les réseaux de neurones artificiels matérialisés par des composants électroniques ou simulés par informatique imitent le fonctionnement des cerveaux animaux. La seconde discipline s'intéresse à l'évolution, à la reproduction et au métabolisme avec l'idée que les organismes vivants ne sont finalement que des machines dont les composants sont des molécules. Les processus évolutionnistes de Darwin sont ainsi mis en oeuvre en robotique mobile (Floreano et Urzelai, 2000). En ce qui concerne les robots humanoïdes, certains chercheurs espèrent qu'ils permettront de tester les modèles proposés par la psychologie développementale et les sciences cognitives (Adams et al., 2000).

Ces approches s'intéressent aux mécanismes de fonctionnement naturels et tentent de les imiter avec bien des difficultés (Brooks, 2001). Notre objectif est bien distinct : nous n'aspirons pas à la construction d'un cerveau artificiel pour le robot (Brooks et Stein, 1993), mais plutôt à la conceptualisation de la marche. Il s'agit de comprendre et définir les principes de base de la locomotion humaine et de les exploiter dans le but de caractériser le contrôle de la locomotion bipède.



Deuxième partie

Des robots à leurs modèles

Sommaire

- ★ **Problématique**

présenter les robots bipèdes et leurs modèles dynamiques qui seront utilisés pour valider en simulation et expérimentalement nos résultats

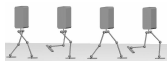
- ★ **Plates-formes expérimentales**

présentation du robot BIP dans ses versions 3D et 2D et du robot RABBIT sous-actionné

- ★ **Modélisation dynamique**

description des techniques utilisées pour modéliser les comportements des systèmes

- ★ **Conclusion**



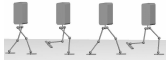
Problématique

Connaître les robots pour mieux les utiliser.

Les études et résultats présentés dans cette thèse ont été volontairement dirigés vers l'expérimentation et la simulation. Ce parti pris suppose une bonne connaissance des robots concernés et un soin tout particulier à apporter à leur modélisation. Nous nous sommes attachés à développer de nouvelles approches pour le robot BIP de l'INRIA². La particularité essentielle de ce robot est d'être anthropomorphe de par ses dimensions. Cette caractéristique s'inscrit dans les objectifs énoncés dans la partie précédente : développer des plates-formes expérimentales pour l'étude de la marche humaine naturelle et artificielle. Ce robot a été construit en deux étapes, les deux versions successives ayant été utilisées pour nos travaux ; une version à six articulations actives pouvant évoluer dans le plan sagittal : BIP 2D et une version à quinze articulations actives pouvant se mouvoir dans l'espace à trois dimensions : BIP 3D. Nous avons également appliqué certains de nos résultats (quatrième partie) au cas du robot RABBIT sous actionné.

Dans un premier temps nous allons présenter les prototypes BIP et RABBIT, puis nous verrons quels modèles dynamiques nous avons utilisés pour représenter les robots selon nos objectifs (simulation, commande...).

²BIP a été conçu et construit dans le cadre d'une collaboration entre le LMS de Poitiers et l'INRIA RHÔNE-ALPES



Chapitre 1

Plates-formes expérimentales

1.1 Introduction

La robotique bipède utilise souvent des simulateurs aux hypothèses plus ou moins restrictives et a tendance à oublier que derrière les modèles se cachent des machines réelles. Le passage de la simulation à l'expérimentation déjà délicat, l'est d'autant plus que les simplifications ont été importantes et les hypothèses théoriques sévères. Ces remarques concernent bien entendu la mécanique et les actionneurs, mais également la programmation temps réel. C'est la raison pour laquelle nous consacrons ce chapitre à la présentation des robots sur lesquels nous avons travaillé dans cette thèse et dont les caractéristiques matérielles ont été prises en compte autant que possible en simulation. Les différents résultats théoriques et expérimentaux rapportés dans ce mémoire portent sur trois prototypes : le robot BIP dans sa configuration complète 3D, le robot BIP dans sa configuration simplifiée 2D et le robot RABBIT 2D.

Nous nous proposons de présenter ici les caractéristiques principales de ces systèmes qui seront utilisés tout au long de ce mémoire pour valider nos développements théoriques par des expérimentations et des simulations.

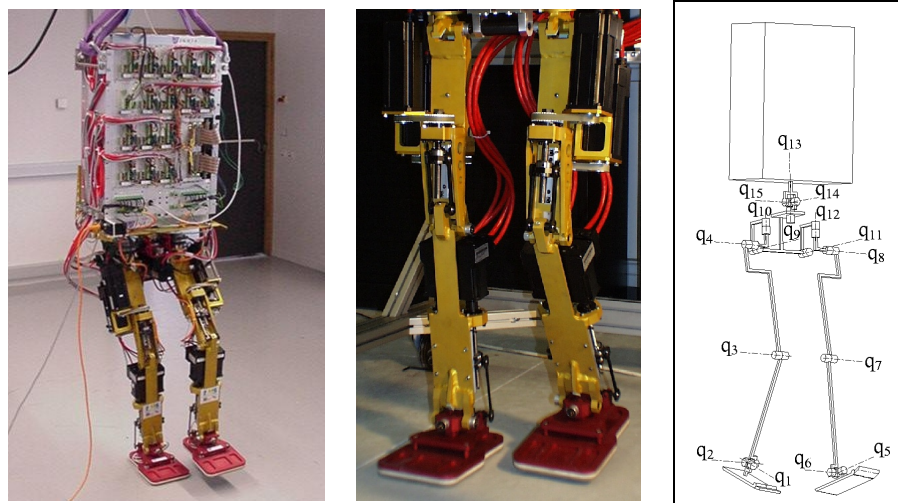


FIG. 1.1 – Le robot BIP 3D

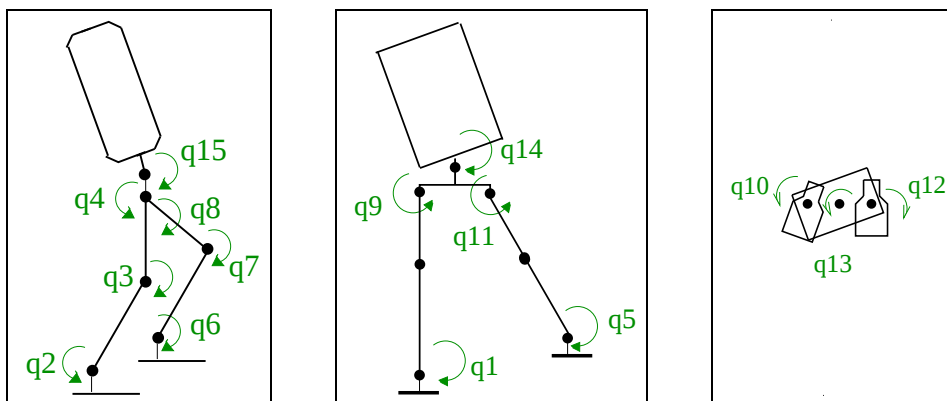
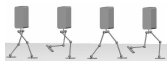


FIG. 1.2 – De gauche à droite : mouvements dans le plan sagittal, mouvements dans le plan frontal, mouvements dans le plan horizontal.

1.2 Robot BIP dans sa configuration 3D

1.2.1 Architecture mécanique et cinématique

La conception du robot BIP s'est inspirée des données cinématiques et des capacités dynamiques anthropomorphes (Espiau et Sardain, 2000 ; Espiau et Bip team, 1997). BIP mesure 1 m 80 pour 105 kg. Il a été estimé que quinze articulations (fig.1.1) étaient suffisantes pour reproduire les démarches humaines de base (marche dynamique, montée et descente d'escalier). Ces articulations peuvent être regroupées en fonction du plan (fig.1.2) dans lequel se fait le mouvement engendré :

- 6 articulations (2 chevilles, 2 genoux, 2 hanches) permettent la flexion/extension dans le plan sagittal. Leurs rotations contrôlent le déplacement du corps vers l'avant.
- 5 articulations (2 chevilles, 2 hanches, 1 “vertèbre lombaire”) permettent l'abduction/adduction dans le plan frontal. Leurs rotations contrôlent l'équilibre latéral.
- 3 articulations (2 hanches, 1 “vertèbre lombaire”) permettent la rotation interne/externe dans le plan horizontal. Leurs rotations contrôlent les changements de direction (virages, demi tours...).
- 1 articulation supplémentaire entre le tronc et le bassin permet de dissocier les systèmes de déplacement et d'équilibre dans le plan sagittal.

Les actionneurs sont des moteurs à courant continu sans balais. Cinq articulations sont équipées de réducteurs harmonic-drive (rotations $q_9, q_{10}, q_{11}, q_{12}, q_{13}$ dans la figure 1.2). Les dix autres transmissions sont des systèmes de vis-écrou à rouleaux satellites - donc à mouvement réversible - avec biellettes (fig.1.3). Leurs rapports de réduction varient en fonction de la configuration (Sardain et al., 1998). Ces transmissions sont disposées en parallèle au niveau des chevilles et de la liaison tronc/bassin.

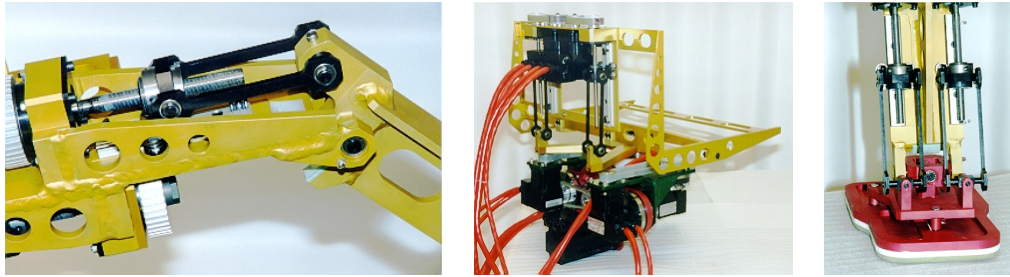


FIG. 1.3 – Transmissions spécifiques (de gauche à droite : genou, tronc, cheville)

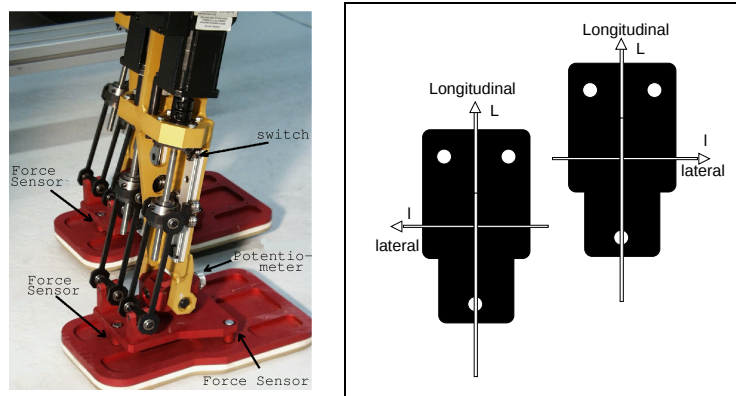


FIG. 1.4 – De gauche à droite : différents capteurs disponibles sur BIP - disposition des capteurs d'effort sur les pieds

1.2.2 Capteurs

Le robot est équipé de différents types de capteurs (fig.1.4). Les codeurs des moteurs-varianteurs fournissent la position angulaire relative des axes moteurs. Des potentiomètres placés directement sur les articulations permettent de déterminer leur position angulaire absolue à l'initialisation du système. Trois capteurs d'effort sont placés sur chaque pied, permettant de mesurer la composante verticale de la force de réaction du sol, les deux moments correspondants et la position (l, L) du centre de pression (cdp) dans le plan de la semelle (fig.1.4).

1.2.3 Architecture logicielle

Les unités de puissance et les différentes cartes électroniques nécessaires au pilotage des différents moteurs sont incorporées dans une armoire qui fait office de tronc au robot. Le robot est alimenté par un cordon ombilical. Les pilotes (drivers) et programmes sont exécutés sur une carte MOTOROLA Mvme162 basée sur un processeur

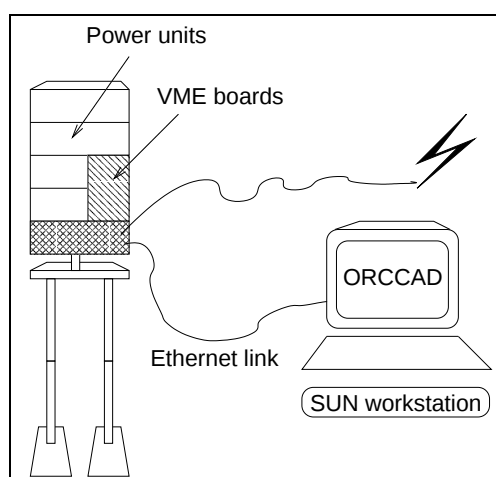


FIG. 1.5 – Architecture logicielle du système BIP à 15 ddl

68040-32MHz. Le système d'exploitation temps réel multi-tâches multi-processeurs VXWORKS¹ est utilisé. Les programmes sont d'abord conçus et compilés sur une station de travail, puis téléchargés grâce à une liaison Ethernet via une commande VXWORKS vers la carte cible Mvme162 (fig. 1.5). Ils sont exécutés sur le robot et la liaison Ethernet permet de dialoguer avec le logiciel embarqué, notamment pour faire de l'acquisition de données. Hormis les pilotes, tous les programmes sont réalisés à partir de l'environnement ORCCAD.

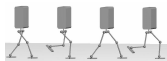
1.2.4 ORCCAD

ORCCAD (**O**pen **R**obot **C**ontroller **C**omputer **A**ided **D**esign) (TheOrccadTeam, 1998) est à la fois une méthodologie et un environnement logiciel. Il a été développé à l'INRIA et est destiné à l'implantation de lois de commande sur des plates-formes robotiques. Il est particulièrement adapté aux systèmes qui interagissent fortement avec leur environnement (actionneurs, capteurs). La notion clef de la méthodologie ORCCAD est la **Tâche Robot** qui représente une action robotique élémentaire. Une application robotique complète (mission) est définie comme la composition logique de plusieurs Tâches Robot qui s'exécutent en séquence et/ou en parallèle. Cette composition est appelée **Procédure Robot**.

La Tâche Robot :

Les Tâches Robot sont des schéma-blocs composés de modules qui communiquent entre eux via des ports de données. Elles rassemblent les aspects continus d'une loi de commande et les aspects discrets d'un comportement logique.

¹<http://www.windriver.com/>



Une Tâche Robot est définie en spécifiant :

- (1) une *Ressource Physique* qui représente le système (mécanique, électrique, hydraulique...) que l'on souhaite contrôler,
- (2) une *Loi de Commande* considérée comme une chaîne de calculs invariante entre capteurs et actionneurs, exécutée de façon périodique jusqu'à ce que l'objectif soit atteint ou jusqu'à satisfaction d'une condition donnée (continu),
- (3) un *Comportement Logique* qui est représenté par plusieurs états logiques. Le passage d'un état à l'autre survient lorsque certaines conditions sont satisfaites (discret). Le comportement logique est composé de trois états : initialisation, calcul et fin. Les transitions entre états se font grâce à des événements discrets qui sont générés au niveau des modules et reliés à un module dédié : le module automate.

La Procédure Robot :

Les Procédures Robot sont utilisées pour composer hiérarchiquement et logiquement les Tâches Robot pour créer une mission complète. Une fois ces procédures spécifiées, la Procédure Robot est traduite en code temps réel C++ pour un système d'exploitation (*OS*) donné. Un des avantages de ORCCAD est de faciliter énormément le travail de l'utilisateur en générant automatiquement le code temps réel et en gérant les événements discrets déclarés et classifiés par ordre de "danger" pour le robot. Nous verrons par la suite comment utiliser cet outil pour implanter un contrôleur de suivi de trajectoires pour BIP.

1.3 Robot BIP dans sa configuration 2D

Une version simplifiée du robot BIP présenté §.1.2 a également été construite. Cette version correspond aux 2 jambes de BIP, soit 8 articulations. Un chariot muni de 4 roues supposées sans frottement avec le sol est relié au bassin du robot grâce à une barre de liaison (fig.1.6). On dispose ainsi d'un bipède à 6 degrés de liberté. Le robot est libre de ses mouvements dans le plan sagittal et contraint dans le plan frontal. Les caractéristiques mécaniques, électriques et logicielles sont identiques à celles de la version complète. L'armoire de commande n'est pas portée par le robot (fig. 1.7). Dans cette configuration le robot mesure 95 cm pour 46 kg. Il est possible de rajouter des plaques en fonte sur le bassin du robot, pour remonter son centre de masse.

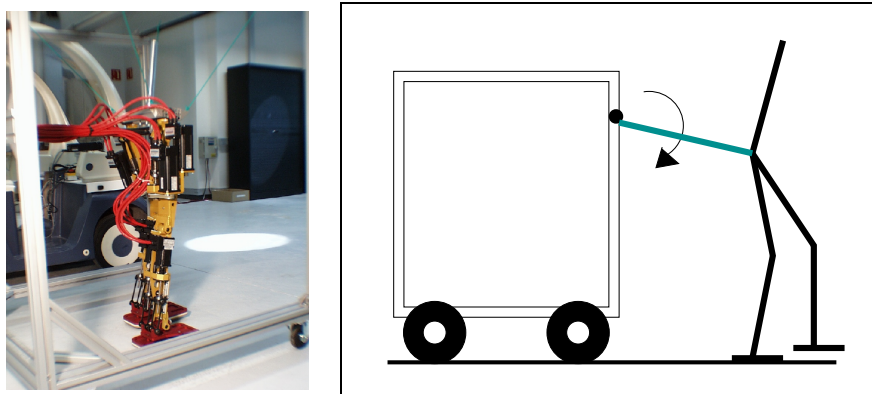
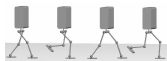


FIG. 1.6 – BIP 2D

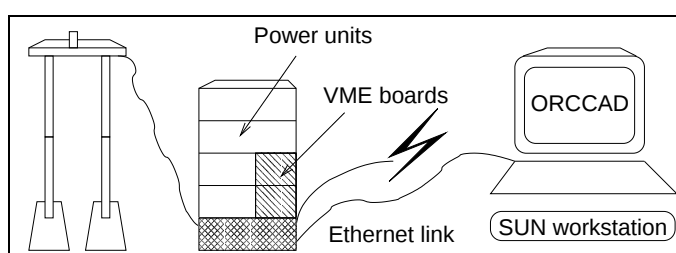


FIG. 1.7 – Architecture logicielle du système BIP à 6 ddl

Remarque : Pour information nous indiquons ci-après la répartition des masses dans le cas du corps humain et des squelettes BIP 3D et BIP 2D.

	% haut du corps /poids corps	% masse cuisse /poids corps	% masse tibia /poids corps	% masse pied /poids corps
BIP 3D	56 %	14 %	5,6%	2,4 %
BIP 2D	0 %	32 %	13%	5 %
Humain	67 %	10,5 %	4,5%	1,5 %

1.4 Le robot RABBIT

Le prototype RABBIT a été conçu et réalisé par un groupe de recherche composé de huit laboratoires français² dans le cadre de deux projets CNRS successifs (GdR puis ROBEA). Le dimensionnement a été choisi de manière à ce que le robot puisse marcher et courir. Le système est composé de deux parties : le robot proprement dit et un système de guidage en rotation (fig.1.8). Le système comporte 5 segments (tronc, cuisses, tibias) et 4 actionneurs (hanches et genoux). La hauteur totale du robot est de 1,425 m pour 40 kg. Le système de guidage est constitué d'une colonne centrale et d'une barre radiale. Le tronc du robot peut pivoter librement autour

²IRCCyN, LAG, LGIPM, LIRMM, LMS, LRP, LVR, INRIA

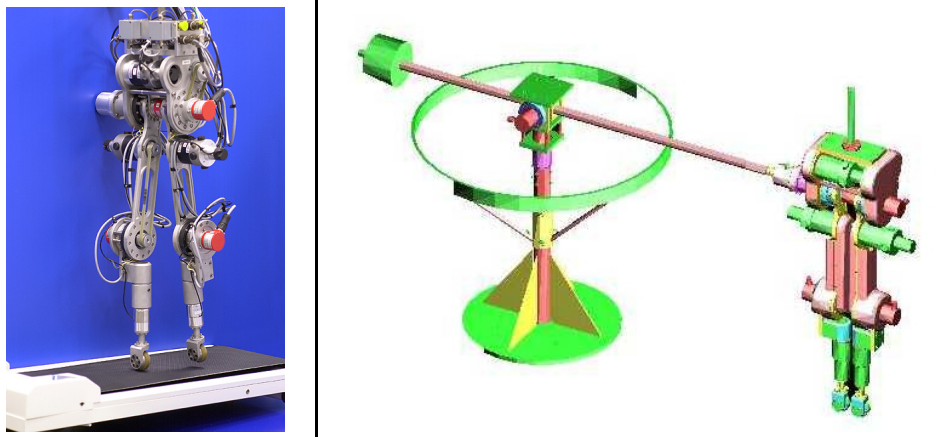
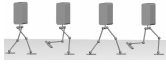
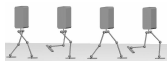


FIG. 1.8 – Le robot RABBIT

de l'axe de la barre. Ainsi, la barre maintient la tangence entre le plan sagittal du robot, et une sphère de centre la colonne centrale dont le rayon vaut la longueur de la barre. Chaque tibia est muni d'une roulette orientée selon le plan frontal. Le contact pied/sol est supposé ponctuel. Les actionneurs sont des moteurs à courant continu. Chaque articulation motorisée est munie d'un capteur de position monté directement sur l'articulation. Ces capteurs mesurent la position absolue entre les corps qui constituent l'articulation. La position du tronc est mesurée par un codeur incrémental. Chaque cheville est munie d'un capteur d'effort 2 axes à l'extrémité de chaque tibia. La particularité de ce robot est d'être sous-actionné en phase de simple appui par sa structure (absence de pieds et mouvements du tronc).

1.5 Conclusion

Le robot BIP, principal support expérimental de cette thèse, permet du fait de ses caractéristiques anthropomorphes des comparaisons avec les données récoltées chez l'homme (troisième partie). Il est également destiné à servir de plate-forme expérimentale pour des études biomécaniques et pour les recherches en électro-stimulation fonctionnelle (LIRMM et al., 2002). Nous avons utilisé le robot RABBIT pour ses caractéristiques de sous-actionnement afin de vérifier la capacité de notre commande à être appliquée dans ce cadre (quatrième partie).



Chapitre 2

Modélisation dynamique

2.1 Introduction

Comme nous l'avons déjà mis en avant précédemment, la modélisation et la simulation sont moins complexes dans le cas d'un bipède artificiel qui a été conçu et développé par l'homme que pour le corps humain (muscles, tendons, fatigue...). Dès lors que l'on s'intéresse à la marche, les problèmes majeurs de la modélisation dynamique concernent la prise en compte et la gestion des efforts de contact avec le sol. Les travaux de (Wieber, 2000 ; Génot, 1998 ; Brogliato, 1996 ; Hurmuzlu et al., 2001 ; Pfeiffer et Glocker, 1996) nous ont servi de trame pour cette partie.

2.2 Dynamique des robots

Note : tout au long de ce document nous noterons $\dot{x}$ la dérivée première (vitesse) de x et $\ddot{x}$ la dérivée seconde (accélération) de x .

2.2.1 Dynamique continue d'un robot non soumis à des efforts de contact

Sous l'hypothèse de rigidité des structures, l'équation de la dynamique peut s'exprimer sous la forme Lagrangienne suivante :

$$\boxed{M(q)\ddot{q} + N(q, \dot{q})\dot{q} + G(q) = \Gamma} \quad (2.1)$$

- $q \in \mathbb{R}^n$ est le vecteur des variables de configuration (positions articulaires, position et orientation absolues du robot),
- $\Gamma = [\tau, 0]^T \in \mathbb{R}^n$ est le vecteur des efforts généralisés. $\tau = TU \in \mathbb{R}^{n'}$ est le vecteur des couples articulaires, $U \in \mathbb{R}^m$ est le vecteur des couples moteurs, T est la matrice $n' \times m$ des répartitions des actions des moteurs sur les articulations, ($n' = n - 6$ et $m < n'$)
- M est la matrice d'inertie de dimension $n \times n$,
- N est la matrice $n \times n$ des effets centrifuges, gyroscopiques et de Coriolis,
- G est le vecteur des forces de gravité généralisées de dimension n .

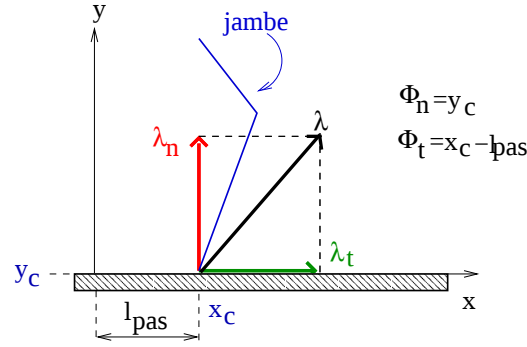
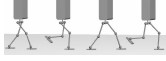


FIG. 2.1 – Notations utiles pour la représentation des contacts

2.2.2 Dynamique continue d'un robot soumis à des efforts de contact

La dynamique d'un robot soumis à des efforts de contact s'écrit sous la forme générale :

$$\boxed{M(q)\ddot{q} + N(q, \dot{q})\dot{q} + G(q) = \Gamma + \Gamma_c} \quad (2.2)$$

où : Γ_c sont les couples engendrés par des forces de contact telles que les contacts avec le sol, une poussée dans le dos ou bien encore le heurt d'une marche d'escalier...

Contrainte de liaison

Dans le cas de contact avec le sol, lorsque des points appartenant au robot sont en contact avec le sol, on peut écrire une équation dite de fermeture portant sur les positions de ces points :

$$\phi(q) = \begin{pmatrix} \phi_n(q) \\ \phi_t(q) \end{pmatrix} = 0 \quad (2.3)$$

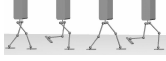
$\phi_n(q)$ porte sur la position suivant l'axe Oy des points de contact et $\phi_t(q)$ sur leur position suivant l'axe Ox (fig.2.1).

Couples de contact

On peut exprimer les couples de contact sous la forme suivante :

$$\Gamma_c = C(q)^T \lambda(q, \dot{q}) \quad (2.4)$$

où : $C(q)$ est la matrice jacobienne des contraintes de positions des points sur lesquels s'appliquent les forces de contact, λ représente ces forces.



Lorsque les efforts de contact correspondent à des efforts de contact avec le sol, les couples de contact s'expriment sous la forme :

$$\boxed{\Gamma_c = C_n(q)^T \lambda_n(q, \dot{q}) + C_t(q)^T \lambda_t(q, \dot{q})} \quad (2.5)$$

Ces efforts correspondent à un ensemble de forces s'exerçant sur chaque point posé sur le sol, orthogonalement ou parallèlement au sol (d'où les dénominations de forces normales et tangentielles et les indices $_n$ et $_t$). Les multiplicateurs de Lagrange λ_n et λ_t traduisent l'amplitude de ces efforts.

Contraintes unilatérales

Les contraintes unilatérales traduisent le fait que les points de contact ne peuvent en aucun cas descendre au dessous du niveau du sol mais qu'ils peuvent décoller à tout moment :

$$\ddot{\phi}_n(q) \geq 0 \quad (2.6)$$

Par dérivations successives de ϕ_n on obtient :

$$\begin{cases} \dot{\phi}_n(q) = \frac{\partial \phi_n(q)}{\partial q} \dot{q} = C_n(q) \dot{q} \\ \ddot{\phi}_n(q) = C_n(q) \ddot{q} + \dot{C}_n(q) \dot{q} = C_n(q) \ddot{q} + s_n(q, \dot{q}) \end{cases} \quad (2.7)$$

La contrainte sur l'accélération normale (2.6) devient alors :

$$\boxed{C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0} \quad (2.8)$$

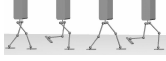
Conditions de complémentarité

Des conditions de complémentarité rendent compte du fait que lorsqu'un contact se rompt, la force normale correspondante doit être nulle (Pfeiffer et Glocker, 1996 ; Wieber, 2000). Ces conditions de complémentarité entre les forces d'appui normales λ_n et la contrainte de fermeture ϕ_n s'écrivent :

$$\boxed{\lambda_n^T(q, \dot{q}) \ddot{\phi}_n(q) = 0, \quad \lambda_n(q, \dot{q}) \geq 0, \quad \ddot{\phi}_n(q) \geq 0} \quad (2.9)$$

Hypothèse de non glissement

Dès lors qu'une extrémité entre en contact avec le sol, elle peut glisser dessus (Génot, 1998 ; Wieber, 2000). Des forces de frottements entrent alors en jeu et s'opposent au glissement. Selon le modèle de frottements secs d'Amontons-Coulomb, lorsque deux



objets sont en contact, il n'y a pas glissement si l'amplitude de la force tangentielle engendrée par le contact demeure dans le cône de frottement, soit :

$$\|\lambda_t\| \leq \mu_0 \lambda_n \quad (2.10)$$

où : μ_0 est le coefficient de frottement lié aux matériaux en contact.

Si l'on suppose que la condition de non glissement est satisfaite, cela signifie que les déplacements latéraux des extrémités sont nuls, ce qui se traduit par une contrainte sur les positions tangentielles des points de contact qui s'écrit :

$$\ddot{\phi}_t(q) = 0 \quad (2.11)$$

Par dérivations successives de ϕ_t , on obtient :

$$\begin{cases} \dot{\phi}_t(q) = \frac{\partial \phi_t(q)}{\partial q} \dot{q} = C_t(q) \dot{q} \\ \ddot{\phi}_t(q) = C_t(q) \ddot{q} + \dot{C}_t(q) \dot{q} = C_t(q) \ddot{q} + s_t(q, \dot{q}) \end{cases} \quad (2.12)$$

La contrainte sur l'accélération tangentielle (2.11) devient alors :

$$C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0 \quad (2.13)$$

Dynamique continue complète

La formulation de la dynamique continue complète qui prend en compte à la fois les contraintes unilatérales et les conditions de complémentarité sous hypothèse de non glissement s'écrit finalement :

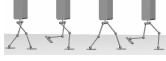
$$\begin{cases} M(q) \ddot{q} + N(q, \dot{q}) \dot{q} + G(q) = \Gamma + C_n(q)^T \lambda_n(q, \dot{q}) + C_t(q)^T \lambda_t(q, \dot{q}) \\ C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0 \\ C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0 \\ \lambda_n(q, \dot{q}) \geq 0 \\ \lambda_n^T(q, \dot{q}) (C_n(q) \ddot{q} + s_n(q, \dot{q})) = 0 \end{cases} \quad (2.14)$$

Problème d'optimisation quadratique équivalent

La dynamique peut s'écrire en appliquant le principe de Gauss de moindre déviation (l'accélération d'un système de solides soumis à des contraintes s'écarte le moins possible de l'accélération qu'aurait ce même système sans ces contraintes) sous la forme d'un problème d'optimisation quadratique (Wieber, 2000), qui s'écrit :

$$\begin{cases} \min_{\ddot{q}} \frac{1}{2} \ddot{q}^T M(q) \ddot{q} + \ddot{q}^T (-\Gamma + N(q, \dot{q}) \dot{q} + G(q)) \\ C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0 \\ C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0 \end{cases} \quad (2.15)$$

Note : cela est vrai uniquement dans le cas du non glissement.



Les conditions d'optimalité de ce problème sont celles de Kuhn-Tucker. Elles correspondent aux multiplicateurs de Lagrange c'est-à-dire aux efforts λ_n et λ_t . Tout l'intérêt d'une telle formulation réside dans la gestion simultanée des contacts et des décollements lors des différentes phases de la marche (Pfeiffer et Glocker, 1996). Les algorithmes de résolution des problèmes quadratiques (2.15) étant actuellement plus performants que ceux des problèmes de complémentarité non linéaire (2.14), l'utilisation de la formulation (2.15) est plus favorable d'un point de vue numérique (Génot, 1998).

2.2.3 Impacts

Lorsqu'un solide entre en contact avec le sol, on appelle *impact* la déviation instantanée de sa vitesse induite par la contrainte unilatérale de non pénétration. Il est possible d'exprimer la dynamique du choc par (Lötstedt, 1982) :

$$M(q)(\dot{q}^+ - \dot{q}^-) = C(q)^T \Lambda(q, \dot{q}) = C_n(q)^T \Lambda_n(q, \dot{q}) + C_t(q)^T \Lambda_t(q, \dot{q}) \quad (2.16)$$

où :

- $\dot{q}^-$ et $\dot{q}^+$ sont respectivement les vitesses juste avant et juste après l'impact,
- $\Lambda = [\Lambda_n \ \Lambda_t]^T$ est le vecteur des forces impulsionnelles.

En faisant l'hypothèse que les points de contact ne glissent pas suite à l'impact, on déduit :

$$C_t(q)\dot{q}^+ = 0 \quad (2.17)$$

De même en faisant l'hypothèse d'un choc inélastique (les points de contact ne décollent pas suite à l'impact) :

$$C_n(q)\dot{q}^+ = 0 \quad (2.18)$$

L'équation (2.16) peut s'écrire :

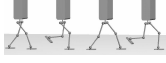
$$\dot{q}^+ = \dot{q}^- + M^{-1}(q)C(q)^T \Lambda \quad (2.19)$$

Le multiplicateur de lagrange Λ est obtenu en utilisant les équations précédentes :

$$\Lambda = -(C(q)M(q)^{-1}C(q)^T)^{-1}C(q)\dot{q}^- \quad (2.20)$$

Le problème de la dynamique impulsionnelle s'écrit alors :

$$\boxed{\begin{cases} \dot{q}^+ = \dot{q}^- + M^{-1}(q)C(q)^T \Lambda \\ \Lambda = -(C(q)M(q)^{-1}C(q)^T)^{-1}C(q)\dot{q}^- \end{cases}} \quad (2.21)$$



Problème d'optimisation quadratique équivalent

Comme pour la dynamique continue, le problème de l'impact, qui consiste à calculer la vitesse $\dot{q}^+$ après le choc, peut, par extension du principe de Gauss, se résumer sous la forme d'un problème quadratique (Wieber, 2000) :

$$\left\{ \begin{array}{l} \min_{\dot{q}^+} \frac{1}{2}(\dot{q}^+ - \dot{q}^-)^T M(q)(\dot{q}^+ - \dot{q}^-) \\ C_n(q)\dot{q}^+ = 0 \\ C_t(q)\dot{q}^+ = 0 \end{array} \right. \quad (2.22)$$

Cas du glissement

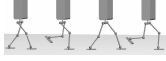
Dans le cas où le modèle ne s'affranchit pas des glissements, la dynamique ne peut pas être obtenue par application directe du principe de moindre déviation. Le calcul est alors beaucoup plus complexe car aboutissant à la formulation d'un problème de complémentarité non linéaire (Génot, 1998 ; Pfeiffer et Glocker, 1996). Ce modèle dynamique introduit par ailleurs de nombreuses inconsistances (Wieber, 2000). Nous avons fait le choix de nous placer dans des cas de figure où le robot ne glisse pas, sachant que pour une activité de marche cela correspond à un comportement normal.

Synthèse de la dynamique

Les équations (2.15) et (2.22) représentent la dynamique complète du robot (sous hypothèse de non glissement) et intègrent les lois de contact et la gestion des impacts :

$$\begin{array}{l} \textit{Dynamique continue} : \\ \min_{\ddot{q}} \frac{1}{2}\ddot{q}^T M(q)\ddot{q} + \ddot{q}^T (-\Gamma + N(q, \dot{q})\dot{q} + G(q)) \\ C_n(q)\ddot{q} + s_n(q, \dot{q}) \geq 0 \\ C_t(q)\ddot{q} + s_t(q, \dot{q}) = 0 \\ \\ \textit{Impact} : \\ \min_{\dot{q}^+} \frac{1}{2}(\dot{q}^+ - \dot{q}^-)^T M(q)(\dot{q}^+ - \dot{q}^-) \\ C_n(q)\dot{q}^+ = 0 \\ C_t(q)\dot{q}^+ = 0 \end{array} \quad (2.23)$$

Les aspects numériques liés à la simulation de la dynamique seront abordés dans la quatrième partie du mémoire.



2.2.4 Résolution automatique de la dynamique

L'application ROBOTYN développée par F. Génot et modifiée par P.-B. Wieber (Génot, 1998 ; Wieber, 2000) permet de générer les modèles de robots ayant un grand nombre de degrés de liberté. Cette application utilise le logiciel MAPLE. L'utilisateur doit fournir deux fichiers : *cinematique.maple* et *dynamique.maple*. Le fichier *cinematique.maple* précise les paramètres correspondant à la représentation de Denavit Hartenberg modifiée par Khalil et Kleinfinger (Khalil et Kleinfinger, 1986) de chaque solide du robot. Le fichier *dynamique.maple* contient pour chaque solide du robot sa masse, la position de son centre de masse et sa matrice d'inertie. À partir des deux fichiers écrits par l'utilisateur, ROBOTYN permet de générer les matrices M , N , G , C et s de la dynamique du système libre dans l'espace ou contraint.

2.3 Application aux prototypes

2.3.1 Robot BIP 2D

Le robot BIP 2D a été utilisé en simulation et en expérimentation dans deux configurations : robot suspendu au dessus du sol et robot posé sur le sol. Pour la seconde version nous avons utilisé deux modèles : le premier considère que le robot a son pied de support fixé au sol et que le transfert des appuis est instantané ; le second considère que le pied de support est libre de décoller et autorise le double support.

- **Robot suspendu** : dans les phases de réglages et d'initialisation du système, le robot n'est pas posé sur le sol. Il est maintenu par le bassin à l'aide de sangles de sécurité. Nous avons choisi de positionner la base de référence au niveau de la hanche de manière à symétriser le rôle des deux jambes. Dans cette configuration, le système comporte $n = 6$ degrés de liberté et $m = 6$ actionneurs.

Les variables de configuration q (fig.2.2) sont : $q = [q_1, q_2, q_3, q_4, q_5, q_6]^T$

Dans cette configuration, la dynamique continue est modélisée par l'équation (2.1).

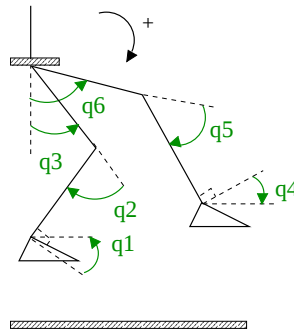
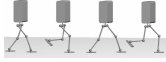


FIG. 2.2 – Modèle du robot BIP 2D suspendu



- **Pied de support fixe** : lorsque le robot évolue sur le sol, une technique classique consiste à faire l'hypothèse que le pied de support est fixé au sol. Le système comporte alors $n = 6$ degrés de liberté et $m = 6$ actionneurs.

Les variables de configuration q (fig.2.3) sont : $q = [q_1, q_2, q_3, q_4, q_5, q_6]^T$

En positionnant la base de référence au niveau du pied de support (orteils), et en faisant l'hypothèse que le transfert d'appui est instantané, on s'affranchit du calcul des contacts avec le sol. La dynamique continue est donc modélisée par l'équation (2.1). Cette approche implique de calculer un modèle pour le cas où le pied droit est le pied de support, et un modèle dans le cas où c'est le pied gauche qui assure ce rôle. Lorsque la jambe de vol entre en contact avec le sol, un impact se produit. La dynamique impulsionnelle est calculée à partir de l'équation (2.22).

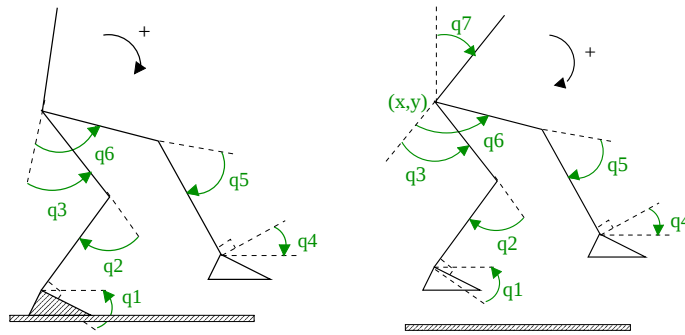
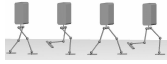


FIG. 2.3 – BIP 2D **À gauche** : modèle avec pied fixe - **À droite** : modèle avec base de référence libre

- **Base de référence libre** : dans le cas où les pieds sont libres de décoller du sol, le système comporte $n = 9$ degrés de liberté et $m = 6$ actionneurs.

Les variables de configuration q (fig.2.3) sont : $q = [q_1, q_2, q_3, q_4, q_5, q_6, q_7, x, y]^T$

Cette configuration est sous-actionnée uniquement lorsque le robot est en phase de vol. Le robot BIP n'étant pas censé courir, cela ne survient jamais : en pratique, il a toujours au moins l'un de ses pieds en contact avec le sol. Nous avons choisi le bassin comme base de référence. Le modèle utilisé pour la dynamique continue est celui représenté par l'équation (2.15) qui comprend la gestion des contacts des pieds avec le sol sous l'hypothèse de non glissement. La dynamique impulsionnelle est, quant à elle, calculée à partir de l'équation (2.22).



2.3.2 Robot BIP3D

Trois stratégies différentes sont également proposées pour modéliser le robot BIP 3D.

- **Robot suspendu** : dans les phases de réglages et d'initialisation du système, le robot n'est pas posé sur le sol. L'armoire qui sert de tronc à BIP est alors soutenue en son sommet par des sangles de sécurité. Dans cette configuration, le système comporte $n = 15$ degrés de liberté et $m = 15$ actionneurs. Les variables de configuration q sont : $q = [q_1, q_2 \dots q_{15}]^T$. La dynamique continue est modélisée par l'équation (2.1).
- **Pied de support fixe** : nous avons utilisé le robot pour des expérimentations de mouvements posturaux en simple support. Le pied de support est supposé fixé au sol et le pied de vol est constamment au dessus du sol. Le système comporte $n = 15$ degrés de liberté et $m = 15$ actionneurs. Les variables de configuration q sont : $q = [q_1, q_2 \dots q_{15}]^T$. En positionnant la base de référence au niveau de ce pied de support, on s'affranchit du calcul des contacts. La dynamique est modélisée par l'équation (2.1).
- **Base de référence libre** : dans le cas où le robot est libre d'évoluer dans l'espace, le système comporte $n = 21$ degrés de liberté et $m = 15$ actionneurs. Les variables de configuration q (fig.2.3) sont : $q = [q_1 \dots q_{15}, x, y, z, \theta_1, \theta_2, \theta_3]^T$. Nous avons choisi le bassin comme base de référence. Le modèle utilisé pour la dynamique continue est donné par l'équation (2.15) en faisant l'hypothèse de non glissement. La dynamique impulsionnelle est calculée à partir de l'équation (2.22).

2.3.3 Robot RABBIT

Une stratégie est proposée pour modéliser le robot RABBIT. La base de référence est libre et placée au bassin. Le système comporte $n = 7$ degrés de liberté et $m = 4$ actionneurs (Roussel, 1998). Les variables de configuration q (fig.2.4) sont : $q = [q_1, q_{31}, q_{41}, q_{32}, q_{42}, x, y]^T$. La dynamique continue est celle de l'équation (2.15). La dynamique impulsionnelle est calculée à partir de l'équation (2.22).

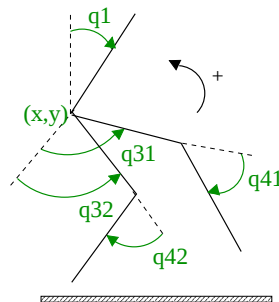
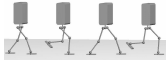
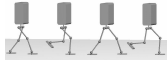


FIG. 2.4 – Modèle du robot RABBIT



2.4 Conclusion

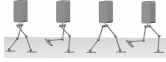
La littérature présente souvent des modèles aux hypothèses très fortes. Des simplifications sont inévitables, de plus elles peuvent être nécessaires pour des raisons de temps de calcul ou simplement correspondre au cas de figure considéré. Il est indispensable d'être conscient des conditions de validité des modèles utilisés, tout particulièrement lorsqu'il s'agit de passer de la simulation à l'expérimentation. Les modèles que nous avons retenus présentent l'avantage de tenir compte dans leur formulation : des contacts, des impacts et des décollements. Cette formulation synthétique est adaptée à la résolution numérique, comme nous le verrons dans la quatrième partie du mémoire.



Conclusion

Nous venons de présenter les prototypes réels ainsi que leurs modèles que nous allons utiliser dans la suite de ce mémoire. Nos travaux ont été axés volontairement vers l'expérimentation et la simulation. En effet, le cadre théorique, bien qu'indispensable, nécessite dans l'état d'avancement actuel des résultats, de faire des hypothèses importantes. La plate-forme BIP qui nous a essentiellement concernés s'inscrit parfaitement dans les objectifs de cette thèse du fait de sa structure et les applications envisagées : souligner les interactions entre marche humaine et marche bipède robotisée. Comme nous le verrons, la propriété anthropomorphique du robot permet certaines analyses comparatives avec la locomotion naturelle.

La première partie de ce mémoire a permis de poser le cadre de notre étude. La seconde partie a présenté les outils sur lesquels s'appuie notre travail. La suite va permettre de mettre en relief les résultats obtenus en fonction des choix considérés.



Troisième partie

Approche expérimentale de la locomotion

Sommaire

- ★ **Problématique**

expérimenter un panel d'allures de marche pour évaluer l'influence des paramètres

- ★ **Génération d'allures de marche**

description des méthodes de conception de mouvements pour les robots BIP 2D et BIP 3D

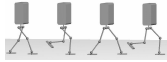
- ★ **Contrôleur pour la poursuite d'allures de marche**

expérimenter les mouvements prédéfinis avec précision et robustesse pour les deux versions du robot

- ★ **Expérimentation de marches**

analyse comparative des déplacements réalisés sur le robot BIP 2D

- ★ **Conclusion**



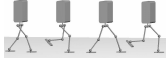
Problématique

Expérimenter un panel d'allures de marche pour évaluer l'influence des paramètres.

Les recherches en robotique mobile s'attachent essentiellement à développer l'autonomie et l'adaptabilité des applications. C'est également l'approche commune en ce qui concerne les robots bipèdes. Pourtant, nous pensons que le mode de locomotion sur deux pattes est en lui-même un vaste domaine d'investigation et qu'il mériterait plus d'intérêt de la part de la communauté. Les problèmes de stabilité et l'optimisation des marches sont des questions qui demeurent ouvertes. La mise en œuvre de méthodes de contrôle, même classiques, prouve que les mouvements obtenus sont influencés par de nombreux paramètres (conditions initiales, longueur des pas, vitesse de déplacement, hauteur de décollage du pied de vol...). Le réglage de ces paramètres n'est pas toujours évident et est souvent réalisé empiriquement.

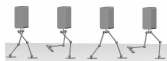
Lorsque l'on travaille sur des systèmes réels et non plus avec des modèles, de nouveaux facteurs influencent fortement les démarches : frottements, jeux et flexibilités mécaniques, limites des actionneurs, implantation de la commande, nature effective des contacts pied/sol... Ignorer ces phénomènes ou les considérer via des modèles, nécessairement simplificateurs et biaisés, ne nous semble pas viable. Une analyse pertinente de la démarche d'un robot bipède doit faire l'objet d'expérimentations précises en vue d'être évaluée et ajustée. La simulation est insuffisante car les modèles de bipèdes consensuellement utilisés ne permettent pas de prendre en compte tous les phénomènes physiques, notamment les perturbations et incertitudes.

Par ailleurs, les aspects temps réel sont essentiels et intimement liés au contrôle-commande et dépendent fortement de l'architecture matérielle et logicielle. De plus, cette architecture ainsi que tous les composants du robot influencent les dépenses énergétiques dont la maîtrise est d'un intérêt primordial du point de vue de l'autonomie du système. La littérature présente généralement une allure de marche unique (pas toujours expérimentée), sans explicitation claire du choix des paramètres de la démarche.



Notre objectif est donc de présenter une campagne d'expérimentations d'allures de marche sur BIP afin de mettre en exergue les influences des paramètres des marches et d'évaluer la précision de nos modèles.

Dans un premier temps nous présentons la technique utilisée pour générer des trajectoires de marche paramétrées. Puis, nous introduirons le contrôleur mis en œuvre pour la poursuite de ces trajectoires. Enfin, nous analyserons comparativement les résultats expérimentaux obtenus dans le cas du robot BIP 2D.



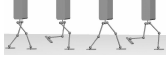
Chapitre 1

Génération d'allures de marche

1.1 Introduction

La marche bipède peut être dynamiquement ou statiquement stable, les deux aspects étant importants et permettant d'atteindre des objectifs distincts. La marche dynamique correspond à la marche humaine normale stationnaire : le déplacement étant engendré par des déséquilibres avant rattrapés. Ce type de locomotion ne s'adapte pas selon nous à un suivi précis de trajectoires de référence et nous présenterons en quatrième partie les solutions que nous proposons pour l'aborder. La marche statique est adoptée dans les situations délicates (franchissement d'obstacle, sol en pente, environnement 3D difficile, travail à un poste fixe), les déplacements correspondant à l'enchaînement de postures statiquement stables. Ce type de mouvements doit être précis et une solution est de recourir à une commande de poursuite de trajectoires préétablies. Ce type de méthode nécessite de posséder une bibliothèque d'allures de marche parmi lesquelles choisir en fonction de la situation (obstacles, pentes, escaliers, virages...). Pour être utilisables ces mouvements doivent être paramétrés (longueur de pas, hauteur de pas...). Nous nous sommes intéressés au développement d'outils facilitant la génération de telles allures de marche.

Un simulateur a été réalisé par P.-B. Wieber (Wieber, 2000) au cours de sa thèse au sein du projet BIP de l'INRIA : ROBOTSIMU. Ce simulateur a intégré successivement le modèle du robot BIP 3D avec pied d'appui fixe puis le modèle du robot BIP 3D avec base libre. ROBOTSIMU comporte, outre la simulation de la dynamique, une partie génération de mouvements. Le principe est de définir des postures clefs du robot. Ces positions sont solutions d'un problème d'optimisation qui minimise l'énergie nécessaire pour maintenir la posture (compensation du vecteur gravité) tout en se rapprochant de l'allure générale définie par le concepteur (position du pied de la jambe libre, position du centre de masse...). Une fois ces postures clefs obtenues, une interpolation polynomiale entre ces positions de passage permet d'obtenir des trajectoires continues. Sur la base de ce travail, nous nous sommes attachés à développer des outils permettant de définir aisément des allures de marche statiques paramétrées pour le robot BIP dans sa version 2D dans un premier temps, puis dans sa version 3D.



1.2 Robot BIP 2D

Nous présentons dans cette partie l'outil de génération d'allures de marche qui a été développé pour la version 2D de BIP (Azevedo et al., 2002a ; Azevedo et Andreff, 2000). Nous avons utilisé les modèles du robot suspendu, et en simples appuis droit et gauche. Les trajectoires présentées correspondent à des marches statiquement stables, symétriques dans le plan sagittal. Ces marches n'ont pas la prétention d'être anthropomorphes ou optimales.

1.2.1 Fonction de sortie

Les trajectoires étant difficiles à définir dans l'espace de configuration (q_d) (fig.1.1, page suivante), nous avons défini un espace de sortie dans lequel la manipulation des 6 degrés de liberté du robot est plus intuitive. On dispose d'une *fonction de sortie* paramétrée par la géométrie du robot, comme suit :

$$f = \begin{pmatrix} x_{cdm} & y_{han} & \theta_t & x_{che\ vol} & y_{che\ vol} & \theta_{sem\ vol} \end{pmatrix}^T \quad (1.1)$$

où : x_{cdm} est la position de la projection du centre de masse dans la direction de la marche, y_{han} l'altitude de la hanche, θ_t l'orientation du tronc, $(x_{che\ vol}, y_{che\ vol})$ les coordonnées cartésiennes de la cheville libre (vol), $\theta_{sem\ vol}$ l'orientation de la semelle de la jambe de balancement (fig.1.2).

Note : l'origine du repère cartésien est placée au niveau des orteils du pied d'appui.

1.2.2 Marche symétrique statique

Il s'agit de définir l'évolution au cours du temps de la fonction de sortie :

$$\forall t_0 \leq t \leq t_f$$

$$f(t) = \begin{pmatrix} x_{cdm}(t) & y_{han}(t) & \theta_t(t) & x_{che\ vol}(t) & y_{che\ vol}(t) & \theta_{sem\ vol}(t) \end{pmatrix}^T$$

où : t_0 (resp. t_f) est le temps d'initiation (resp. de terminaison) du pas.

Les valeurs doivent garantir la continuité dans le temps des éléments de f et de leur dérivée, l'évitement du sol, et la stabilité statique :

- *Stabilité statique* : afin de préserver la stabilité statique, on assure que la projection du centre de masse demeure constamment à l'intérieur du polygone de sustentation. En simple appui, la projection du centre de masse (x_{cdm}) doit donc se trouver entre le talon et les orteils du pied d'appui. Dans le cas du double appui, x_{cdm} doit se trouver entre le talon le plus en arrière et les orteils les plus en avant.

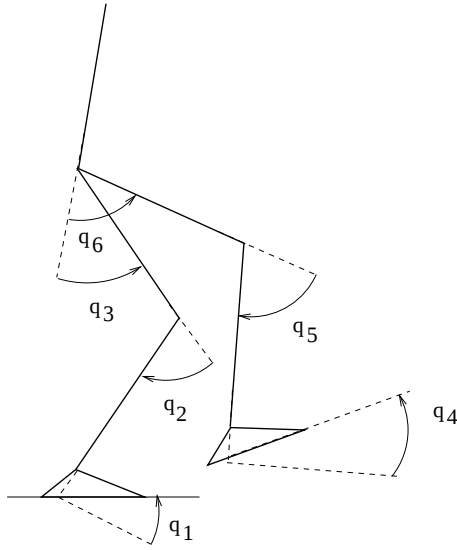
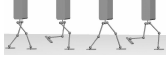


FIG. 1.1 – Espace de configuration

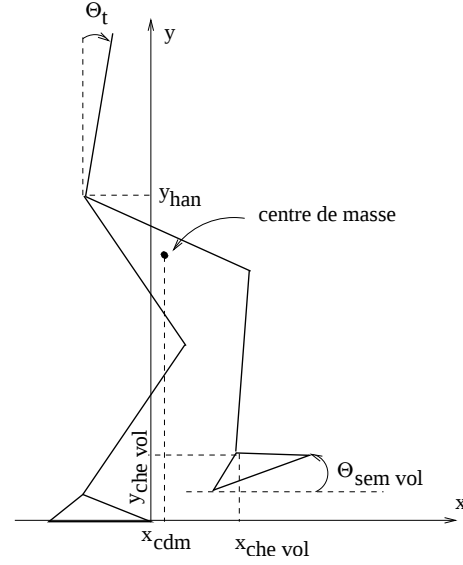


FIG. 1.2 – Espace de sortie

- *Gestion du double appui* : dans un souci de simplification, nous n'autorisons pas de mouvement lorsque le robot a ses deux pieds en contact avec le sol. Ainsi, le simple appui droit doit être immédiatement suivi par le simple appui gauche et réciproquement.
- *Enjambée* : du fait des contraintes précédentes, à chaque fin de pas, la projection du centre de masse (x_{cdm}) doit se trouver entre le talon et les orteils de chacun des pieds. Cela limite la longueur des pas (l_{pas}) à un maximum égal à deux fois la longueur du pied (l_p). Dans un souci de robustesse expérimentale, nous avons défini une marge de sécurité par rapport aux talons (x_{tal}) et une marge par rapport aux orteils (x_{ort}). Cela réduit l'intervalle autorisé pour les positions de la projection du centre de masse. Les deux pieds sont donc contraints de se recouvrir (dans le plan sagittal) d'une longueur minimale $x_{tal} + x_{ort}$.

On a :

$$l_{pas} = 2(l_p - x_{tal} - x_{ort})$$

On note x_{med} la position médiane de x_{cdm} :

$$x_{med} = \frac{-l_p + x_{tal} - x_{ort}}{2} = x_{ort} - \frac{l_{pas}}{4}$$

D'où :

$$\begin{cases} x_{tal} = l_p - x_{med} - \frac{3l_{pas}}{4} \\ x_{ort} = x_{med} + \frac{l_{pas}}{4} \end{cases}$$

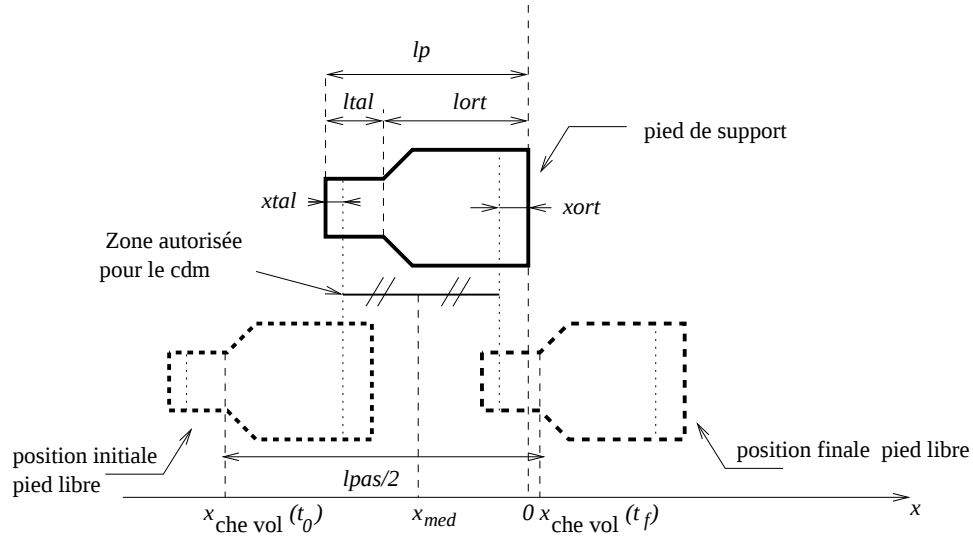
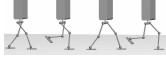


FIG. 1.3 – Contraintes sur les positions des pieds et de la projection du centre de masse

Une des conséquences du recouvrement minimal des pieds est que la projection du centre de masse doit se trouver en début de pas ($t = t_0$), à une distance x_{tal} du talon d'appui et en fin de pas ($t = t_f$) à une distance x_{ort} des orteils d'appui :

$$\begin{aligned} x_{cdm}(t_0) &= -l_p + x_{tal} \\ x_{cdm}(t_f) &= -x_{ort} \end{aligned}$$

Le recouvrement minimal contraint également les positions initiale et finale de la cheville libre :

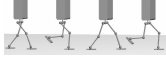
$$\begin{aligned} x_{che\ vol}(t_0) &= -l_p + x_{tal} + x_{ort} - l_{tal} \\ x_{che\ vol}(t_f) &= l_{ort} - x_{tal} - x_{ort} \end{aligned}$$

où l_{ort} (resp. l_{tal}) est la distance entre la cheville et l'avant (resp. l'arrière) du pied.

- *Cote de la cheville libre* : nous avons choisi de maintenir horizontal le pied libre ($\theta_{sem\ vol}(t) = 0 \text{ rad } \forall t$). Le pied libre ne doit pas heurter le sol en cours de pas, la hauteur de la cheville libre $y_{che\ vol}$ est donc maintenue au dessus de la hauteur de la cheville du pied d'appui (h_{che}) :

$$\begin{aligned} y_{che\ vol}(t_0) &= y_{che\ vol}(t_f) = h_{che} \\ y_{che\ vol}(t) &> h_{che}, \forall t_0 < t < t_f \end{aligned}$$

- *Orientation du tronc et hauteur de la hanche* : nous avons imposé dans un souci d'esthétique que le tronc demeure vertical ($\theta_t(t) = 0 \forall t$). La hauteur de la



hanche est laissée libre en cours de pas, mais afin de satisfaire la condition de symétrie les hauteurs en début et fin de pas doivent être égales :

$$y_{han}(t_0) = y_{han}(t_f).$$

- Enfin, une contrainte est donnée au temps intermédiaire $t_m = (t_0 + t_f)/2$ afin de permettre au pied de décoller du sol. La fonction de sortie (1.1) peut donc s'écrire :

$$f(t_m) = \left(\frac{x_{cdm}(t_0) + x_{cdm}(t_f)}{2} \quad y_{han}(t_m) \quad 0 \quad \frac{x_{che\ vol}(t_0) + x_{che\ vol}(t_f)}{2} \quad y_{che\ vol}(t_m) \quad 0 \right)^T$$

où : $y_{han}(t_m)$ permet de fixer l'élévation du bassin au cours du pas et $y_{che\ vol}(t_m)$ détermine la hauteur maximale de l'enjambée.

- *Récapitulatif* : la fonction de sortie doit donc satisfaire :

$$\begin{aligned} \forall t_0 < t < t_f \\ f(t) &= \begin{pmatrix} x_{cdm}(t) & y_{han}(t) & 0 & x_{che\ vol}(t) & y_{che\ vol}(t) & 0 \end{pmatrix}^T \\ f(t_0) &= \begin{pmatrix} -l_p + x_{tal} & y_{han}(t_0) & 0 & -l_p + x_{tal} + x_{ort} - l_{tal} & h_{che} & 0 \end{pmatrix}^T \\ f(t_m) &= \begin{pmatrix} \frac{x_{cdm}(t_0) + x_{cdm}(t_f)}{2} & y_{han}(t_m) & 0 & \frac{x_{che\ vol}(t_0) + x_{che\ vol}(t_f)}{2} & y_{che\ vol}(t_m) & 0 \end{pmatrix}^T \\ f(t_f) &= \begin{pmatrix} -x_{ort} & y_{han}(t_0) & 0 & l_{ort} - x_{tal} - x_{ort} & h_{che} & 0 \end{pmatrix}^T \\ x_{tal} &= l_p - x_{med} - \frac{3l_{pas}}{4} \\ x_{ort} &= x_{med} + \frac{l_{pas}}{4} \end{aligned}$$

1.2.3 Trajectoires polynomiales

Une fois les contraintes sur la fonction de sortie exprimées, on peut passer à la génération du mouvement. Une manière de le faire est d'interpoler de façon polynomiale entre des positions intermédiaires dans l'espace de sortie :

$$f(t) = \sum_{i=0}^n A_i t^i$$

Dans le cas de la marche sur sol plat, si le pied se pose sans impact sur le sol, des polynômes de degré 4 sont suffisants. Pour éviter les impacts, on impose une vitesse nulle dans l'espace de sortie :

$$\frac{df}{dt}(t_f) = 0$$

Pour assurer la continuité de la vitesse lors du changement d'appui, la vitesse initiale est elle aussi égale à zéro :

$$\frac{df}{dt}(t_0) = 0$$

Ainsi, les trajectoires dans l'espace de sortie sont obtenues à l'aide d'un polynôme d'ordre 4 entre des positions initiale, intermédiaire et finale. Elles assurent la

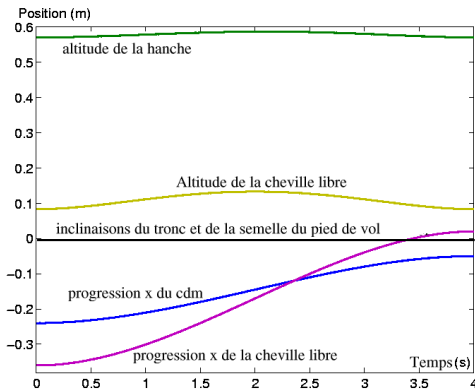
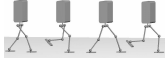


FIG. 1.4 – Évolution de la fonction de sortie sur 1 pas

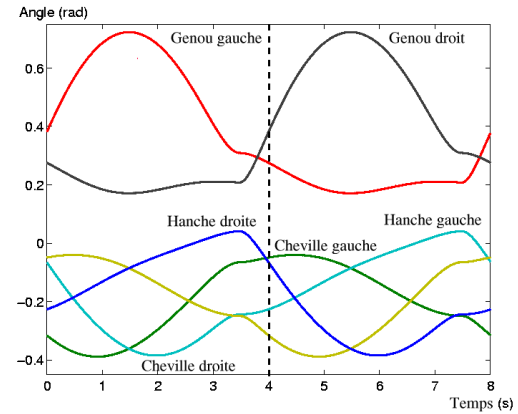


FIG. 1.5 – Inversion de la fonction de sortie puis symétrisation pour obtenir les trajectoires articulaires sur 2 pas

stabilité statique sans mouvement pendant le double appui, et sans impact du pied libre sur le sol. Un exemple de trajectoire est donné figure 1.4. Ces trajectoires sont ensuite traduites en trajectoires articulaires en inversant la fonction de sortie (voir fig.1.5 et fig.1.6). Différentes allures de marche ont ainsi été générées et expérimentées, les résultats sont présentés dans le dernier chapitre de cette partie du mémoire.

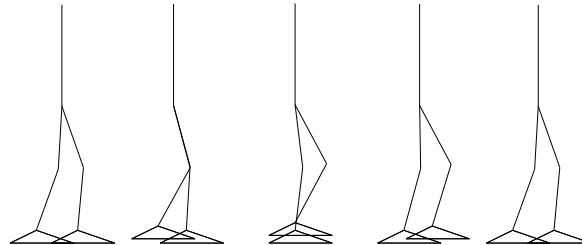


FIG. 1.6 – Exemple d'allure de marche générée pour BIP 2D

1.3 Robot BIP 3D

La définition de marches dans l'espace 3D pose de nombreux problèmes liés en particulier à la complexité du modèle cinématique. Nous proposons une méthode pour la génération d'allures paramétrées de manière intuitive et simple à partir de l'optimisation sous contraintes d'une fonction quadratique des couples. Ce travail a été réalisé en collaboration avec F. Lydoire dans le cadre de son stage de DEA (Lydoire et al., 2002 ; Lydoire, 2001).

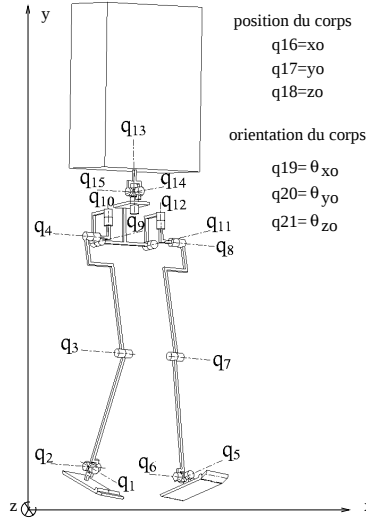
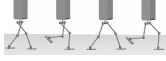


FIG. 1.7 – Espace de configuration

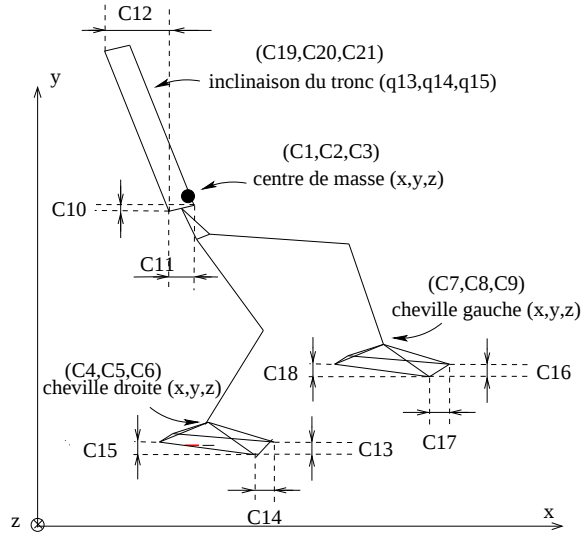


FIG. 1.8 – Espace de sortie

1.3.1 Fonction de sortie

Le robot BIP dans sa version complète est composé de 15 articulations actionnées. Il peut se déplacer librement dans l'espace et comporte ainsi 21 degrés de liberté. Le vecteur de configuration $\mathbf{q}$ est composé d'une part des positions articulaires, et d'autre part de la position et de l'orientation du robot dans l'espace (fig.1.8). Nous avons défini un espace de sortie dans lequel la manipulation des 21 degrés de liberté du robot est plus intuitive (fig.1.8). Le vecteur $\mathbf{C}$ contient les positions cartésiennes de points particuliers et l'orientation de certains segments du robot.

1.3.2 Optimisation sous contraintes

Une marche statiquement stable peut être vue comme une succession de postures statiques, chacune stable. Nous avons posé le problème de génération de postures sous la forme d'un problème d'optimisation sous contraintes classique dans lequel il faut résoudre :

$$\min_{\mathbf{q}} J(\mathbf{q}) = \frac{1}{2} \mathbf{U}(\mathbf{q})^T \mathbf{U}(\mathbf{q})$$

$$\text{soumis à : } \mathbf{q}^{\min} \leq \mathbf{q} \leq \mathbf{q}^{\max}$$

$$g_j(\mathbf{C}) \leq 0 \quad j = 1 \dots n_g$$

$$h_j(\mathbf{C}) = 0 \quad j = 1 \dots n_h$$

- le critère J est une fonction quadratique des tensions appliquées aux moteurs pour maintenir la posture, c'est-à-dire, pour compenser la gravité ;
- $\mathbf{U}(\mathbf{q})$ est le vecteur de dimension 15 des tensions nécessaires aux moteurs pour conserver la position $\mathbf{q}$;

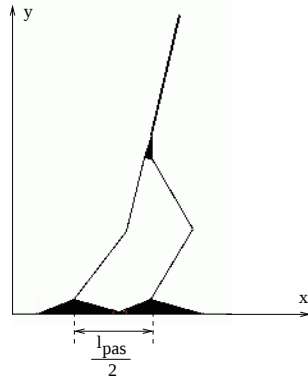
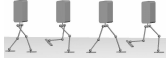


FIG. 1.9 – Posture initiale

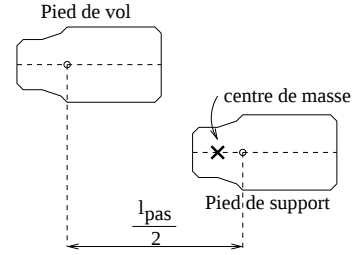


FIG. 1.10 – Au démarrage du pas, la projection du cdm se trouve sur la semelle du pied d'appui, sur la ligne de symétrie du pied

- q^{min} et q^{max} sont les vecteurs des butées mécaniques des articulations ;
- g regroupe les contraintes inégalités ;
- h regroupe les contraintes égalités.

La résolution de notre problème est basée sur un algorithme d'optimisation FSQP¹ sous contraintes non linéaires (fsqp (Craig et al., 1997)).

1.3.3 Posture initiale

La première étape de la conception d'une marche consiste à définir la posture de démarrage du pas. La marche étant symétrique, les configurations en début et en fin de pas sont symétriques. La posture initiale définit donc la longueur du pas (fig.1.9). Cette posture est générée en fixant des degrés de liberté dans l'espace de sortie C , c'est-à-dire en définissant des contraintes égalités h pour le problème d'optimisation :

$$h_j = C_j - C_j^d \quad j \in [1, \dots, N]$$

où : C_j^d sont les valeurs désirées et C_j les valeurs effectives.

Les valeurs C_j^d traduisent les contraintes suivantes :

- les deux pieds sont posés à plat sur le sol,
- le pied libre est placé à une distance qui correspond à la demi-longueur de pas souhaitée en arrière du pied d'appui,
- la projection du cdm se trouve sur la semelle du pied d'appui, proche du talon et sur la ligne de symétrie du pied (fig.1.10).

Les degrés de liberté restant peuvent être librement définis pour satisfaire à des critères esthétiques, par exemple.

¹FSQP : Feasible Sequential Quadratic Programming

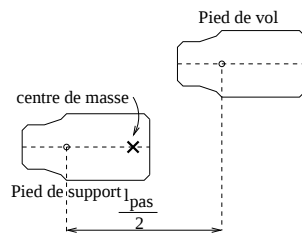
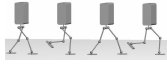


FIG. 1.11 – En fin de simple appui, la projection du cdm se trouve sur la semelle du pied d'appui, proche des orteils et sur la ligne de symétrie du pied.

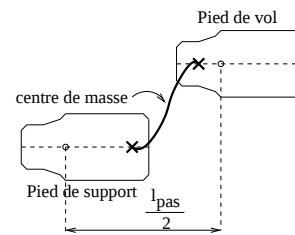


FIG. 1.12 – En double appui, la projection du cdm se déplace selon une trajectoire polynomiale des orteils du pied d'appui jusqu'au talon du pied libre.

1.3.4 Génération du mouvement

Partant de la posture initiale, une série de postures est générée successivement à partir des postures précédentes. Cette séquence définit le mouvement discret du robot. On distingue les phases de simple et de double appui.

Phase de simple appui

La phase de simple appui doit permettre d'amener le robot de sa configuration de départ dans une configuration où les pieds se retrouvent dans une situation symétrique. La projection du cdm doit toujours se trouver sur le pied d'appui, mais proche des orteils (fig.1.11). Pendant la phase de simple appui :

- le pied d'appui est fixe,
- le pied libre suit une parabole dans le plan sagittal,
- le pied libre est maintenu horizontal,
- la projection du cdm se déplace sur la ligne de symétrie du pied d'appui.

En contraignant la projection du cdm à demeurer sur la ligne de symétrie du pied et entre le talon et les orteils, on s'assure constamment de la stabilité statique du robot. La parabole suivie par le pied libre permet de gérer simultanément la longueur et la hauteur des pas. La vitesse de progression du pied libre en avant est définie en indiquant la distance à parcourir entre chaque posture. La vitesse de progression du centre de masse en avant est liée à la vitesse de progression du pied libre en fonction du rapport entre la distance à parcourir par le cdm et la longueur du pas.

Phase de double appui

La phase de double appui doit permettre d'amener le robot dans une configuration symétrique de celle de la posture initiale. Les pieds étant déjà positionnés correctement à la fin de la phase de simple appui, il s'agit essentiellement de ramener la

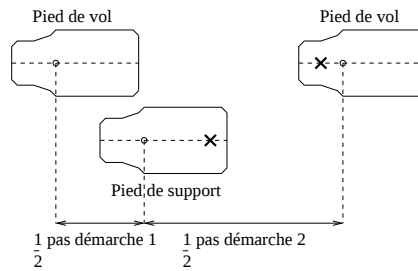
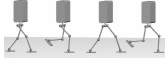


FIG. 1.13 – Pas de transition entre deux marches

projection du cdm du pied d'appui du pas précédent sur le pied libre qui deviendra pied d'appui du pas suivant, à l'issue du double appui. Pendant le double appui :

- le pied d'appui est fixe,
- le pied libre est fixe,
- la projection du cdm se déplace en suivant une trajectoire polynomiale depuis les orteils du pied d'appui jusqu'au talon du pied libre (fig.1.12).

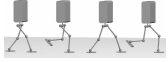
La vitesse de progression du cdm en avant est liée directement à la discrétisation de la fonction polynomiale. La posture finale étant une posture optimisée pour une projection du cdm en double appui, lorsque le cdm atteint la position finale, le reste du corps se retrouve nécessairement dans la posture souhaitée.

Le pas de transition

Il ne serait guère intéressant de disposer de différentes allures de marches sans pouvoir passer de l'une à l'autre. En effet, lorsque le robot se déplace et qu'il rencontre un obstacle, il doit être capable d'enchaîner la marche en cours avec une marche qui lui permettra, par exemple, de passer par dessus l'obstacle. Pour ce faire, nous avons prévu des pas de transition entre chacune de nos marches. Ce pas est généré sur le même principe qu'un pas "normal", à la différence qu'il n'est pas symétrique. La posture initiale correspond à un simple appui de la marche précédente et la posture finale à atteindre correspond au simple appui du pied opposé de la marche suivante.

Résultats

Nous avons pu générer différentes marches. Il est également possible de passer de l'une à l'autre de ces allures par un pas de transition. Nous donnons ici un exemple où la première marche correspond à des pas de longueur 15 cm et de hauteur 5 cm (fig.1.14) qui s'enchaîne (fig.1.16) avec une allure de pas de longueur 50 cm et de hauteur 10 cm (fig.1.15).



1.4 Conclusion

Les méthodes présentées permettent de définir de façon simple et intuitive les allures de marche souhaitées en spécifiant les paramètres d'un pas. La génération des marches est rapide et garantit la stabilité statique et la faisabilité des mouvements en termes de limites articulaires et de saturations des actionneurs. La définition des marches est réalisée dans un espace de sortie, le résultat obtenu correspond à des trajectoires dans l'espace articulaire. Le succès de l'expérimentation repose sur le contrôleur dont la mission est de poursuivre les mouvements désirés pour le robot de manière la plus précise possible.

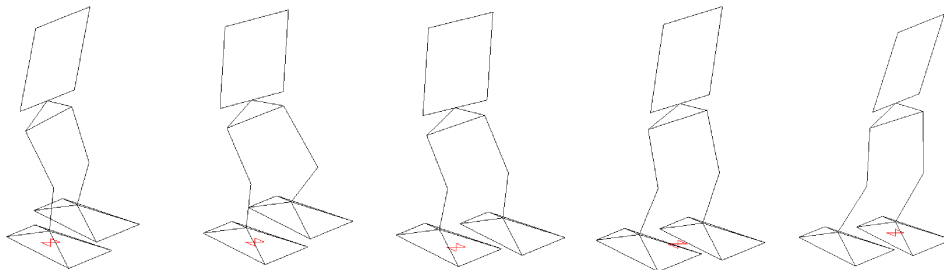


FIG. 1.14 – Marche D_1 : longueur du pas 15 cm, hauteur 5 cm

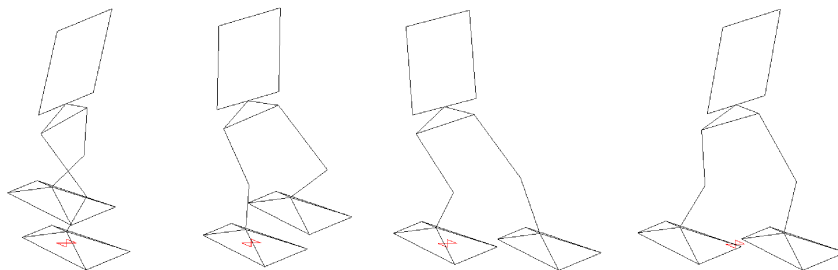


FIG. 1.15 – Marche D_2 : longueur du pas 50 cm, hauteur 10 cm

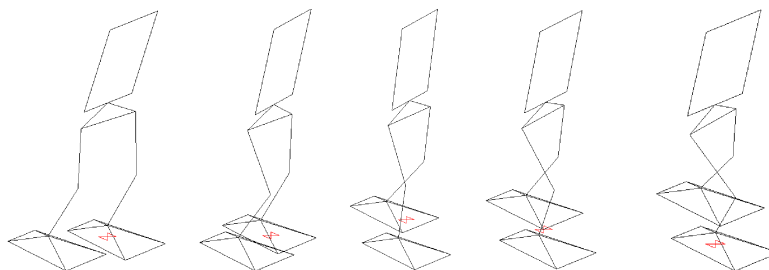
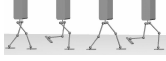


FIG. 1.16 – Pas de transition entre les marches D_1 et D_2

Note : la croix représentée sur les schémas correspond à la projection du centre de masse



Chapitre 2

Contrôleur pour la poursuite d'allures de marche

2.1 Introduction

Le chapitre précédent présente la méthodologie utilisée pour concevoir des mouvements de marche statique pour le robot BIP. Ces mouvements sont exprimés sous la forme de trajectoires articulaires discrétisées. Nous avons développé, implanté et validé expérimentalement un contrôleur permettant un suivi robuste et efficace des mouvements pré-calculés pour le robot BIP dans ses configurations 2D et 3D (Azevedo et Pissard-Gibollet, 2000 ; Azevedo et al., 2000). Dans un premier temps, nous présentons la loi de commande utilisée pour la poursuite des allures de marche, ensuite nous décrivons les réglages des paramètres de la commande. L'implantation du contrôleur est alors détaillée, suivie d'un exemple d'application sur le robot BIP 3D.

2.2 Loi de commande

L'objectif de la loi de commande proposée est de poursuivre des trajectoires $q_d(t)$ calculées hors ligne. Nous utilisons une commande bas niveau qui nous permet de respecter les contraintes de période d'échantillonnage réduite sous une puissance de calcul limitée.

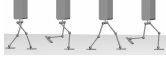
Le modèle dynamique utilisé est celui présenté dans la deuxième partie du mémoire page 48 et il s'écrit sous la forme lagrangienne suivante :

$$M(q)\ddot{q} + N(q, \dot{q})\dot{q} + G(q) = \Gamma$$

Les allures de marche que nous considérons sont statiques, elles correspondent à l'enchaînement de postures statiquement stables. Par définition, en équilibre postural les termes d'inertie $M(q)\ddot{q}$ et des effets de Coriolis et centrifuges $N(q, \dot{q})\dot{q}$ sont négligeables devant les effets de la gravité $G(q)$.

Nous avons donc choisi de calculer les couples articulaires $\Gamma(t)$ du robot à l'aide d'un correcteur PD complété par une compensation du vecteur gravité :

$$\Gamma(t) = \hat{K}_p(q_d(t) - q(t)) + \hat{K}_v(\dot{q}_d(t) - \dot{q}(t)) + \hat{G}(q(t)) \quad (2.1)$$



où :

- $\hat{K}_p$ et $\hat{K}_v$ sont des constantes ;
- $\hat{G}$ le vecteur gravité ;
- $q_d, \dot{q}_d$ sont respectivement les positions et vitesses articulaires souhaitées ;
- $q, \dot{q}$ les positions et vitesses articulaires réelles.

Les actionneurs sont commandés en tension U . On introduit une compensation des frottements secs F au niveau des moteurs suivant le modèle de Armstrong (Armstrong-Hélouvy, 1991) :

$$U = U' + F.sign(\dot{\Theta}_m) \quad (2.2)$$

où : $\dot{\Theta}_m$ sont les consignes de vitesses de rotation des moteurs et U' les tensions obtenues à partir des couples moteur Γ_m :

$$U' = K.\Gamma_m \quad (2.3)$$

où : K est la matrice diagonale contenant les constantes électriques des moteurs.

Les couples moteurs Γ_m s'obtiennent à partir des couples articulaires Γ (2.1) :

$$\Gamma_m = J^T(q)\Gamma \quad (2.4)$$

où : J est la matrice contenant les rapports de réduction des transmissions, qui, nous l'avons vu (page 42), sont dépendants des configurations dans le cas des transmissions de type vis à rouleaux satellites (Wieber, 2000).

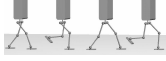
2.3 Réglage des paramètres

• Gains du correcteur

Nous avons ajusté empiriquement les gains $\hat{K}_p$ et $\hat{K}_v$. Cette tâche est épineuse et délicate étant donné la complexité du robot et la spécificité des transmissions. Les articulations actionnées par deux moteurs en parallèle (tronc et cheville) sont tout particulièrement difficiles à contrôler. Deux jeux de gains ont été déterminés pour le robot posé sur le sol et pour le robot suspendu :

$$\hat{K} = \begin{pmatrix} \hat{K}_p \\ \hat{K}_v \end{pmatrix} = \begin{cases} K_{sol} & \text{si robot au sol} \\ K_{vol} & \text{si robot suspendu} \end{cases} \quad (2.5)$$

Lors du poser du robot dans sa position $q_d(t_0)$ sur le sol on utilise une combinaison linéaire des 2 jeux de gains dont les coefficients sont fonction du poids supporté par les pieds en appui.



- *Vecteur gravité*

On dispose de trois modèles dynamiques en fonction de la phase de support :

$$\begin{cases} M_{SD}(q) \cdot \ddot{q} + N_{SD}(q, \dot{q}) \cdot \dot{q} + G_{SD}(q) = \Gamma \\ M_{SG}(q) \cdot \ddot{q} + N_{SG}(q, \dot{q}) \cdot \dot{q} + G_{SG}(q) = \Gamma \\ M_V(q) \cdot \ddot{q} + N_V(q, \dot{q}) \cdot \dot{q} + G_V(q) = \Gamma \end{cases} \quad (2.6)$$

où : l'indice SD est utilisé pour simple support droit, SG pour simple support gauche et V pour le cas du vol (robot suspendu).

Le vecteur gravité utilisé dans la partie compensation de la gravité de la commande se calcule en fonction de la phase de support détectée par les capteurs d'effort. Bien que théoriquement les trajectoires ne présentent pas de phases de double support, les tests réels sur le robot ont montré que cela n'était pas toujours le cas. Pour gérer le double support, nous utilisons une combinaison linéaire des modèles simple support qui est fonction du poids supporté par chaque pied :

$$\hat{G}(q) = \begin{cases} G_{SD}(q) & \text{si simple support droit} \\ G_{SG}(q) & \text{si simple support gauche} \\ G_V(q) & \text{si robot suspendu} \\ \tau G_{SD} + (1 - \tau) G_{SG} & \text{si double support} \end{cases} \quad (2.7)$$

où : τ est le rapport entre le poids supporté par le pied droit et le poids total du robot mesurés en ligne à partir des capteurs d'effort.

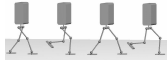
Nous verrons que cette solution simple convient tout à fait à l'évolution des forces de support et au passage du poids d'un pied sur l'autre lors du changement de support.

- *Frottements secs*

Nous avons estimé les frottements secs F expérimentalement. On peut en effet considérer que les frottements sont équivalents à la tension nécessaire pour créer le mouvement de l'articulation concernée lorsque celle-ci est immobile.

2.4 Implantation du contrôleur

L'implantation des algorithmes de commande sur la plate-forme robotique est faite avec le logiciel ORCCAD (§.1.2.4 page 44). À titre d'illustration de l'utilisation de cet environnement, nous détaillons ci-après les divers éléments de la commande implantée.



2.4.1 Tâche Robot ORCCAD

La loi de commande définie précédemment est décomposée en modules de la même manière qu'on l'aurait fait pour des fonctions en langage C : chaque module a ses entrées et sorties propres et réalise un calcul particulier qui pourra être réutilisé par d'autres modules ou par une autre loi de commande. La figure 2.1 page suivante présente l'allure générale de la Tâche Robot qui correspond à la loi de commande utilisée sur BIP pour le suivi de trajectoires.

Le module ressource physique contrôlé **BIP** correspond à une interface avec le robot. Un premier port de sortie est associé aux valeurs délivrées par les codeurs et le second port de sortie correspond aux valeurs fournies par les potentiomètres. Le port d'entrée est associé aux tensions de commande U envoyées aux moteurs du robot.

Le module ressource physique non contrôlé **sensors** correspond à une interface avec les capteurs du robot. Ce module ne comporte que des ports de sortie qui sont associés aux différents capteurs extéroceptifs du robot (capteurs d'effort, ultrason, vision...).

Les conversions entre positions articulaires et positions moteurs, couples articulaires et tensions moteurs se font par l'intermédiaire de deux modules algorithmiques : **transmission** et **conversion**. Ces transformations ne sont pas immédiates du fait des caractéristiques des transmissions du robot aux rapports de réduction variables (page 42).

Le module algorithmique **support** détermine la phase de support courante à partir des informations des capteurs d'effort reçues du module **sensors**.

Les impacts sont déterminés à partir d'une méthode de détection de rupture classique basée sur l'analyse séquentielle du signal (Basseville et Nikiforov, 1993). Nous disposons d'une fonction de détection F_d qui correspond à la somme cumulée de la force de pression sur le pied considéré, corrigée du bruit moyen. Lorsqu'un impact est détecté, on calcule la valeur τ (§.2.7).

Les vecteurs des positions et vitesses articulaires désirées à chaque instant sont fournis par le module **trajectoires** par lecture du fichier contenant les trajectoires pré-calculées hors ligne.

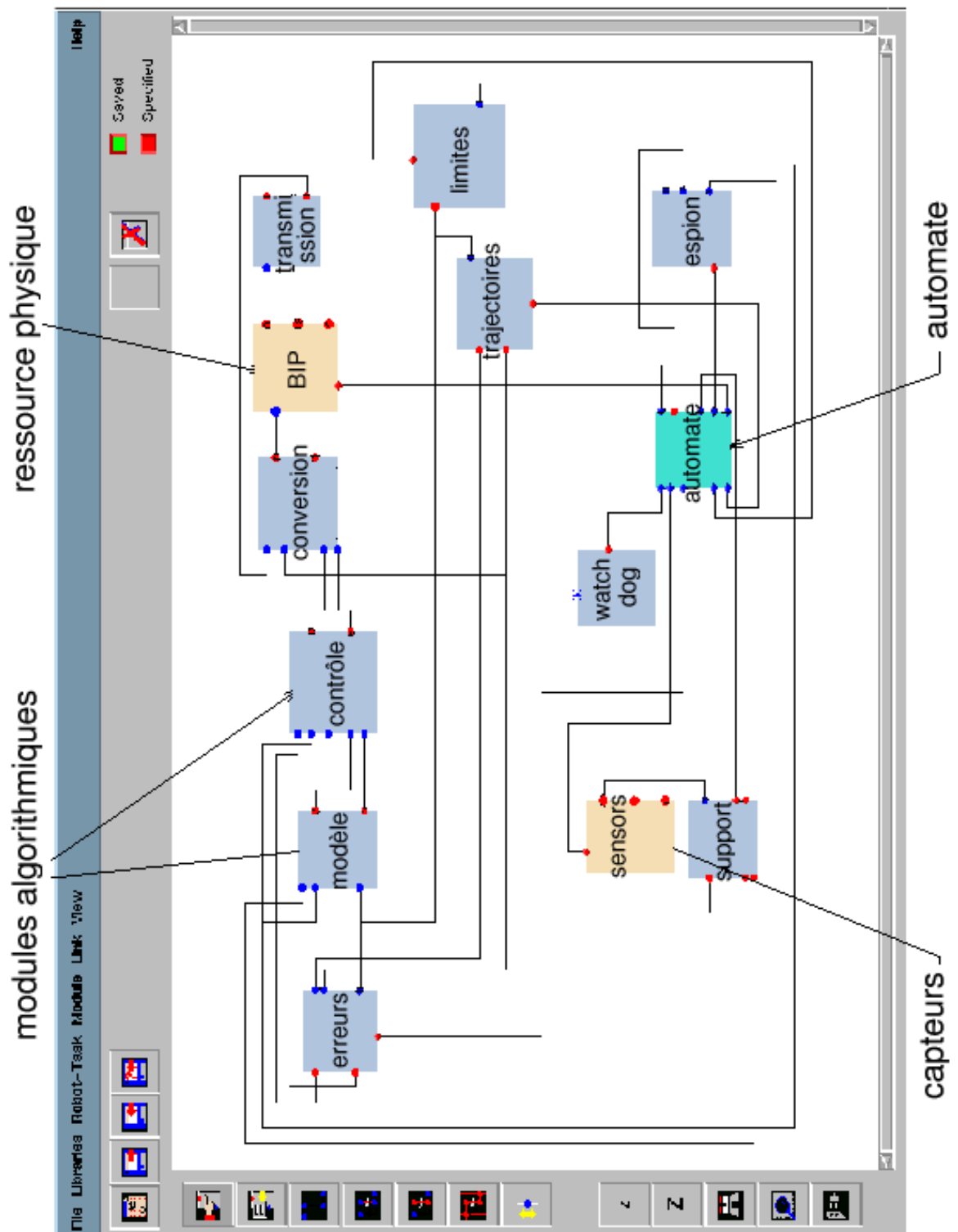
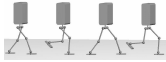


FIG. 2.1 – Tâche Robot ORCAD pour le suivi de trajectoires sur BIP

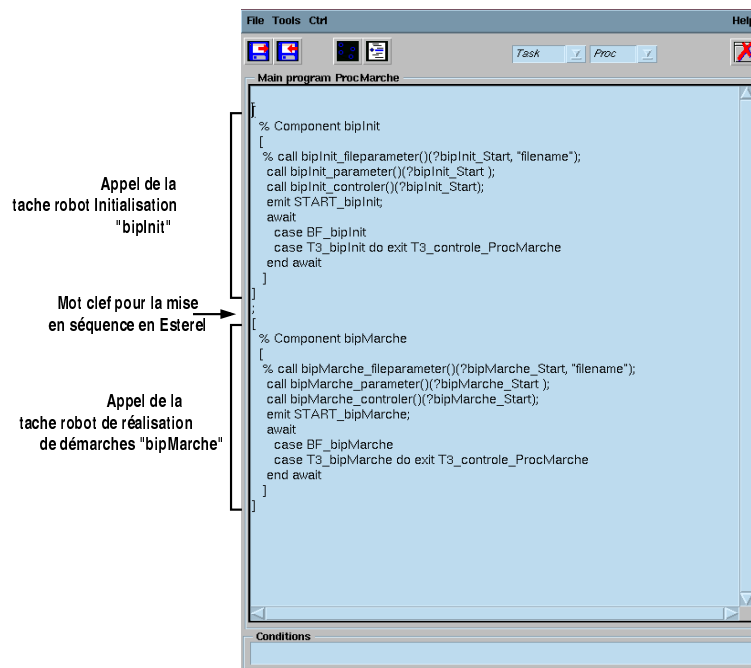
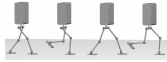


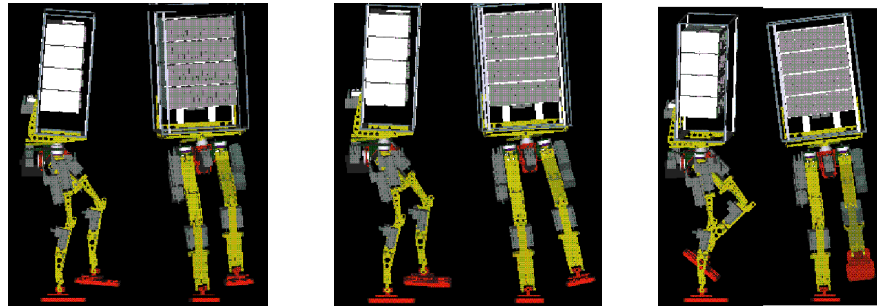
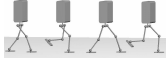
FIG. 2.2 – Procédure Robot pour la mission exécution de mouvements

Le suivi de trajectoires présenté §.2.2 est réalisé grâce à trois modules. Le module **erreurs** calcule les erreurs entre positions et vitesses désirées et réelles. Le module **modèle** détermine le vecteur gravité à utiliser en fonction du support détecté. Le module **contrôle** utilise les erreurs et le vecteur gravité déterminés par les deux modules précédents pour calculer la commande en couple articulaire.

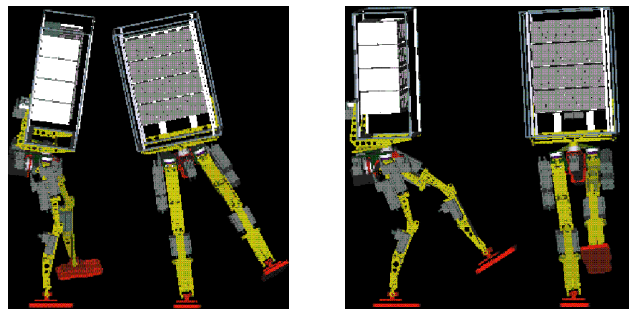
Le comportement logique est représenté par le module **automate** qui gère les différents événements générés par l'ensemble des modules et induit une action en fonction du type de ces signaux (fin normale de la Tâche Robot, arrêt d'urgence...). Le code généré étant monocadence, les périodes d'exécution des différents modules ont été fixées de façon identique. La tâche robot présente une période d'exécution de 8ms.

2.4.2 Procédure Robot ORCCAD

L'application principale correspond à la séquence d'une tâche robot qui initialise le système robotique et de la tâche robot, présentée ci-dessus, réalisant le contrôle de mouvements. La Procédure Robot est écrite dans le langage ESTEREL (Berry et Gonthier, 1992) (fig.2.2).



P0 : position initiale P1 : petit écart de côté P4 : flamant rose



P2 : grand écart de côté P3 : pied en avant

FIG. 2.3 – Mouvements réalisés par le robot BIP 15 ddl

2.5 Application au suivi de mouvements 3D

Le contrôleur a été utilisé pour une application de suivi de trajectoires sur le robot BIP 3D. Nous avons utilisé des modèles du robot suspendu, et en simple support droit (page 56). Des mouvements statiquement stables ont été définis et validés en simulation¹ à partir du simulateur réalisé par P.-B. Wieber (ROBOTSIMU). Cinq postures clefs du robot en appui sur son pied droit ont été définies (fig.2.3). Les trajectoires exécutées correspondent à des mouvements d'une posture à l'autre.

L'ensemble de ces mouvements a été expérimenté avec succès sur le robot (fig.2.4). Il faut souligner qu'au moment où ont été produits ces résultats, aucun autre robot dans le monde n'était à notre connaissance capable d'exécuter de tels mouvements de grande amplitude en équilibre sur un pied.

¹Je tiens à remercier P.-B. Wieber, B. Espiau et N. Cisló de m'avoir fourni moult trajectoires plus acrobatiques les unes que les autres !

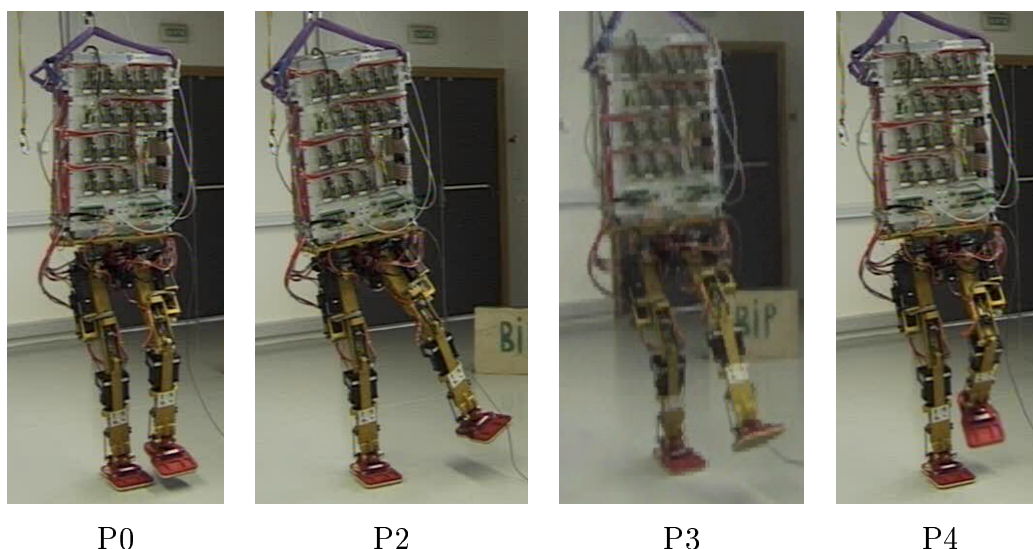
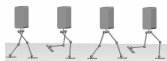


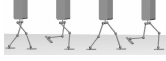
FIG. 2.4 – Images d'un film réalisé lors d'une manipulation

2.6 Application au suivi de mouvements 2D

Le contrôleur a également été utilisé pour du suivi de trajectoires de marche sur le robot BIP 2D. Nous avons utilisé des modèles du robot suspendu, en simples supports droit et gauche (page 55). Des mouvements de locomotion statiquement stables ont été définis et validés en simulation à partir du simulateur que nous avons développé et présenté dans le chapitre précédent. Cette campagne expérimentale fait l'objet du chapitre suivant.

2.7 Conclusion

Différentes trajectoires ont été générées et ont pu être directement rejouées sur les versions 2D et 3D du robot BIP. Elles n'ont nécessité ni ajustement ni modification. Les positions des centres de pression simulés et mesurés sont identiques à quelques millimètres près. Nous avons ainsi pu valider la modélisation des robots. La robustesse du contrôleur aux perturbations nous a permis de réaliser une campagne d'expérimentations sur la marche sagittale statique du robot dans sa version 2D, cette étude fait l'objet du chapitre suivant.



Chapitre 3

Expérimentation de marches

3.1 Introduction

L'application principale du contrôleur a été le suivi de trajectoires sur le robot BIP 2D. Une campagne expérimentale a pu être conduite à partir de marches générées par la méthode vue dans le chapitre précédent. Nous avons réalisé une étude comparative de ces marches (Azevedo et al., 2002b ; Azevedo et Andreff, 2000) dans le but d'évaluer l'impact des paramètres sur l'optimalité des trajectoires en termes d'énergie dépensée, de sollicitations mécaniques...

3.2 Présentation des marches

Huit trajectoires numérotées de 0 à 7 sont considérées (fig.3.1), elles sont décrites dans le tableau suivant :

TAB. 3.1 – Description des différentes marches testées sur BIP.

	L_P	D_P	ΔH_{che}	$y_{han}(t_0)$	$y_{han}(t_m)$	
M0	38 cm	4,0 s	0,050 m	0,89 m	0,90 m	référence
M1	38 cm	2,0 s	0,050 m	0,89 m	0,90 m	pas rapides
M2	50 cm	4,0 s	0,050 m	0,89 m	0,90 m	enjambées longues
M3	38 cm	4,0 s	0,050 m	0,79 m	0,80 m	jambes très pliées
M4	38 cm	4,0 s	0,050 m	0,84 m	0,85 m	jambes pliées
M5	38 cm	4,0 s	0,050 m	0,89 m	0,89 m	altitude bassin fixée
M6	38 cm	4,0 s	0,100 m	0,89 m	0,90 m	enjambées hautes
M7	38 cm	4,0 s	0,050 m	0,89 m	0,90 m	marche arrière
Humain	77.5 cm	0,52 s	0.013 m	marche	normale	adulte mâle (70kg)

Remarque : L_P est la longueur des pas, D_P la durée d'exécution d'un pas, $t_m = \frac{t_f - t_0}{2}$ et $\Delta H_{che} = y_{che\ vol}(t_m) - y_{che\ vol}(t_0)$

On notera que nos déplacements sont constitués de pas plus courts et plus lents que dans le cas de la marche humaine normale.

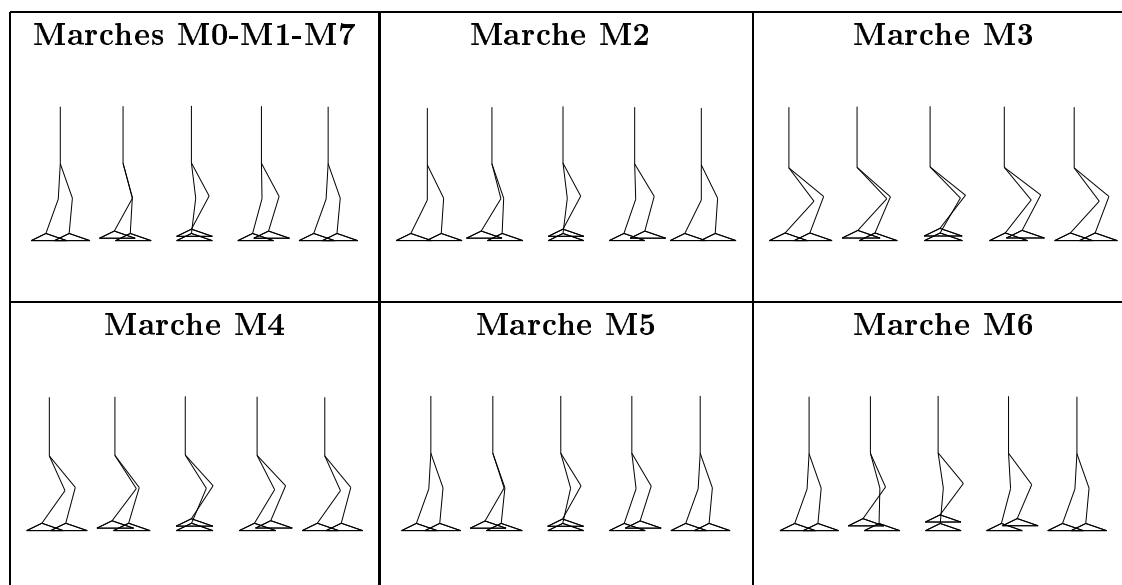
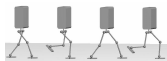


FIG. 3.1 – Différentes marches expérimentées.

La marche dite de référence est la marche **M0** que nous avons modifiée pour obtenir les sept autres.

3.3 Analyse de la marche de référence

Toutes ces marches ont été exécutées sur le robot (fig.3.2) et les données mesurées au cours de ces expérimentations ont été sauvegardées et analysées. Nous proposons ci-après une présentation du cas de la marche de référence **M0**, suivie d'une analyse comparative des différents résultats obtenus.

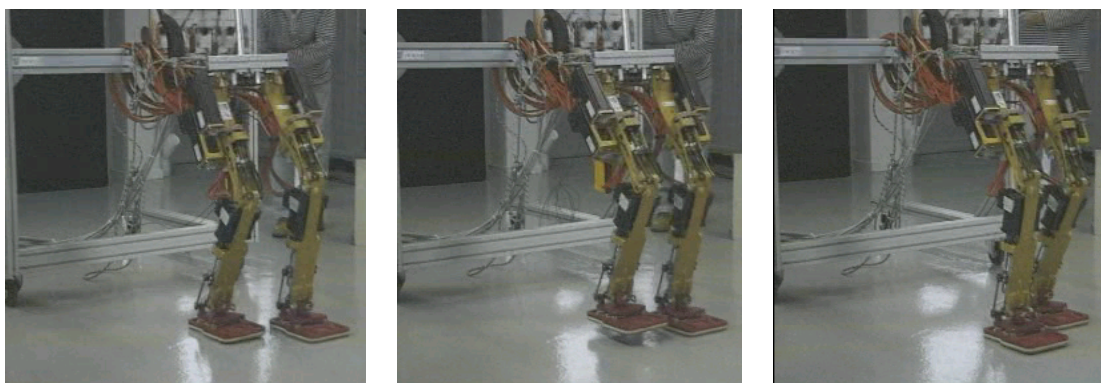
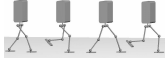


FIG. 3.2 – **M0** - images d'un film réalisé lors d'une expérimentation.



Poursuite de trajectoires articulaires

Les couples articulaires (fig.3.3) ont permis de suivre les trajectoires avec une erreur de poursuite (fig.3.3) inférieure à 0,02 rad (1,3 degrés).

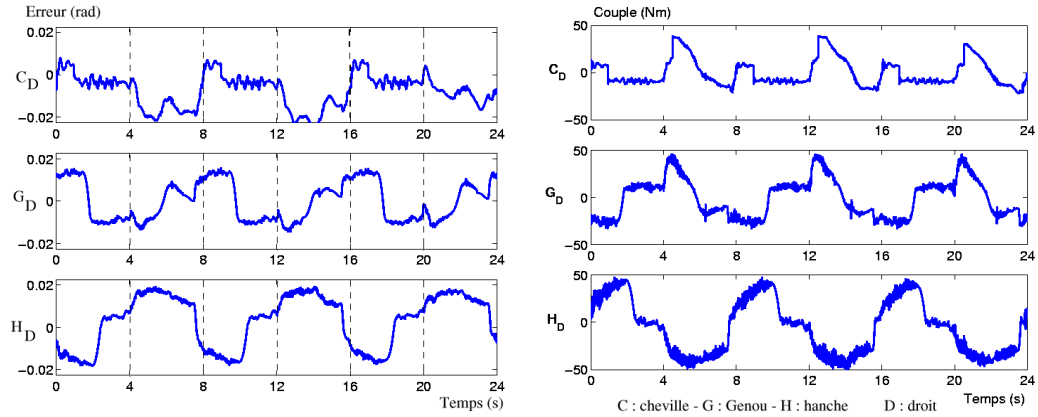


FIG. 3.3 – M0 - À gauche : erreurs de poursuite - À droite : couples articulaires mesurés (6 pas).

Changements d'appui

Nous avons vu que les marches générées ne présentent théoriquement pas de phase de double appui. On s'aperçoit en observant l'évolution des forces de réaction au sol (fig.3.4) qu'à l'exécution sur le robot, du fait des diverses erreurs, il existe de courtes périodes (200 ms) durant lesquelles le robot a ses deux pieds en contact avec le sol.

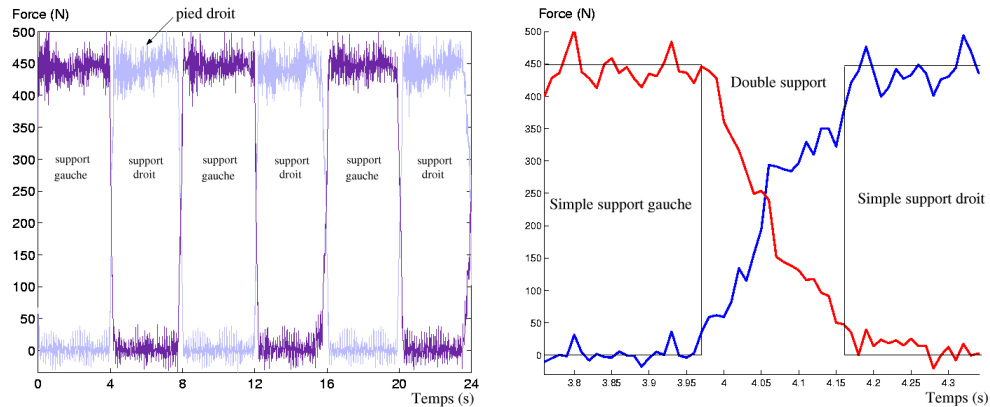
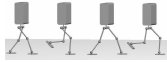


FIG. 3.4 – M0 - À gauche : forces de support normales (6 pas) (SG : support gauche, SD : support droit) - À droite : zoom sur la force de support (transition entre 2 pas).

On remarque également que le transfert de poids d'un pied sur l'autre lors du changement d'appui se fait linéairement et justifie ainsi le choix fait pour



le calcul du modèle en double appui. On peut remarquer l'absence de pics au niveau des efforts de support lors des poser de pieds, la somme des forces étant toujours proche du poids total du robot : 450 N (46 kg). Cela signifie qu'il n'y a pas d'impact lorsque le pied entre en contact avec le sol (les trajectoires ont d'ailleurs été définies dans cet objectif : vitesse nulle au contact).

Mouvements du centre de masse et du centre de pression

On peut vérifier que la projection du centre de masse demeure à l'intérieur du polygone de sustentation (fig.3.5). Malgré la présence du chariot et de la barre de liaison, on voit que le robot n'est pas complètement contraint dans le plan sagittal car le centre de pression (cdp) se déplace latéralement sur la semelle de support. Cela peut provenir des jeux mécaniques, de l'élasticité du montage, voire d'une défaillance de la barre,. Une autre explication est que si le robot ne comporte en théorie que 6 ddl, en réalité 8 ddl sont contrôlés. En effet, on a vu que les mouvements sagittaux et frontaux des chevilles sont couplés du fait de l'arrangement des moteurs en parallèle (page 42). Il est en pratique très difficile de les découpler totalement, notamment d'empêcher tout mouvement frontal de la cheville lorsqu'on actionne le mouvement sagittal. En théorie, lors de déplacements statiquement stables sur sol plat, centre de pression cdp et centre de masse (cdm) sont confondus : on voit que ce n'est pas tout à fait le cas en pratique (fig.3.5). L'explication est sans doute identique à celle énoncée précédemment, les erreurs de poursuite au niveau des chevilles ne garantissant pas le maintien horizontal des pieds.

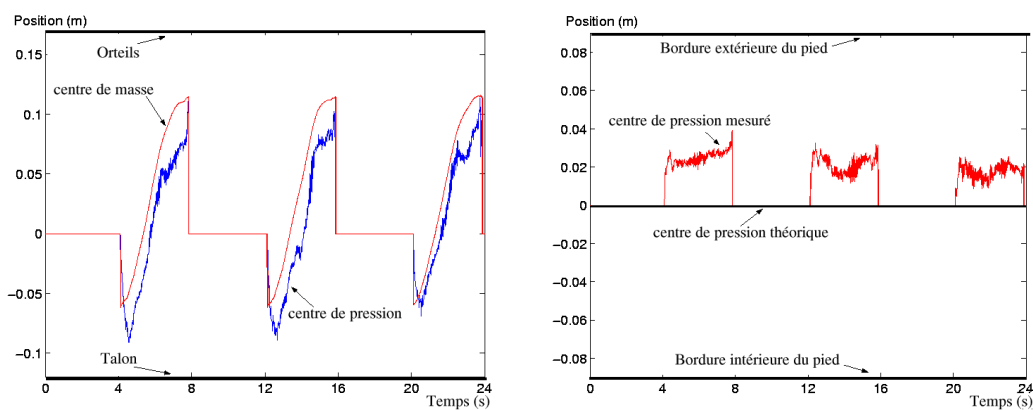


FIG. 3.5 – M0 - À gauche : comparaison entre les déplacements en avant du cdm et du cdp par rapport au pied droit sur 6 pas - À droite : comparaison entre les déplacements latéraux théoriques et réels du cdp sur 6 pas.

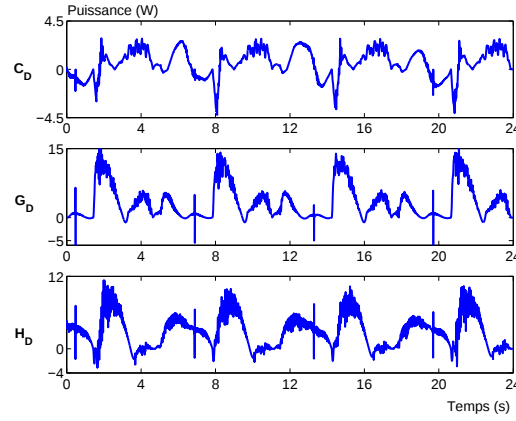


FIG. 3.6 – **M0** : Puissances articulaires sur 6 pas.

Puissance des couples articulaires

La puissance de chaque couple articulaire (fig.3.6) Γ_i peut-être calculée à partir de la formule :

$$P_i(t) = \Gamma_i(t)\dot{q}_i(t) \quad (3.1)$$

L'intégration de la puissance de l'instant 0 à l'instant t_f permet de calculer le travail des couples articulaires :

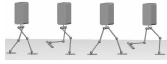
$$W_i(\Gamma_i) = \int_0^{t_f} |P_i(t)| dt \simeq \sum_{j=1}^{N_e} |P_i(j)| T_e \quad (3.2)$$

où : N_e correspond au nombre d'échantillons sauvegardés et T_e à la période d'échantillonnage.

On peut également sommer le travail de chacune des articulations pour avoir la dépense mécanique (énergie) totale. On obtient alors dans le cas de la marche **M0** une dépense d'énergie de : $14,5 \text{ J.s}^{-1}$ soit 153 J.m^{-1} (351 J.s^{-1} soit 234 J.m^{-1} pour un homme adulte) .

3.4 Comparaison des marches

Les différentes marches sollicitent plus ou moins les articulations. Le tableau 3.2 donne les vitesses articulaires maximales. On peut remarquer que les marches les plus contraignantes sont **M1** (marche rapide) et **M6** (enjambées plus hautes).



TAB. 3.2 – Vitesses articulaires maximales.

	$\dot{q}_{max}$ hanche	$\dot{q}_{max}$ genou	$\dot{q}_{max}$ cheville
M0	0,3 rad.s ⁻¹	0,4 rad.s ⁻¹	0,2 rad.s ⁻¹
M1	0,4 rad.s ⁻¹	0,9 rad.s ⁻¹	0,6 rad.s ⁻¹
M2	0,2 rad.s ⁻¹	0,3 rad.s ⁻¹	0,2 rad.s ⁻¹
M3	0,2 rad.s ⁻¹	0,2 rad.s ⁻¹	0,2 rad.s ⁻¹
M4	0,2 rad.s ⁻¹	0,2 rad.s ⁻¹	0,2 rad.s ⁻¹
M5	0,3 rad.s ⁻¹	0,5 rad.s ⁻¹	0,2 rad.s ⁻¹
M6	0,4 rad.s ⁻¹	0,7 rad.s ⁻¹	0,3 rad.s ⁻¹
M7	0,3 rad.s ⁻¹	0,4 rad.s ⁻¹	0,2 rad.s ⁻¹
Humain	4 rad.s ⁻¹	7 rad.s ⁻¹	4 rad.s ⁻¹

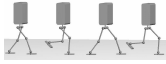
Le travail articulaire a été calculé pour l'ensemble des articulations, les valeurs sont données dans le tableau 3.3.

TAB. 3.3 – Comparaison des consommations énergétiques.

Marche	Vitesse déplacement	Dépense totale	Répart.
M0	0,095 m.s ⁻¹	14,5 J.s ⁻¹ /153 J.m ⁻¹	17-43-40 référence
M1	0,190 m.s ⁻¹	32 J.s ⁻¹ /168 J.m ⁻¹	17-46-37 pas plus rapides
M2	0,125 m.s ⁻¹	16 J.s ⁻¹ /128 J.m ⁻¹	19-42-39 enjambées longues
M3	0,095 m.s ⁻¹	12 J.s ⁻¹ /130 J.m ⁻¹	18-19-63 jambes très pliées
M4	0,095 m.s ⁻¹	12 J.s ⁻¹ /129 J.m ⁻¹	20-25-55 jambes pliées
M5	0,095 m.s ⁻¹	15 J.s ⁻¹ /166 J.m ⁻¹	17-41-42 altitude bassin fixée
M6	0,095 m.s ⁻¹	22 J.s ⁻¹ /235 J.m ⁻¹	15-50-35 enjambées hautes
M7	0,095 m.s ⁻¹	14 J.s ⁻¹ /146 J.m ⁻¹	24-43-33 marche arrière
Humain	1,500 m.s ⁻¹	351 J.s ⁻¹ /234 J.m ⁻¹	marche adulte mâle

La colonne Répart. contient le pourcentage d'énergie consommée pour les mouvements des chevilles-genoux-hanches.

Il apparaît nettement que la marche qui est la plus coûteuse est **M6**. Le choix d'une marche optimale dépendant des objectifs visés, il faut trouver un compromis entre puissances instantanées et vitesse de déplacement. On peut remarquer que la différence de consommation entre **M0** et **M1** est très faible pour une distance parcourue identique et des vitesses allant du simple au double. Le fait de bloquer



le bassin en hauteur (**M5**) est une contrainte importante sur la consommation. Le fait d'augmenter la longueur des enjambées (**M2**) permet d'augmenter la vitesse de déplacement pour un accroissement énergétique instantané faible et donc une énergie par distance parcourue moindre. Contrairement à ce que l'on pourrait attendre intuitivement, le fait de plier davantage les jambes n'induit pas d'augmentation d'énergie. Nous avons indiqué dans nos différents tableaux les valeurs anthropomorphes correspondantes. Il est néanmoins difficile de comparer nos données avec celles issues de la biomécanique car :

- les données présentées ici correspondent à des marches statiques très lentes en comparaison de la marche dynamique humaine,
- le robot BIP 2D ne comportant pas de tronc, la position de son centre de masse est très différente de celle de l'homme,
- l'énergie chez l'homme est essentiellement (à 70%) d'origine métabolique (le fait de plier les jambes chez l'homme créerait une fatigue musculaire que l'on ne retrouve pas dans le cas du robot (**M3** et **M4**)).

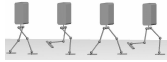
3.5 Conclusion

Une première analyse comparative de marches planes statiques a été menée pour évaluer l'influence des paramètres des pas sur la consommation du robot. Comme on pouvait s'y attendre, il s'agit essentiellement de déterminer une posture initiale de confort puis de trouver une allure et des enjambées qui permettent un compromis entre énergie instantanée et vitesse de déplacement.

Il est difficile de comparer les mouvements du robot à ceux de l'homme, notamment en ce qui concerne les dépenses énergétiques.

La marche de référence a fait l'objet d'un film présenté à l'Exposition Universelle de Hanovre en 2000.

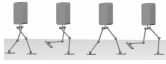
Il sera intéressant de réaliser des expérimentations similaires dans le cas de marches dynamiques avec le robot BIP 3D. La comparaison avec des données anthropomorphes sera alors beaucoup plus appropriée.



Conclusion

L'approche que nous avons présentée dans cette partie du mémoire est originale en ce sens qu'elle aborde le problème de la locomotion robotique bipède sous un angle complètement expérimental. En effet, depuis la génération de mouvements jusqu'à l'implantation du contrôleur, notre travail s'est articulé autour du prototype et de ses caractéristiques. Nous avons ainsi choisi des solutions simples et rapides de mise en œuvre, facilement adaptables aux version 2D et 3D du robot BIP.

L'autonomie est en général la préoccupation centrale de la robotique bipède. Certes, elle est indispensable lorsqu'il s'agit de réaliser une application où le système doit évoluer dans un environnement inconnu et décider de ses mouvements en temps réel pour faire face aux modifications de terrain ou aux perturbations. En revanche, si l'objectif est d'analyser des marches particulières dans le cadre d'une étude biomécanique, par exemple, un suivi de trajectoires peut être suffisant.



Quatrième partie

Optimiser et prédire : vers l'autonomie

Sommaire

- ★ **Problématique**

- contrôler la marche bipède sans spécifier de trajectoire de référence

- ★ **La commande prédictive**

- état de l'art

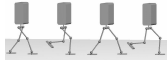
- ★ **Stratégie de commande**

- application de la stratégie prédictive pour la locomotion

- ★ **Vers la mise en œuvre de la commande**

- perspectives d'implantation temps réel

- ★ **Conclusion**

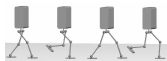


Problématique

Contrôler la marche bipède sans spécifier de trajectoire de référence.

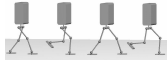
L'observation des comportements humains vis à vis de la locomotion est, selon nous, une source d'inspiration pour le contrôle des machines bipèdes. Dans la première partie de ce mémoire, nous avons mis en évidence les principes de base à respecter pour appréhender la marche dynamique, nous allons maintenant les exploiter pour synthétiser notre loi de commande.

L'originalité de notre approche repose sur la volonté de contrôler la locomotion d'un robot bipède sans spécifier de mouvement de référence. Ce parti pris de ne pas utiliser de trajectoire précalculée nous place d'emblée en opposition avec les stratégies existantes dans le domaine. La première difficulté va donc être de conceptualiser la notion de marche, autrement dit : comment expliquer au robot ce que l'on attend de lui sans le lui décrire, au sens classique du terme, avec la spécification de trajectoires. Nous avons précédemment souligné l'optimalité des marches humaines par rapport à des critères qui dépendent des objectifs poursuivis lors du déplacement (marche, course...). Cela nous oriente clairement vers l'utilisation d'une méthode de type commande optimale. L'optimisation en ligne des mouvements du robot est soumise à des contraintes liées aux capacités de ses articulations et actionneurs ainsi qu'aux caractéristiques de l'environnement. Il nous est apparu naturel de nous appuyer sur une commande de type prédictive (au sens horizon fuyant) qui, comme nous le verrons, se résout efficacement sous la forme d'un problème d'optimisation sous contraintes. Par ailleurs, l'horizon de commande permet de conserver les capacités d'anticipation existantes chez l'homme. La notion de prédiction repose nécessairement sur un modèle du bipède et de ses interactions avec l'environnement (contacts, impacts...) qui devra être aussi précis que possible. Dans l'approche que nous proposons, nous adaptons son cadre habituel basé essentiellement sur la poursuite de trajectoires désirées, à celui de la marche bipède sans suivi de marche de référence.



La notion de locomotion bipède est définie par un ensemble de contraintes inégalités. L'objectif a été de définir un cadre de travail sans se préoccuper dans un premier temps des difficultés numériques liées à la mise en œuvre de cette stratégie pour des systèmes à dynamique rapide, ainsi que les aspects associés au temps réel.

Le premier chapitre propose un état de l'art de la commande prédictive en insistant essentiellement sur le cas des systèmes non linéaires. Le chapitre suivant décrit la méthode que nous avons conçue et développée. La commande telle que nous l'avons définie ne rentre pas dans les cadres formels existants dont les hypothèses sont souvent très restrictives. Afin d'évaluer l'influence des différents choix possibles et réglages pour notre méthode, nous avons effectué une étude systématique sur le cas simple d'un pendule inverse. Nous présenterons des résultats encourageants pour l'application de la commande à la locomotion bipède obtenus en simulation. Enfin, nous ferons quelques propositions pour la mise en œuvre pratique de notre stratégie, tout particulièrement sur le point délicat du temps de calcul.



Chapitre 1

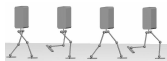
La commande prédictive

1.1 Introduction

La stratégie appelée **commande prédictive** consiste à optimiser, à partir des entrées (commande, état, forces de contact...), le comportement futur et prédit du système considéré. En anglais cette technique porte le nom de **model predictive control (MPC)**. La prédiction est faite à partir d'un modèle du système sur un intervalle de temps fini appelé **horizon de prédiction**. La première entrée de la séquence optimale est injectée au système et le problème est à nouveau résolu sur l'intervalle de temps suivant en utilisant les données du système mises à jour. La commande prédictive est également appelée **commande à horizon glissant** ou **commande à horizon fuyant** : **receding horizon control (RH)** en anglais, en référence à la manière dont la fenêtre de temps considérée pour les calculs est décalée à chaque itération. Le principal atout de la commande prédictive est sa capacité à prendre en compte dans son expression même les contraintes fonctionnelles et les contraintes d'exploitation du système considéré. L'inconvénient d'une telle méthode concerne les temps de calcul. C'est la raison pour laquelle elle a été essentiellement utilisée dans l'industrie du génie des procédés où les systèmes contrôlés sont suffisamment lents pour en permettre la mise en œuvre avec des périodes d'échantillonnage assez élevées.

Les origines de la commande prédictive remontent aux années 1960 avec (Propoi, 1963 ; Lee et Makus, 1967). Elle suscite un réel intérêt industriel depuis la fin des années 70 (Richalet et al., 1978 ; Cutler et Ramaker, 1979).

La commande prédictive consiste à prendre en compte à l'instant présent le comportement à venir en utilisant explicitement un modèle numérique du système afin de prédire les sorties futures sur un horizon fini. Il n'existe pas une stratégie unique, mais plutôt un ensemble de méthodes de commande prédictive bâties autour de principes communs, mais présentant néanmoins quelques différences dans l'interprétation des concepts clefs. Ces commandes sont habituellement utilisées pour les problèmes de poursuite et de suivi de trajectoire.



La méthodologie de commande prédictive se caractérise par les points suivants :

- à chaque instant, il faut prédire les sorties sur un certain horizon à l'aide d'un modèle du système,
- une trajectoire de référence peut être définie dans un futur proche afin de spécifier le ralliement de la sortie à la consigne,
- une séquence de commandes est élaborée afin de minimiser les erreurs futures entre la sortie et la consigne, ou entre la sortie et la trajectoire de référence,
- seul le premier élément de la séquence optimale précédente est appliqué au système. Tous les autres éléments peuvent être oubliés car à la période d'échantillonnage suivante les séquences sont décalées ; une nouvelle sortie est mesurée et la procédure complète est répétée.

À partir de ces points fondamentaux, plusieurs options permettent de distinguer une méthode particulière d'une autre.

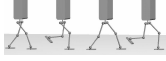
Dans le cas linéaire, les résultats de stabilité et les études de robustesse sont bien avancés (Garcia et al., 1989 ; Rawlings et Muske, 1993 ; Boucher et Dumur, 1996). De plus des expressions exactes de la commande peuvent être établies, ce qui ne nécessite pas de faire l'optimisation en ligne. On peut citer par exemple la commande prédictive généralisée (GPC) (Clarke et al., 1987) et la commande prédictive fonctionnelle (PFC) (Richalet, 1993).

Les premiers résultats concernant les systèmes non linéaires n'ont été publiés que dans la fin des années 1980 (Keerthi et Gilbert, 1988). La présence de contraintes inévitables dans les applications industrielles et l'augmentation de la puissance des calculateurs ont motivé le développement des commandes prédictives pour des systèmes décrits par des modèles non linéaires.

1.2 Principes

Dans la suite de ce mémoire et compte-tenu de la nature non linéaire de nos modèles, nous considérerons uniquement le cas de la commande prédictive non linéaire : **non-linear model predictive control (NMPC)**. Nous rappelons ici seulement les principes de la commande prédictive, pour plus de détails le lecteur est invité à consulter (Courtial, 1996 ; Allgöwer et al., 1999 ; DeNicolao et al., 2000 ; Dufour, 2000).

L'idée centrale est d'utiliser la connaissance issue d'un modèle pour envisager divers scénarios de fonctionnement du système dans le futur. Le scénario qui correspond le mieux aux objectifs visés est retenu.



Quatre éléments composent la stratégie de commande prédictive :

1. un **objectif de fonctionnement** qui définit la problématique posée. Cet objectif de fonctionnement comporte deux entités :
 - un **objectif** (ex. optimiser un critère de performance, assurer la poursuite d'une trajectoire de référence...)
 - des **contraintes de fonctionnement** sur les entrées et sur l'état (ex. limites des actionneurs, zones de fonctionnement...)
2. un **modèle du système** à commander qui traduit l'évolution du comportement du système
3. un **problème d'optimisation** qui est la traduction mathématique des objectifs de contrôle du système sous les contraintes de fonctionnement
4. une **méthode de résolution** pour le problème d'optimisation qui permet de déterminer les commandes à appliquer au système.

1.2.1 Objectif et problème d'optimisation

Il s'agit d'optimiser (minimiser en général) une fonctionnelle J appelée **critère de performance** ou **fonction de coût**. Ce critère dépend des variables de commande manipulables u , des grandeurs du système (vecteurs d'état x et de sortie y) et d'un comportement désiré c . Ces grandeurs dépendent du temps t .

$$\min_{u \in K} \{ J(x, u) = \int_{\tau} F(u(\tau), y(\tau), x(\tau), c(\tau)) d\tau \} \quad (1.1)$$

Pour un système continu, τ est un intervalle $[t, t + T_p]$ de longueur T_p correspondant au temps de prédiction. L'ensemble K définit les contraintes imposées.

Comme pour tout problème d'optimisation, la résolution numérique de ce problème nécessite une discrétisation.

Le problème d'optimisation est ainsi ramené à un problème d'optimisation classique en dimension finie :

$$\int_{\tau} (.) d\tau \rightarrow \sum_{\tau_i} (.) \Delta\tau \quad (1.2)$$

où :

- $u(t)$ est échantillonnée par une suite d'arguments $u(t_k)$ pris à des instants t_k connus :
 $k = 1, 2, \dots < \infty$
- l'intervalle τ est ramené à une succession d'éléments τ_i

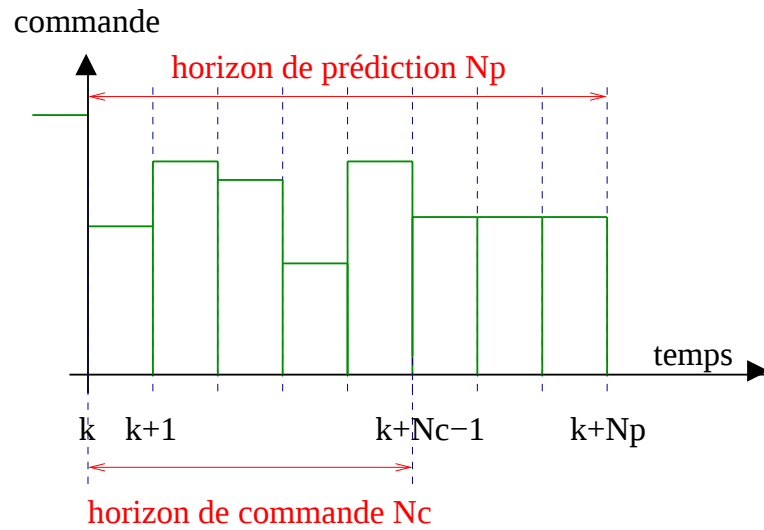


FIG. 1.1 – Principe de la commande prédictive : horizons de commande et de prédiction

La discrétisation précédente conduit à distinguer deux types d'horizon :

- l'horizon de commande N_c qui correspond à la longueur de la séquence de commande,
- l'horizon de prédiction N_p qui correspond à la longueur sur laquelle le critère de performance est évalué.

Le problème de la commande revient à déterminer la séquence des N_c commandes permettant de minimiser le critère de performance choisi tout en assurant une vérification des contraintes (fig.1.1).

Le principe de la commande est d'optimiser à chaque période d'échantillonnage le critère de performance soumis à des contraintes et de déterminer la meilleure séquence des N_c commandes sur l'horizon de prédiction N_p . La première séquence de commande solution est appliquée à la période d'échantillonnage suivante et la résolution recommence en prenant en compte les mesures actualisées du procédé. La répétition de cette procédure à chaque période permet de balayer le temps avec un horizon fini (horizon glissant). Pendant chaque période de temps, il faut donc disposer d'une méthode d'optimisation associée à une méthode de résolution (intégration) du modèle du système sur l'horizon de prédiction.

1.2.2 Poursuite de trajectoire

Une application classique de la commande prédictive est la poursuite de trajectoires de référence (fig.1.2). Le critère alors défini doit permettre d'apprécier la qualité de poursuite. Le critère se présente généralement sous la forme d'une mesure qua-

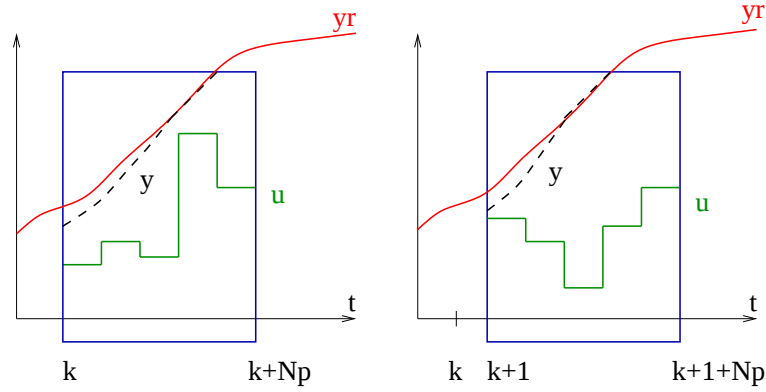
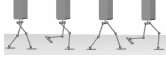


FIG. 1.2 – Principe de la commande prédictive : poursuite de trajectoire de référence

dratique d'erreur entre sortie de référence y^r et sortie du système y avec un terme limitant la variation de la commande afin d'assurer une continuité (Courtial, 1996) :

$$J = \sum_{i=0}^{N_p-1} (y_i^r - y_i)^2 + \beta(u_{i-1} - u_i)^2 \quad (1.3)$$

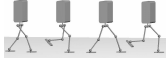
1.2.3 Faisabilité

La commande prédictive pose des problèmes structurels liés à la faisabilité (Scokaert et al., 1999 ; Kouvaritakis et al., 2000). La **faisabilité a priori** concerne l'existence ou non de solution au problème d'optimisation sous contraintes traduisant le problème de commande. La **stabilité** correspond à la **faisabilité a posteriori**, elle dépend des comportements futurs du système, c'est-à-dire du critère de performance et des contraintes. Il s'agit de vérifier que la commande, solution de l'algorithme, permet d'assurer la stabilité du procédé en boucle fermée (Dufour, 2000).

Cette notion de faisabilité est fondamentale au sens où dans le cadre d'une utilisation en temps réel, il est possible de se contenter d'une solution sous-optimale à partir du moment où la solution obtenue en arrêtant le processus d'optimisation avant convergence est garantie faisable.

1.2.4 Choix des horizons

Dans le cas des systèmes linéaires, il existe des méthodes pour fixer les horizons de prédiction et de commande (par rapport au temps de réponse désiré en boucle fermée). Dans le cas général, le problème reste ouvert puisque l'optimalité d'un



horizon est fortement liée à la complexité du problème. Le choix de l'horizon de prédiction influe sur la quantité d'information fournie à l'algorithme et sur l'existence de solutions numériques au problème d'optimisation. Sa détermination doit prendre en considération les caractéristiques physiques du comportement du modèle en boucle ouverte, l'objectif à atteindre et les contraintes. Un compromis est à trouver entre une grande fenêtre de prédiction assurant la maîtrise sur un temps plus long et un petit horizon garantissant des temps de calcul plus réalistes en vue d'un contexte temps réel. Une valeur élevée de l'horizon de commande permet a priori, en ayant plus de degrés de liberté, d'atteindre des objectifs plus difficiles. Par ailleurs, il faut garder à l'esprit que l'ensemble des calculs doit pouvoir être mené durant une période d'échantillonnage (Dufour, 2000).

1.2.5 Mise en œuvre temps réel

L'aspect le plus restrictif pour l'applicabilité de la commande prédictive est le temps de calcul. Il est nécessaire, en effet, pendant une période de temps d'échantillonnage, de trouver une solution admissible au problème d'optimisation sous contraintes. D'un point de vue formel, il faut noter que le temps de convergence est non borné, c'est-à-dire non connu a priori en pratique. Ceci est incompatible avec les contraintes de temps réel et un échantillonnage régulier. Par ailleurs, le modèle est également à résoudre et à intégrer pendant ce même temps et ce, plusieurs fois. Effectivement, les sorties futures du procédé doivent être évaluées sur l'horizon de prédiction. Le choix de l'algorithme est donc crucial. On peut exclure a priori les algorithmes de recherche d'optimum global au profit de la recherche d'un optimum local (mais faisable). Le temps de recherche d'une solution fait partie des contraintes de mise en œuvre de l'algorithme.

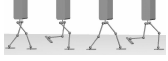
1.3 Commande prédictive non linéaire

1.3.1 Rappels préliminaires sur les systèmes non linéaires

Nous considérons la classe des systèmes non linéaires discrets, invariants par rapport au temps, décrits par l'ensemble d'équations différentielles :

$$x(k+1) = f(x(k), u(k)) \quad (1.4)$$

où : $x(k) \in \mathbb{R}^n$ est l'état, $u(k) \in \mathbb{R}^m$ est l'entrée (commande).



Hypothèses :

(H1) l'origine de f est un point d'équilibre $\rightarrow f(0, 0) = 0$

(H2) à chaque instant k l'état x_k est connu

Note : dans la suite on notera $x_k = x(k)$ et $u_k = u(k)$.

⊕ Contraintes

Le système (1.4) est soumis à des contraintes sur l'état et sur la commande :

$$x_k \in \mathcal{X}, \quad u_k \in \mathcal{U} \quad (1.5)$$

où : $\mathcal{X}$ et $\mathcal{U}$ sont des ensembles de $\mathbb{R}^n$ et $\mathbb{R}^m$ respectivement, contenant chacun l'origine :

$$\begin{aligned} \mathcal{X} &:= \{x_k^i \in \mathbb{R} \mid x_{min}^i \leq x_k^i \leq x_{max}^i\} \quad i = 1, \dots, n \\ \mathcal{U} &:= \{u_k^i \in \mathbb{R} \mid u_{min}^i \leq u_k^i \leq u_{max}^i\} \quad i = 1, \dots, m \end{aligned} \quad (1.6)$$

Hypothèses :

(H3) $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est deux fois différentiable dans $\mathcal{X} \times \mathcal{U}$

(H4) $\mathcal{U} \subset \mathbb{R}^m$ est compact et convexe, $\mathcal{X} \subset \mathbb{R}^n$ est fermé

(H5) le système ne subit aucune perturbation

(H6) le modèle utilisé représente parfaitement le système

⊕ Stabilité

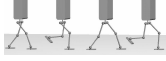
Soit la loi de commande $u = \mathcal{K}(x)$ avec $\mathcal{K}(0) = 0$. Le système en boucle fermée associé s'écrit :

$$x_{k+1} = F(x_k) \quad \text{avec } F(x) = f(x, \mathcal{K}(x)) \quad (1.7)$$

$\mathcal{K}(\cdot)$ est dite **stabilisante** si l'origine de (1.7) est un point d'équilibre **asymptotiquement stable**.

L'origine de (1.7) est un point d'équilibre **asymptotiquement stable** si pour tout $\epsilon > 0$ il existe $\delta_\epsilon > 0$ tel que si $\|x_t\| < \delta_\epsilon$ alors $\lim_{k \rightarrow \infty} \|x_k\| = 0, \forall k \geq t$ avec x_k solution de (1.7).

Il est intéressant de savoir à quelle distance du point d'équilibre peut se trouver la trajectoire tout en conservant la convergence vers l'origine lorsque t tend vers l'infini. L'ensemble des points à partir desquels la convergence est satisfaite est appelé **région d'attraction** (ou *domaine d'attraction* ou *bassin d'attraction*).



⊕ **Commande quadratique linéaire** (*linear quadratic* LQ)

Une façon classique de stabiliser (1.4) est d'utiliser une commande linéaire quadratique (qui dans son concept se rapproche d'une commande prédictive) basée sur la linéarisation de (1.4) autour du point d'équilibre.

Hypothèses :

(H7) la fonction de coût F est une fonction quadratique en x et u :

$$F(x, u) = x^T Q x + u^T R u \text{ où : } Q, R \text{ sont des matrices définies positives}$$

(H8) le modèle linéarisé de (1.4) autour du point d'équilibre ($x = 0, u = 0$) :

$$x_{k+1} = A x_k + B u_k \text{ est stabilisable (avec } A = \frac{\partial f}{\partial x}|_{x=0, u=0} \text{ et } B = \frac{\partial f}{\partial u}|_{x=0, u=0})$$

Le but est de déterminer K tel que $A + BK$ soit asymptotiquement stable et d'appliquer alors la loi de commande $u = Kx$ au système non linéaire (1.4). K peut être déterminé en résolvant le problème de minimisation sur un horizon infini :

$$\min_{u(\cdot)} \sum_{k=t}^{\infty} x_k^T Q x_k + u_k^T R u_k \quad (1.8)$$

soumis à la contrainte issue de la dynamique linéarisée.

La solution du problème LQ précédent est donnée par le retour d'état linéaire :

$$u_k = K^{LQ} x_k \quad (1.9)$$

où :

$$K^{LQ} = -(R + B^T P_{\infty} B)^{-1} B^T P_{\infty} A \quad (1.10)$$

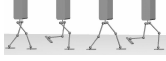
P_{∞} est la solution définie non négative de l'équation de Riccati algébrique :

$$P = A^T P A + Q - A^T P B (R + B^T P B)^{-1} B^T P A \quad (1.11)$$

Lorsque (1.9) est appliquée à (1.4) on obtient le système en boucle fermée :

$$x(k+1) = f(x_k, K^{LQ} x_k) \quad (1.12)$$

La stabilité asymptotique de l'origine est démontrée à partir de la fonction de Lyapunov $V(x) = x^T P_{\infty} x$ (Vidyasagar, 1993).



Cette stratégie de commande optimise la performance uniquement lorsque l'on se trouve proche du point d'équilibre. Ainsi sa validité dépend de la manière dont le système peut être approximé par sa linéarisation sur la région considérée. Cependant sa mise en œuvre est aisée car les calculs sont effectués hors ligne. Les limitations liées à l'approximation linéaire ont conduit à proposer des stratégies de type commande prédictive non linéaire (NMPC).

1.3.2 Commande prédictive non linéaire

⊕ Formulation

Le problème de la commande prédictive non linéaire consiste à trouver une stratégie de commande stabilisante telle que la fonctionnelle :

$$J(x_k, u_k^{N_p}) = \sum_{i=0}^{N_p-1} F(x_{i|k}, u_{i|k}) \quad (1.13)$$

soit minimale sous les contraintes :

$$\begin{aligned} x_{i+1|k} &= f(x_{i|k}, u_{i|k}), \quad x_{0|k} = x_k && \text{modèle dynamique du système} \\ u_{i|k} &\in \mathcal{U} && \text{contraintes sur la commande} \\ x_{i|k} &\in \mathcal{X} && \text{contraintes sur l'état} \end{aligned} \quad (1.14)$$

La notation $s_{i|k}$ signifie que la variable s est prédite à l'instant courant k . i indique l'instant de prédiction du contrôleur à partir de k . Ainsi :

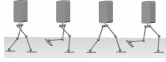
$$u_k^{N_p} = \{u_{0|k}, u_{1|k}, \dots, u_{N_p-1|k}\} \quad (1.15)$$

À chaque instant le premier élément $u_k = u_{0|k}$ de la séquence optimale ($u_k^{N_p}$) est appliqué au système.

Remarque : si les longueurs des horizons de prédiction N_p et de commande N_c sont distincts, on a : $u_k^{N_p} = \{u_{0|k}, u_{1|k}, \dots, u_{N_c-1|k}, u_{N_c-1|k}, \dots, u_{N_p-1|k}\}$

Nous récapitulons ci-dessous les hypothèses auxquelles nous aurons à faire référence :

- (H1) l'origine de f est un point d'équilibre $\rightarrow f(0, 0) = 0$
- (H2) à chaque instant k l'état x_k est connu
- (H3) $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est deux fois différentiable dans $\mathcal{X} \times \mathcal{U}$
- (H4) $\mathcal{U} \subset \mathbb{R}^m$ est compact et convexe, $\mathcal{X} \subset \mathbb{R}^n$ est fermé
- (H5) le système ne subit aucune perturbation
- (H6) le modèle utilisé représente parfaitement le système
- (H7) le coût F est une fonction quadratique en x et u : $F(x, u) = x^T Q x + u^T R u$ avec Q, R définies positives
- (H8) (A, B) est stabilisable (avec $A = \frac{\partial f}{\partial x}|_{x=0, u=0}$ et $B = \frac{\partial f}{\partial u}|_{x=0, u=0}$)



⊕ Horizon infini

La technique du NMPC à horizon infini consiste à prendre une valeur de $N_p = \infty$. Si les hypothèses (H1) à (H6) sont vérifiées et si le problème d'optimisation en boucle ouverte est faisable à $k = 0$, alors l'origine du système en boucle fermée sous la commande NMPC est asymptotiquement stable (Allgöwer et al., 1999 ; Mayne et Michalska, 1990 ; Meadows et Rawlings, 1993). La solution d'un tel problème avec $N_p = \infty$ est extrêmement difficile, voire impossible à obtenir en pratique (contrairement au cas des systèmes linéaires), d'où la nécessité de développer des techniques basées sur des horizons finis.

1.3.3 Différentes approches NMPC à horizon fini

Le problème standard de commande prédictive non linéaire avec un horizon fini arbitraire et une fonction de coût également arbitraire ne conduit en général pas à un système asymptotiquement stable en boucle fermée (Bitmead et al., 1990). Il existe alors différentes approches garantissant la stabilité sous certaines conditions.

★ Contrainte égalitaire terminale

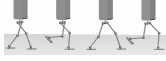
Une première formulation du problème de commande prédictive à horizon fini consiste à introduire dans le problème une contrainte d'état terminale (commandabilité en N_p itérations) :

$$x_{N_p|k} = 0 \quad (1.16)$$

La contrainte égalité force l'état à se trouver sur le point d'équilibre à la fin de l'horizon fini N_p . Si les hypothèses (H1) à (H6) sont vérifiées et si le problème d'optimisation en boucle ouverte est faisable à $k = 0$, alors l'origine du système en boucle fermée sous la commande NMPC est asymptotiquement stable (Allgöwer et al., 1999). Cette stratégie existe en version temps continu (Chen et Shaw, 1982 ; Mayne et Michalska, 1990) et en version temps discret (Alamir et Bornard, 1994). Cependant, d'un point de vue numérique la satisfaction exacte d'une contrainte égalitaire terminale est impossible à mettre en œuvre (Keerthi et Gilbert, 1988). D'où l'idée de converger non plus vers un point d'équilibre mais vers une région centrée autour de ce point d'équilibre.

★ Mode dual

Une façon de contourner le problème de la contrainte terminale est d'utiliser une stratégie de mode dual qui propose d'utiliser deux modes de commande appliqués



dans différentes régions de l'espace d'état en fonction de la position de l'état par rapport à une région terminale pré-définie Ω_f qui contient l'origine. En dehors de Ω_f , la stratégie de commande prédictive est utilisée. À l'intérieur de Ω_f , un retour d'état linéaire pré-défini est appliqué :

$$u_k = \begin{cases} u_{0|k}^* & \text{si } x_k \notin \Omega_f \\ Kx_k & \text{si } x_k \in \Omega_f \end{cases} \quad (1.17)$$

K est défini de manière à garantir que Ω_f soit une région d'attraction positive invariante du système linéaire contrôlé par le retour d'état et que les contraintes sur l'état et la commande demeurent satisfaites par le contrôleur linéaire dans Ω_f . Il existe des procédures de calcul de K et Ω_f sous l'hypothèse que (H8) soit vérifiée (Findeisen et Allgöwer, 2000). Dans le problème dual NMPC, lorsque l'état est en dehors de Ω_f , N_p est pris comme variable de minimisation du critère :

$$\min_{u_k^{N_k}, N_k} J(x_k, u_k^{N_k}, N_k) \quad (1.18)$$

Ω_B désigne la frontière de la région Ω_f . La contrainte terminale inégalité suivante est également ajoutée :

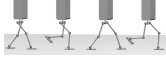
$$x_{N_p|k} \in \Omega_B \quad (1.19)$$

Sous les hypothèses (H1) à (H8) et si le problème en boucle ouverte est faisable à $k = 0$, alors l'origine du système en boucle fermée est asymptotiquement stable.

L'intérêt de cette méthode est de remplacer la contrainte égalité à la fin de l'horizon par une région. La stabilité dépend essentiellement de la faisabilité de la solution du problème d'optimisation sous contraintes alors que l'optimalité globale de cette solution n'est pas strictement nécessaire (Scokaert et al., 1999). Le mode dual existe en version temps continu (Michalska et Mayne, 1993) et en version temps discret (Chisci et al., 1996). Les inconvénients de cette solution sont la nécessité de commuter d'une commande à l'autre et la taille de la région à atteindre qui peut être petite, donc longue à atteindre.

★ Horizon quasi infini

La stratégie horizon quasi infini est basée sur la même idée que celle du mode dual dans laquelle l'état doit se trouver au bout d'un temps fini, dans un domaine Ω_f . Cependant, au lieu de commuter sur une commande linéaire, une contrainte de



pénalité de type inégalité portant sur l'état et représentant une borne supérieure pour le coût à horizon infini est posée :

$$x_{N_p|k} \in \Omega_f \quad (1.20)$$

Un terme de pénalité quadratique portant sur la norme future de l'état, est ajouté dans le critère pour la troncature de ce dernier :

$$\Phi(x_{N_p|k}) = x_{N_p|k}^T P x_{N_p|k} \quad (1.21)$$

On a donc :

$$J(u_k^\infty, x_k) = \sum_{i=0}^{\infty} F(x_{i|k}, u_{i|k}) = \sum_{i=0}^{N_p-1} F(x_{i|k}, u_{i|k}) + \sum_{i=N_p}^{\infty} F(x_{i|k}, u_{i|k}) \quad (1.22)$$

L'objectif est d'approximer le second terme par le terme de pénalité Φ . Cela est possible si l'on s'assure que les trajectoires du système en boucle fermée demeurent dans un voisinage de l'origine (région terminale) sur l'intervalle de temps $[k+N, \infty[$. La région Ω_f est construite telle qu'un retour d'état : $u_{i|k} = Kx_{i|k}$ stabiliserait asymptotiquement le système dans Ω_f (avec Ω_f invariant positif pour la boucle fermée). Cependant, dans le cas de l'horizon quasi-infini, cette commande linéaire n'est jamais appliquée.

On a :

$$\min_{u_k^\infty} J(u_k^\infty, x_k) \leq \min_{u_k^\infty} \sum_{i=0}^{N_p-1} F(x_{i|k}, u_{i|k}) + \sum_{i=N_p}^{\infty} F(x_{i|k}, Kx_{i|k}) \quad (1.23)$$

Ainsi, une construction judicieuse de P et Ω_f permet de majorer le coût lorsque l'on se trouve dans Ω_f (Allgöwer et al., 1999) :

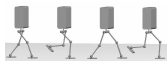
$$\sum_{i=N_p}^{\infty} F(x_{i|k}, Kx_{i|k}) \leq x_{N_p|k}^T P x_{N_p|k} \quad (1.24)$$

D'où :

$$\min_{u_k^\infty} J(u_k^\infty, x_k) \leq \min_{u_k^{N_p}} J(u_k^{N_p}, x_k) \quad (1.25)$$

Cette dernière équation permet de démontrer la stabilité asymptotique du système en boucle fermée sous les hypothèses (H1) à (H8).

Cette stratégie existe en version temps continu (Garcia et al., 1989) et discret (De-Nicolao et al., 1997).



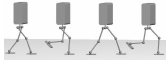
1.4 Conclusion

L'approche commande prédictive présente des difficultés liées à la résolution du problème d'optimisation non linéaire sous contraintes, au choix des algorithmes d'optimisation pour un problème non linéaire sous contraintes, à l'existence de solution, à la convergence en temps fini, à la stabilité et à la résolution du modèle non linéaire du procédé. Ces calculs doivent être effectués suffisamment rapidement pour satisfaire aux contraintes de temps réel. Les problèmes de choix du critère, de l'horizon, de méthodes de résolution et d'études de stabilité demeurent ouverts.

Bien que la commande prédictive non linéaire soit largement utilisée dans l'industrie (Allgöwer et al., 1999), l'application de la commande prédictive aux systèmes non linéaires à dynamique rapide pose des problèmes liés aux aspects numériques et aux difficultés théoriques.

Un principe commun à toutes ces approches est la nécessité de définir une trajectoire de référence ou un état final désiré a priori. Dans la marche bipède, ce concept est quelque peu limitatif quand on pense à la multitude de changements de marches (obstacles, virages, pentes...). De plus, la précision des mouvements n'est pas nécessaire dans la plupart des cas.

Nous verrons dans le chapitre suivant comment nous avons utilisé et adapté la stratégie de NMPC pour le contrôle de la marche dynamique de robots bipèdes, sans avoir besoin de définir au préalable une trajectoire à suivre.



Chapitre 2

Stratégie de commande

2.1 Introduction

L'approche que nous proposons pour le contrôle de la marche dynamique de robots bipèdes est inspirée des techniques de commande prédictive. Leur utilisation se justifie en effet par leur capacité à synthétiser une commande pour des systèmes non linéaires multi-variables dont les variables d'état et/ou de commande peuvent être contraintes.

Le problème de la synthèse de la marche et de sa commande est ici posé sous la forme d'un problème d'optimisation sous contraintes (contraintes physiques et technologiques, contraintes relatives aux déplacements et contraintes liées aux contacts) défini avec un horizon glissant sur lequel il est possible de mettre à jour en ligne les contraintes pour les adapter à l'environnement (Azevedo et al., 2002d ; Azevedo et al., 2002c ; Azevedo et Poignet, 2002). La particularité essentielle de la technique proposée est de générer un mouvement sans trajectoire de référence définie *a priori*. En minimisant une fonction de coût quadratique calculée à partir des couples moteurs, nous générons un mouvement contrôlé du robot lors des phases de marche grâce à des contraintes assurant la locomotion. La spécification de ces contraintes s'appuie sur l'observation de la marche humaine.

Dans un premier temps, nous présentons la méthode développée et sa mise en œuvre numérique. Une présentation des notions de stabilité et de faisabilité est ensuite proposée. La section suivante analyse le problème sur l'exemple simple du pendule inversé afin de caractériser des aspects numériques liés à la résolution du problème posé. Des résultats de simulation seront exposés dans le cas de la locomotion bipède. Enfin, nous ferons quelques propositions en vue d'une mise en œuvre expérimentale de la stratégie.

2.2 Stratégie

La stratégie développée consiste à calculer un ensemble de couples actionneurs sur un horizon glissant en définissant le mouvement, non pas au travers d'une trajectoire de référence définie *a priori*, mais grâce à un ensemble de contraintes physiques

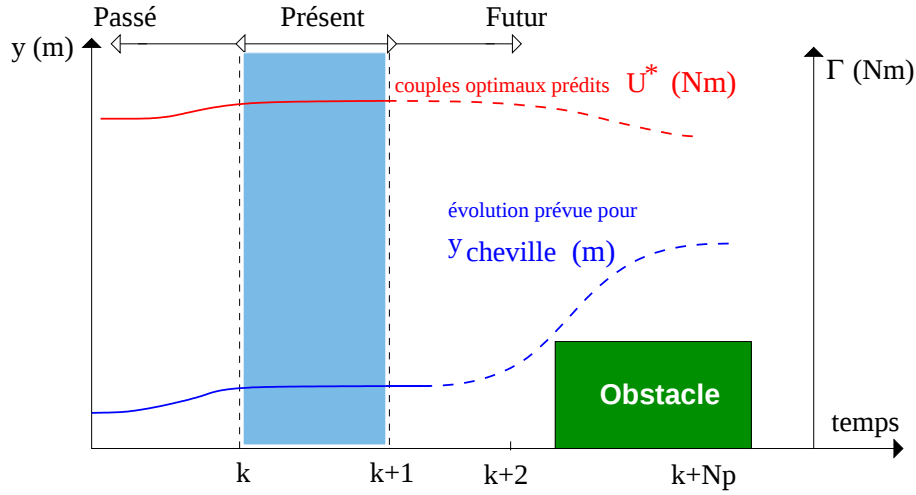
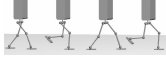


FIG. 2.1 – Utilité de l'horizon de prédiction.

et technologiques qui garantissent le déplacement du robot sans chute. Du fait du principe d'horizon fuyant, il est alors tout à fait envisageable de mettre à jour en ligne les contraintes pour les adapter à l'environnement et éviter les situations bloquantes en les anticipant (fig.2.1).

2.2.1 Critère

Le critère J optimisé est une fonction quadratique des couples des actionneurs nécessaires sur l'horizon $H_p = [k + 1, \dots, k + N_p]$:

$$J(U(k)) = \frac{1}{2} \sum_{j=k+1}^{j=k+N_p} \Gamma(j)^T \Gamma(j) \quad (2.1)$$

Note : Nous supposons dans la suite que les longueurs des horizons de commande et de prédiction sont identiques.

La minimisation du critère permet de déterminer, à chaque période d'échantillonnage, le vecteur optimal $U^*(k)$ de l'ensemble des couples,

$$U(k)^* = [\Gamma(k + 1), \Gamma(k + 2), \dots, \Gamma(k + N_p)]^T$$

défini sur l'horizon H_p . Mais seule la première composante $\Gamma(k + 1)$ est utilisée et appliquée effectivement au système. Le calcul est ainsi réitéré à chaque période d'échantillonnage et $U^*(k + 1)$ est calculé à l'instant $k + 1$ (fig.2.2). L'initialisation peut se faire à chaque pas de temps avec le vecteur : $U_0 = [\Gamma(k), \Gamma(k), \dots, \Gamma(k)]^T$ de dimension $N_p \times m$ (m étant le nombre d'actionneurs), $\Gamma(k)$ étant le couple appliqué à l'instant précédent k .

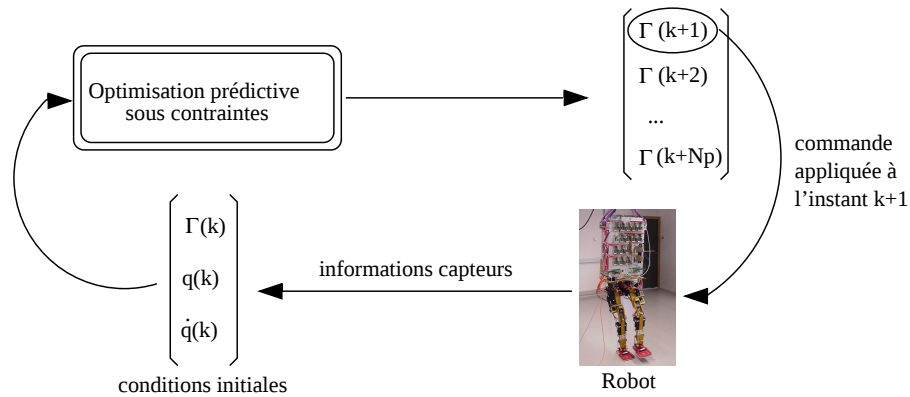
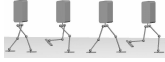


FIG. 2.2 – Principe de la stratégie de commande proposée

2.2.2 Choix des contraintes

Le point fondamental de notre approche est le choix des contraintes qui vont permettre d'assurer un mouvement ou une posture. Les contraintes sont ainsi imposées en observant le comportement humain. Dans (Hurmuzlu, 1993a ; Patla, 1991), trois éléments sont essentiels dans la locomotion (voir la première partie de ce mémoire) :

1. la production d'un rythme de locomotion de base,
2. le maintien de l'équilibre,
3. l'adaptation des mouvements à l'environnement.

Un soin particulier doit être apporté à la trajectoire de la jambe libre qui tient un rôle important au niveau du rythme et de l'équilibre du bipède, mais aussi pour le franchissement d'obstacles (Pratt et Pratt, 1998). À partir de ces considérations intuitives mais physiques, il est possible d'établir un ensemble de contraintes à satisfaire pour assurer la locomotion :

- **Dynamique du robot** : l'équation de la dynamique constitue la première contrainte à satisfaire. Les équations associées dépendent du modèle utilisé.
- **Contraintes fonctionnelles du robot** : les couples optimaux doivent correspondre aux potentialités des actionneurs et les mouvements engendrés demeurer à l'intérieur des limites autorisées.
- **Maintien de la posture érigée** : le robot doit conserver la position debout.
- **Équilibre** : le système doit constamment maintenir sa stabilité statique ou dynamique.
- **Rythme de progression** : la locomotion correspond à un déplacement vers l'avant.
- **Adaptation à l'environnement** : le robot doit s'adapter au sol, aux obstacles et autres escaliers. Cela est essentiellement lié à la trajectoire de la jambe libre.

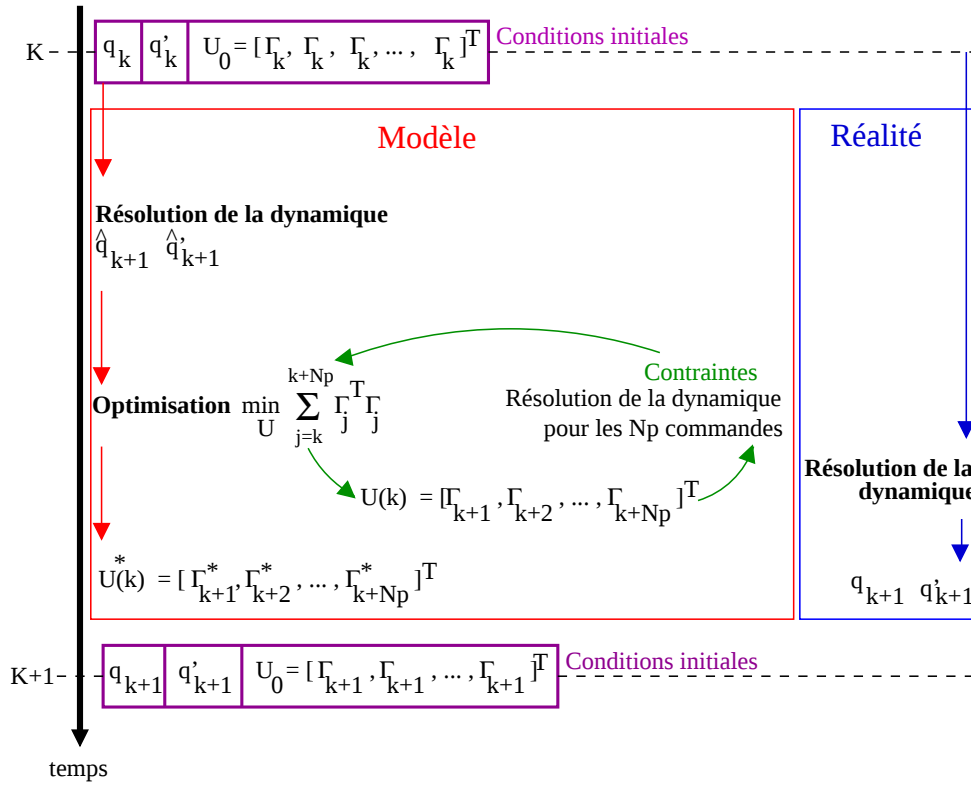
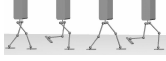


FIG. 2.3 – Principe de la commande

Ces contraintes de base permettent de caractériser la marche bipède ; nous les compléterons et modifierons au cas par cas suivant les exemples traités afin de les adapter aux spécificités du problème (marche statique ou dynamique, marche sous-actionnée...). Toute la difficulté est de déterminer le jeu d'inégalités traduisant les contraintes de base qui va garantir la stabilité du robot tout en lui permettant de se déplacer et en assurant un maximum de réactivité aux changements d'environnement ou aux apparitions de perturbations. Nous reviendrons sur ce point plus précisément au §.2.5.1.

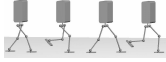
2.2.3 Implantation numérique

⊕ Généralités

La programmation de la loi de commande présentée est assez immédiate sous les environnements `matlab`¹ et `scilab`² qui disposent de fonctions *ad hoc* pour la résolution de problèmes d'optimisation. La figure 2.3 présente le principe de l'algorithme que nous avons mis en œuvre.

¹<http://www.mathworks.com/products/matlab>

²<http://www-rocq.inria.fr/scilab/>



La stratégie doit assurer à chaque pas du calcul k qu'une commande Γ_k est appliquée au système pendant l'intervalle de temps $[k, k + 1]$. Cette commande fait passer le système de l'état $X_k = [q_k; \dot{q}_k]^T$ à l'état $X_{k+1} = [q_{k+1}; \dot{q}_{k+1}]^T$. Pendant l'intervalle de temps $[k, k + 1]$, la commande Γ_{k+1} qui sera appliquée sur l'intervalle suivant $[k + 1, k + 2]$ est calculée.

Il s'agit ainsi de déterminer $U_{k+1} = [\Gamma_{k+1}; \Gamma_{k+2} \dots \Gamma_{k+N_p}]$ de manière à minimiser le critère tout en satisfaisant aux contraintes définies. Ces contraintes portent pour la plupart sur des positions articulaires ou cartésiennes, ce qui implique une résolution et une intégration du modèle dynamique du système à chaque itération de l'algorithme et ce, pour les N_p vecteurs Γ_i du vecteur U_{k+1} . À l'instant $k + 1$, on applique Γ_{k+1} et on recommence la procédure.

Remarque 1 : un premier problème qu'il faudra impérativement résoudre en vue d'une utilisation expérimentale est la maîtrise des temps de calcul. Le choix des algorithmes d'optimisation joue également un rôle crucial ; nous illustrerons leur influence dans les exemples traités par la suite.

Remarque 2 : pour la sécurité du système, si l'algorithme ne trouve pas de solution, la commande précédente est ré-injectée aux actionneurs.

⊕ Algorithmes d'optimisation

Parmi les techniques d'optimisation non linéaire sous contraintes, on distingue classiquement trois types de méthodes (Courtial, 1996) :

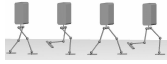
- les méthodes directes ou méthodes primales

Le problème de minimisation sous contraintes est traité sous sa forme directe. Le principe est de trouver une séquence qui minimise la fonction objectif en vérifiant l'admissibilité des solutions obtenues. L'idée est de projeter à chaque étape le déplacement (gradient) de la fonction objectif sur les frontières du domaine des contraintes, de sorte que le nouveau point appartienne à l'ensemble des solutions admissibles.

- les méthodes intermédiaires ou méthodes primales-duales

Le principe des méthodes intermédiaires est de résoudre les conditions de Kuhn-Tucker par des méthodes de programmation linéaire (simplexe) ou de programmation quadratique (Newton). Ces méthodes sont connues pour leur efficacité (au voisinage de l'optimum) :

- Méthodes SLP (sequential linear programming) : le problème linéarisé est résolu séquentiellement autour du point courant. L'inconvénient est que l'approximation tangentielle au point courant peut converger vers un point non optimal pour le problème initial.



Méthodes SQP (sequential quadratic programming) : une approximation tangentielle au second ordre du problème non-linéaire est effectuée. Ces méthodes sont reconnues pour leur comportement asymptotique efficace. L'inconvénient majeur est qu'elles ne possèdent pas la propriété de convergence globale³ (annexe p.151), c'est à dire que si le point courant initial est éloigné de l'optimum, la validité de l'approximation quadratique n'est pas garantie. Une variante intéressante est un algorithme de type **FSQP**. Cette méthode est basée sur l'algorithme SQP modifié pour que les itérations générées soient faisables (Craig et al., 1997).

- les méthodes duales

Le problème est résolu indirectement après la définition d'un problème équivalent qui consiste à ramener les contraintes à travers des variables. On peut citer les **méthodes de pénalités** dont la solution du problème équivalent est une approximation de la solution initiale et y tend asymptotiquement. Un algorithme d'optimisation sans contraintes doit être utilisé. Dans ce domaine, les algorithmes de **Levenberg-Marquardt** et de **Quasi-Newton** (utilisé par Matlab) sont reconnus comme efficaces (Fletcher, 1987).

L'inconvénient de ces méthodes réside dans le manque de résultats prouvés sur la convergence. Les seuls résultats dans le cas non linéaire concernent des problèmes convexes en la fonction objectif et les contraintes.

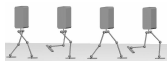
2.3 De la stabilité à la faisabilité

La notion de stabilité est difficile à appréhender dans le cas de notre approche, étant donné que l'on n'utilise pas de trajectoire de référence et encore moins de point d'équilibre. Nous verrons que le problème tel que nous l'avons posé ne rentre pas dans les cadres formels existants. Par ailleurs, nous allons voir que la notion de faisabilité serait davantage adaptée à notre cas de figure.

2.3.1 Étude de stabilité

Les travaux les plus récents sur la stabilité de la commande prédictive à horizon fini pour les systèmes non linéaires discrets (Alamir et Bornard, 1995 ; Scokaert et al., 1997) sont basés sur la définition d'une fonction de Lyapunov qui est égale au critère à minimiser (Keerthi et Gilbert, 1988), ce critère étant généralement une fonction quadratique de l'état et de la commande. Dans les travaux de M. Scokaert,

³la convergence globale est une propriété de faisabilité



le problème est soumis à une contrainte terminale nulle, ce qui n'est pas réalisable en pratique. M. Alamir et G. Bornard donnent des conditions suffisantes à la troncature de l'horizon infini. Ils établissent l'existence d'un horizon fini suffisamment grand garantissant la stabilité. Le choix de cet horizon reste un problème ouvert. Par ailleurs, en pratique, les temps de calcul limitent naturellement la taille de l'horizon. E. Courtial s'est intéressée à la stabilité pratique en introduisant l'idée de travailler sur un système qui serait commandable en temps glissant (Courtial, 1996), hypothèse moins sévère que la commandabilité.

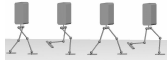
Ces approches sont difficiles à transposer dans notre cas. Tout d'abord, comme nous l'avons dit les hypothèses sont souvent très restrictives et difficilement atteignables en pratique (contraintes terminales, horizon quasi-infini, horizon "suffisamment long", etc.). Ensuite, les raisonnements s'appuient sur des problèmes de poursuite de trajectoire ou régulation difficilement interprétables dans notre cas où points d'équilibre et trajectoires de référence sont des notions difficilement applicables.

2.3.2 L'approche de la stabilité par la faisabilité

Il peut être plus intéressant d'aborder le problème de la stabilité sous un angle différent qui est celui de la faisabilité. Quand le système est non linéaire, le problème de la commande optimale en boucle ouverte est en général non convexe et les algorithmes d'optimisation conventionnels ne donnent qu'une solution locale plutôt que globale. C'est pourquoi il est préférable de penser la stratégie de commande en termes de faisabilité et non en termes d'optimalité.

Les premiers travaux sur la faisabilité concernaient le cas linéaire (Chisci et al., 1996 ; Mayne et al., 2000).

En non linéaire, M. Scokaert et al. proposent deux algorithmes sous-optimaux qui en garantissant la faisabilité assurent une stabilité asymptotique (Scokaert et al., 1999). Le premier algorithme est un schéma classique de MPC avec contrainte de stabilité en imposant $x_{k+N} = 0$. Comme nous l'avons déjà dit, une contrainte terminale égalitaire n'est pas réalisable en pratique. Dans le second cas, un algorithme de type mode dual avec commutation sur une commande localement asymptotiquement stabilisante est utilisé. Sous l'hypothèse qu'il existe un ensemble d'états F pour lesquels il existe une séquence de commande qui ramène l'état dans W , un sous ensemble compact convexe de X , alors la loi de commande MPC mode dual sous-optimale est asymptotiquement stabilisante avec comme région d'attraction l'ensemble F (annexe p.153).



2.3.3 Conclusion

Dans notre cas, nous devons pour l'instant privilégier l'aspect faisabilité puisque les problèmes résolus en terme de stabilité imposent des hypothèses très fortes et le problème que nous posons ne rentre pas dans le cadre conventionnel des techniques prédictives. Il serait éventuellement possible dans le cas qui nous intéresse de considérer également une commande de type mode dual sachant que nous n'aurons pas nécessairement besoin de commuter sur une autre commande puisque nous ne serons pas forcément près d'un point d'équilibre. De plus, cette approche nous semble la plus adaptée d'un point de vue pratique.

Par ailleurs, l'utilisation d'un algorithme d'optimisation tel que FSQP (Feasible Sequential Quadratic Programming) ou une méthode des pénalités avec des poids adaptatifs permettent de garantir à tout instant une solution satisfaisant les contraintes, si elle existe. Ainsi, en présence de perturbations, il est possible de trouver une solution satisfaisant les contraintes qui sont choisies pour garantir l'intégrité du robot et assurer la tâche à accomplir.

2.4 Étude systématique sur un exemple simple

Nous avons réalisé une série de simulations dont l'objectif est d'étudier la sensibilité de notre commande aux différents paramètres (choix du critère, choix de l'horizon, prise en compte des contraintes,...) et au choix du problème d'optimisation. Nous présentons ici une synthèse des résultats obtenus, ces résultats doivent être considérés comme des cas particuliers sans portée générique. Les détails des simulations sont reportés en annexes (pages 158 à 195).

2.4.1 Le pendule inversé

La dynamique du pendule inversé (fig.2.4) s'écrit :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{g}{l} \cdot \sin(x_1) - \frac{f}{ml^2} x_2 + \frac{1}{ml^2} \Gamma \end{cases} \quad (2.2)$$

avec $x_1 = q$ et $x_2 = \dot{q}$.

Les valeurs numériques utilisées sont les suivantes :

Masse	Frottement visqueux	Longueur	constante de gravité
$m=1,0 \text{ kg}$	$f=0,01 \text{ kg.m}^2.s^{-1}$	$l=0,5 \text{ m}$	$g=9,81 \text{ m.s}^{-2}$

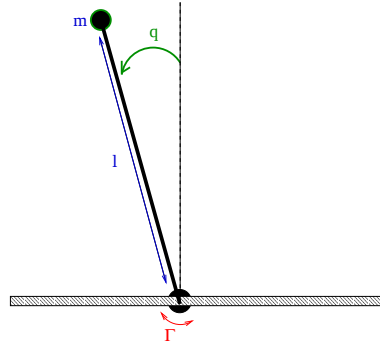


FIG. 2.4 – Pendule inversé

2.4.2 Différents problèmes d'optimisation

Trois types de problèmes ont été traités :

- (i) un problème classique d'optimisation sous contraintes,
- (ii) un problème d'optimisation avec pénalités,
- (iii) un problème avec contrainte de stabilisation.

• Optimisation sous contraintes

Le problème d'optimisation sous contraintes que nous avons traité consiste à résoudre :

$$\min_{U_{k+1}} U_{k+1}^T U_{k+1}$$

soumis à : $g_i \leq 0$

où : U_{k+1} est le vecteur de commande et g_i l'ensemble des contraintes.

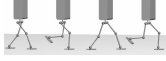
Pour résoudre ce problème, nous avons testé deux algorithmes différents : SQP et FSQP.

• Optimisation avec contraintes pénalités

Il est possible de se ramener à un problème d'optimisation sans contraintes en ajoutant les contraintes dans le critère sous la forme de pénalités :

$$\text{résoudre : } \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} - p_i \sum_i \frac{1}{g_i} \quad (2.3)$$

Le problème est résolu avec un algorithme de type Quasi-Newton qui utilise une mise à jour du Hessien suivant la formule de BFGS.



• Optimisation avec contrainte de stabilité

On peut adjoindre une contrainte de stabilité comprenant un terme quadratique dépendant de l'état terminal $X_{N_p}^T P X_{N_p}$ dans le critère pour forcer le système à atteindre un voisinage du point d'équilibre en un temps fini, de la même façon que dans le cas d'une approche horizon quasi infini (page 102). Il faut alors résoudre :

$$\min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} + X_{N_p}^T P X_{N_p} \quad (2.4)$$

Un retour d'état linéaire équivalent est utilisé pour déterminer la matrice de pénalité P (Chen et Allgower, 1998 ; Scokaert et al., 1999). Les calculs de la matrice P sont disponibles en annexe page 155.

Pour la résolution de ce type de problème, nous avons également utilisé un algorithme de Quasi-Newton.

Note : on peut utiliser ce type de critère avec des contraintes ou des pénalités.

2.4.3 Synthèse des simulations

Les résultats de simulation obtenus dans le cas du pendule inversé (p. 158 à 195) sont présentés de façon synthétique en analysant successivement la façon de poser le problème d'optimisation, le choix des conditions initiales, le choix de l'horizon de prédiction, la limitation du nombre d'itérations de l'algorithme, l'initialisation de la fonction objectif et la robustesse aux erreurs de modélisation.

• Optimisation avec contrainte de stabilité

1. Pour un horizon de prédiction suffisamment grand relativement à l'éloignement initial du système par rapport au point d'équilibre ($q = 0$, $\dot{q} = 0$, $\Gamma = 0$), l'introduction d'un terme portant sur l'état final ($X_{N_p}^T P X_{N_p}$) permet de ramener le système en l'origine. Le calcul de la matrice P (p.155) dispose d'un degré de liberté permettant de jouer sur le temps de réponse (fig. 2.5).

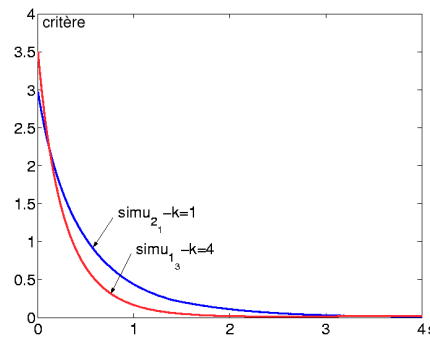
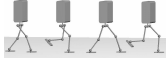


FIG. 2.5 – Influence du paramètre de réglage k sur la fonction objectif



2. Malgré des conditions initiales défavorables (loin du point d'équilibre ou avec des vitesses initiales dirigées dans le sens opposé à l'origine), le système converge vers sa position d'équilibre (fig.2.6), à la condition que l'horizon de prédiction soit suffisamment important.

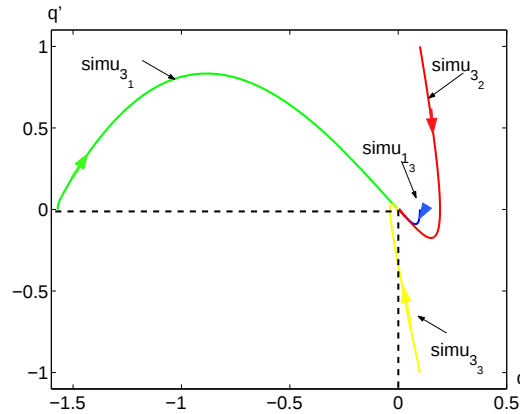


FIG. 2.6 – Influence des conditions initiales sur l'évolution dans le plan de phase

• Optimisation sous contraintes ou avec contraintes de pénalités

1. Si le problème d'optimisation est traité sous la forme d'un problème d'optimisation sous contraintes, les solutions ont tendance à “coller” aux contraintes. Ce phénomène s'atténue avec l'augmentation de l'horizon de prédiction. En utilisant des pénalités au niveau du critère, les solutions tendent au contraire à s'éloigner des contraintes (fig.2.7).
2. On voit par contre, que les temps de calculs sont beaucoup plus importants lorsque les contraintes sont introduites sous la forme de pénalités (fig.2.8).

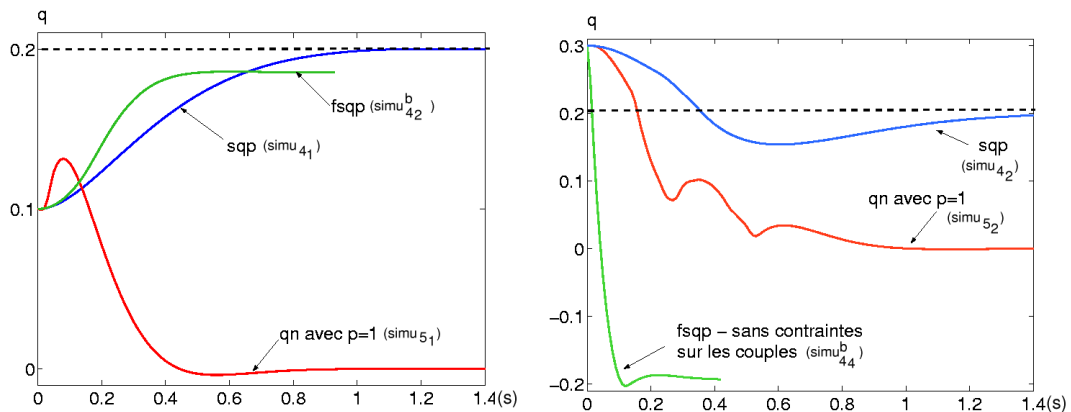


FIG. 2.7 – Évolution de la position articulaire du pendule par rapport à la contrainte $-0.2 \leq q \leq 0.2$ À gauche : CI admissibles - À droite : CI non admissibles

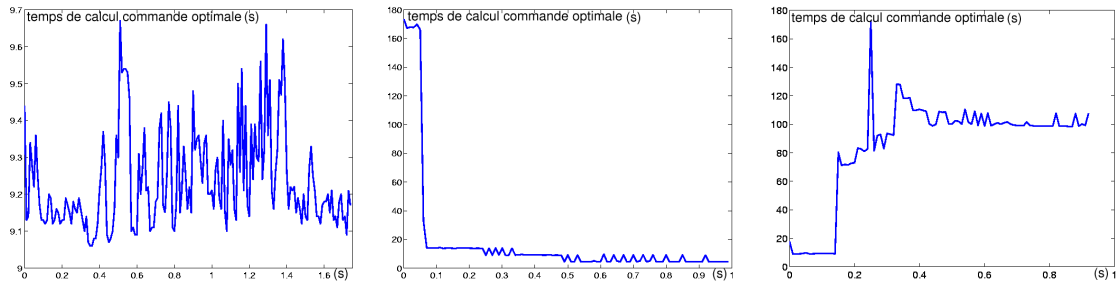
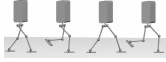


FIG. 2.8 – Influence de la gestion des contraintes sur les temps de calcul. **À gauche** : optimisation sous contraintes SQP - **Au centre** : optimisation sous contraintes FSQP - **À droite** : introduction de pénalités dans le critère QN avec $p=1$

• Faisabilité

1. Si les conditions initiales ne sont pas admissibles (fig.2.7), dans le cas d'un problème sous contraintes ($simu_{4_2}$) ou en utilisant des pénalités ($simu_{5_2}$), le système peut demeurer un certain temps dans une situation où les contraintes sont violées, même si le système tend à satisfaire les contraintes. Avec un algorithme de type FSQP une erreur est générée. Ainsi, lorsque l'une des conditions initiales viole une des contraintes, si les autres contraintes ne permettent pas d'atteindre un problème faisable en un coup, le problème n'a pas de solution. Par exemple si l'on démarre avec une position trop éloignée de la zone admissible, si les couples sont trop fortement limités, il est possible que la zone admissible ne soit pas atteignable. En revanche s'il existe une solution admissible en un coup, celle-ci est calculée ($simub_{4_4}$).
2. Le fait d'augmenter le poids p_i des pénalités permet d'augmenter la vitesse d'éloignement des contraintes ($simu_{6_3}$) (fig.2.9). On voit également qu'il est possible de garantir la faisabilité en augmentant les p_i jusqu'à satisfaire les contraintes ($simu_{5_3}$ et $simu_{5_4}$).

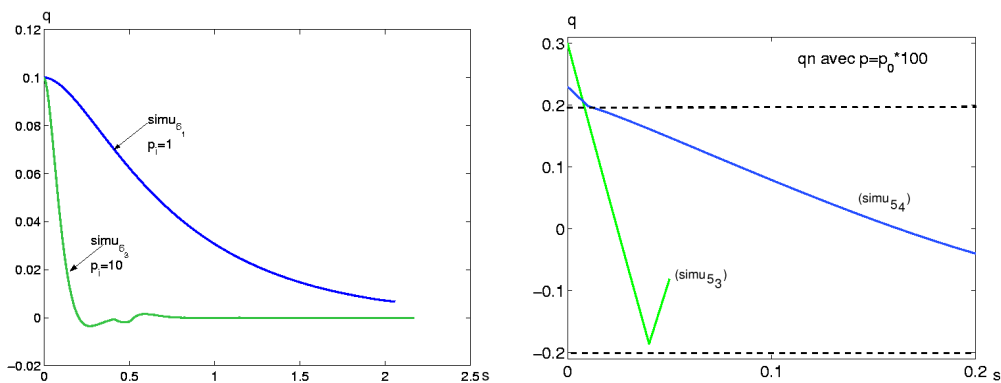
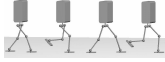


FIG. 2.9 – Influence du choix des poids des pénalités sur la position articulaire



• Influence de l'horizon de prédiction

L'augmentation de l'horizon de prédiction permet d'améliorer la convergence de la fonction objectif (fig. 2.10). Un horizon trop petit ne permet pas de détecter le point d'équilibre : le critère diverge. La convergence est assurée en augmentant l'horizon, et la vitesse de convergence croît avec l'augmentation de cet horizon.

En revanche, il s'agit de trouver un compromis entre convergence et temps de calcul (fig. 2.11) qui augmentent naturellement avec l'horizon !

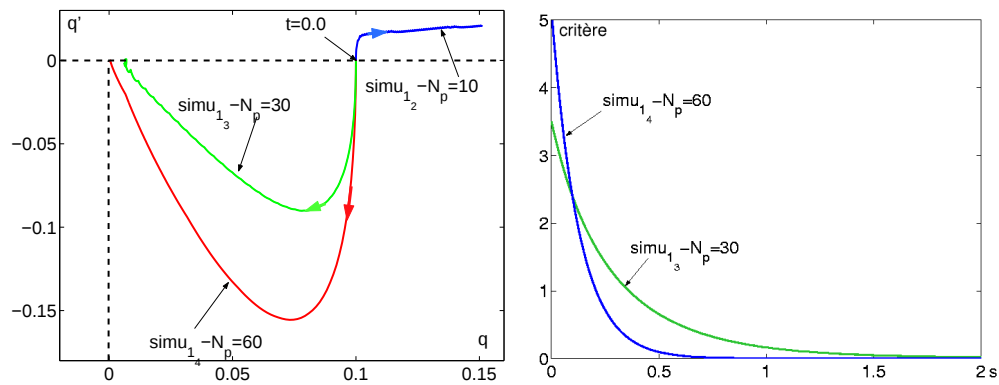


FIG. 2.10 – Influence de l'horizon de prédiction sur la convergence **À gauche** : plan de phase - **À droite** : fonctions objectifs

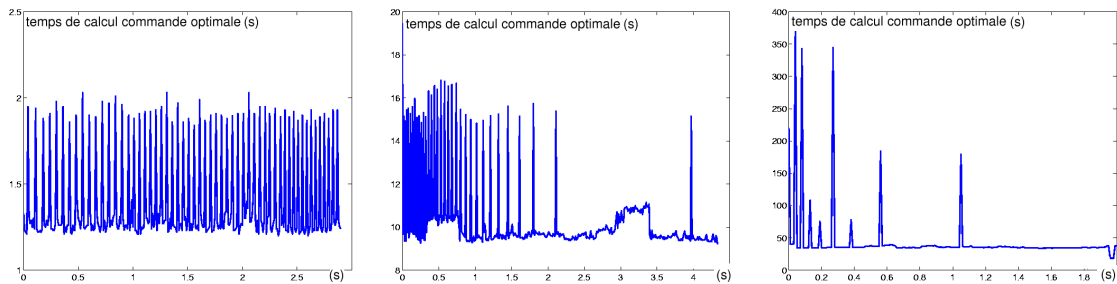


FIG. 2.11 – Influence de l'horizon de prédiction sur les temps de calcul : de gauche à droite $N_p = 10$, $N_p = 30$, $N_p = 60$

• Influence de la limitation du nombre d'itérations

Si le nombre d'itérations de l'algorithme d'optimisation est raisonnablement limité, on assure la convergence du système vers son point d'équilibre (fig. 2.12 et 2.13), en un temps borné (fig. 2.14). En revanche, une restriction trop importante ne permet plus d'assurer la convergence.

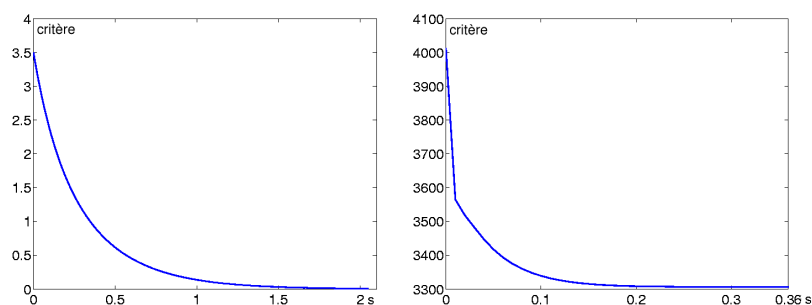
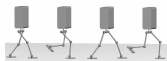


FIG. 2.12 – Fonction objectif - À gauche sans limitation À droite limité à 100 itérations

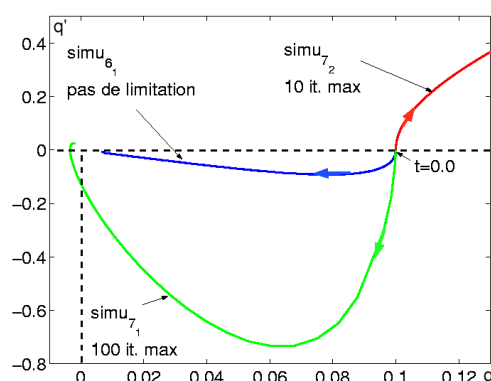


FIG. 2.13 – Comparaison de l'influence de la limitation du nombre d'itérations sur l'évolution du système dans le plan de phase

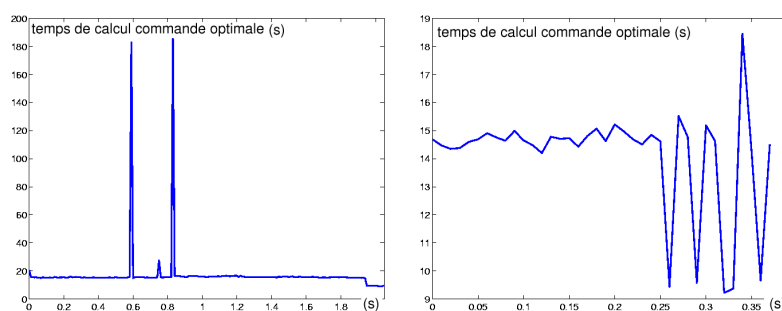
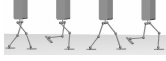


FIG. 2.14 – Comparaison de l'influence de la limitation du nombre d'itérations sur les temps de calcul. De gauche à droite : fonction objectif sans limitation du nombre d'itérations, fonction objectif avec limitation à 100 itérations

• Influence de l'initialisation de l'optimisation

Nous avons testé trois initialisations du problème d'optimisation dans le cas d'un problème avec contrainte de stabilité et pénalités :



- utilisation de l'historique des solutions optimales décalées avec ajout d'un terme égal au dernier couple ($simu_{7_1}$) :
$$U_{k+1}^O = [\Gamma_{k+1|k}^*; \dots \Gamma_{k+N_p-1|k}^*; \Gamma_{k+N_p-1|k}^*]^T$$
- initialisation avec le vecteur nul ($simu_{8_1}$) :
$$U_{k+1}^O = [0; \dots 0]^T$$
- initialisation avec le couple courant ($simu_{8_2}$) :
$$U_{k+1}^O = [\Gamma_k; \dots \Gamma_k; \Gamma_k]^T$$

On voit clairement que l'initialisation joue un rôle important dans la convergence de l'objectif, particulièrement lorsque le nombre d'itérations est limité à 100 comme dans les simulations de la figure 2.15.

Note : L'utilisation de $U_{k+1}^O = [\Gamma_{k+1|k}^*; \dots \Gamma_{k+N_p-1|k}^*; 0]^T$ augmente fortement les temps de calculs.

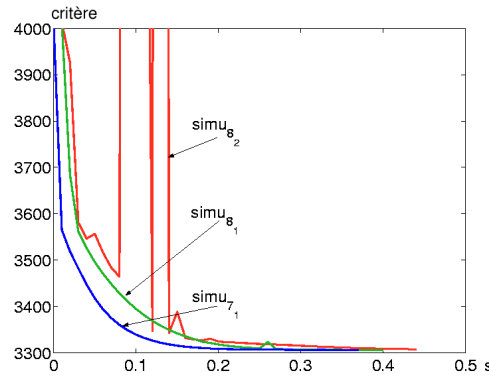


FIG. 2.15 – Influence de l'initialisation de l'optimisation sur la convergence de l'objectif

Robustesse aux erreurs de modèle

Le coefficient de frottement est estimé 10 fois plus petit que sa valeur réelle. Le système converge vers son point d'équilibre (fig. 2.16). Le fait d'augmenter l'horizon de prédiction permet naturellement d'améliorer la robustesse à cette erreur de modèle.

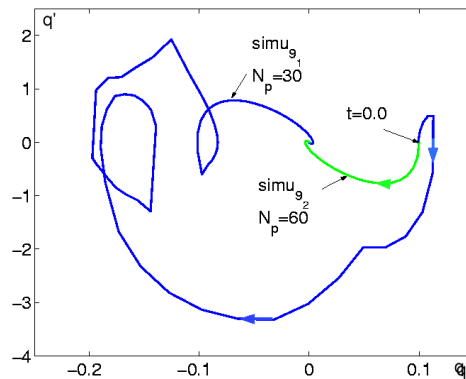
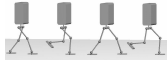


FIG. 2.16 – Influence de l'horizon de prédiction sur la robustesse aux erreurs de modèle



2.4.4 En résumé

Sans être exhaustive, cette synthèse présente les différents paramètres de réglage du problème d'optimisation et leur influence sur le comportement dynamique du système.

On voit clairement sur l'exemple simple du pendule que le réglage de l'algorithme d'optimisation (initialisation, nombre d'itérations) influe fortement sur la convergence du critère ainsi que sur les temps de calcul.

L'horizon de prédiction est également très important. Un horizon trop petit ne permet pas de converger vers la solution optimale. Les horizons de prédiction ont été choisis suffisamment importants pour permettre de comprendre et analyser les approches proposées. Il est évident que lors de la mise en œuvre, mais également de la simulation sur le robot bipède, il ne sera pas possible de prendre des horizons aussi grands. En effet, les temps de calcul augmentent avec l'horizon et deviennent prohibitifs pour l'exécution temps réel.

Le choix de la technique d'optimisation n'est pas non plus sans conséquences, notamment lorsque il s'agit de prendre en compte des contraintes. L'ensemble des trois approches permet de demeurer dans un état admissible si le problème initial est faisable (en l'absence de perturbations). La méthode utilisant des pénalités présente l'avantage de s'éloigner naturellement des contraintes.

Lorsque les conditions initiales sont à l'extérieur du domaine de faisabilité, l'algorithme SQP tend à rejoindre le domaine admissible, mais peut demeurer pendant un temps indéfini dans un état non admissible. FSQP génère quant à lui une erreur s'il ne trouve pas de solution admissible. L'utilisation d'un critère avec pénalités permet de satisfaire la faisabilité en utilisant un poids p_i augmentant dynamiquement jusqu'à obtenir une solution acceptable par rapport aux contraintes. Cette méthode est donc une stratégie garantissant la faisabilité, mais cela, au détriment des temps de calcul.

Les simulations sur la locomotion bipède utilisent un problème d'optimisation sous contraintes car nous souhaitons dans un premier temps démontrer que notre méthode était viable sans nous préoccuper des performances et rapidité de calcul. Nous avons utilisé un algorithme d'optimisation SQP disponible sous Matlab.

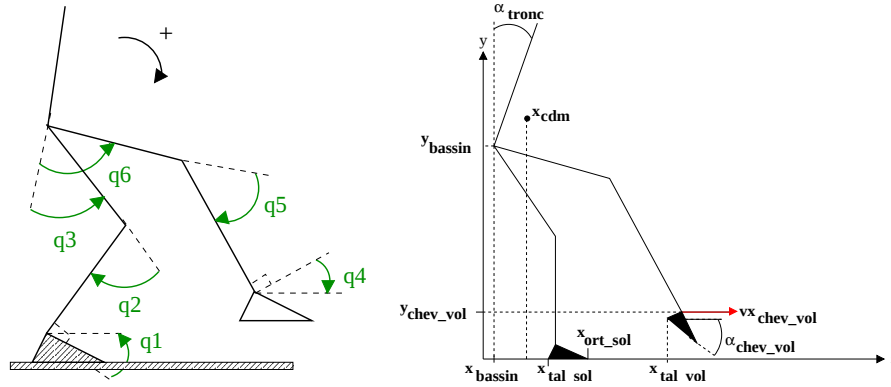
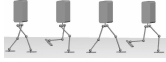


FIG. 2.17 – BIP 2D : À gauche degrés de liberté - À droite contraintes pour la locomotion statique

2.5 Locomotion bipède

La stratégie présentée §.2.2 (p.105) a été appliquée sur plusieurs types de modèles et dans différents contextes. Nous présentons ici des résultats de simulation dans les cas du robot BIP et du robot RABBIT. Le premier exemple concerne la marche statique de BIP sur sol plan et escaliers avec ou sans perturbation. Puis, nous abordons le cas de la marche dynamique de BIP, suivie d'une étude sur l'équilibre postural. Enfin, nous verrons l'application au cas de la marche dynamique du robot RABBIT.

2.5.1 Marche statique : BIP 2D

Nous nous intéressons dans ce premier cas à un mouvement de marche sagittale statiquement stable. Les simulations ont été réalisées à partir d'un modèle du robot BIP 2D avec une masse de 48 kg au niveau du tronc (page 55). La référence est choisie au niveau de la cheville du pied de support, ce dernier étant supposé fixé au niveau du sol. 6 actionneurs sont disponibles (hanches, genoux, chevilles) et 6 degrés de liberté sont considérés (fig.2.17). L'environnement de programmation utilisé est matlab et l'algorithme d'optimisation est de type SQP.

⊕ Critère

Nous avons utilisé un critère composé d'une fonction quadratique en la commande :

$$\min_{U_k} U_k^T U_k$$

avec : $U_k = [\Gamma_{k+1} \dots \Gamma_{k+N_p}]^T$ et $\dim(\Gamma_i) = 6$.

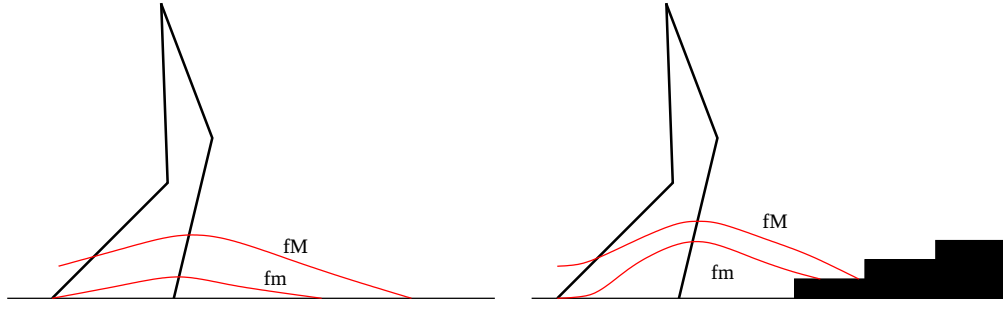
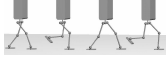


FIG. 2.18 – Ajustement de la zone d'évolution du pied libre

⊕ Récapitulatif des contraintes utilisées

Les contraintes (fig.2.17) utilisées pour cette simulation sont reportées dans le tableau suivant :

★ Dynamique :	Dyn. continue :	$M(q)\ddot{q} + N(q, \dot{q})\dot{q} + G(q) = \Gamma$
	Impact :	$\begin{cases} \dot{q}^+ = \dot{q}^- + M^{-1}(q)C(q)^T \Lambda \\ \Lambda = -(C(q)M(q)^{-1}C(q)^T)^{-1}C(q)\dot{q}^- \end{cases}$
★ Contr. fonctionnelles :	Butées :	$q_{min} \leq q \leq q_{max}$
	Saturation actionneurs :	$\Gamma_{min} \leq \Gamma \leq \Gamma_{max}$
★ Maintien de la posture érigée :		$y_{bassin} \geq h_{bassin_{min}}$
★ Rythme de progression :		$vx_{min} \leq vx_{chev_{vol}} \leq vx_{max}$
★ Pied libre :	Zone limite :	$f_m(x_{chev_{vol}}) \leq y_{chev_{vol}} \leq f_M(x_{chev_{vol}})$
	Semelle parallèle au sol :	$\alpha_{chev_{vol}} = 0$
★ Équilibre statique :		$x_{tal_{sol}} \leq x_{cdm} \leq x_{ort_{sol}}$
★ Autres :	Inclinaison tronc :	$0 \leq \alpha_{tronc} \leq \alpha_{max}$
	Position du bassin :	$x_{tal_{vol}} \leq x_{bassin} \leq x_{ort_{sol}}$

où :

- f_m et f_M sont des fonctions polynomiales qui fixent les bornes de variation de hauteur du pied de vol et de longueur (min/max) des pas (fig.2.18).

Un jeu de contraintes unique est utilisé pour les différentes transitions des pas. Ainsi, les pieds sont maintenus horizontaux et l'impact est supposé survenir au niveau de la projection de la cheville sur le sol. La dernière contrainte, portant sur la position du bassin, facilite le changement de pied de support à l'impact (double support instantané) en assurant que la projection sur le sol du bassin ($\simeq$ centre de masse) se trouve constamment entre les deux pieds.

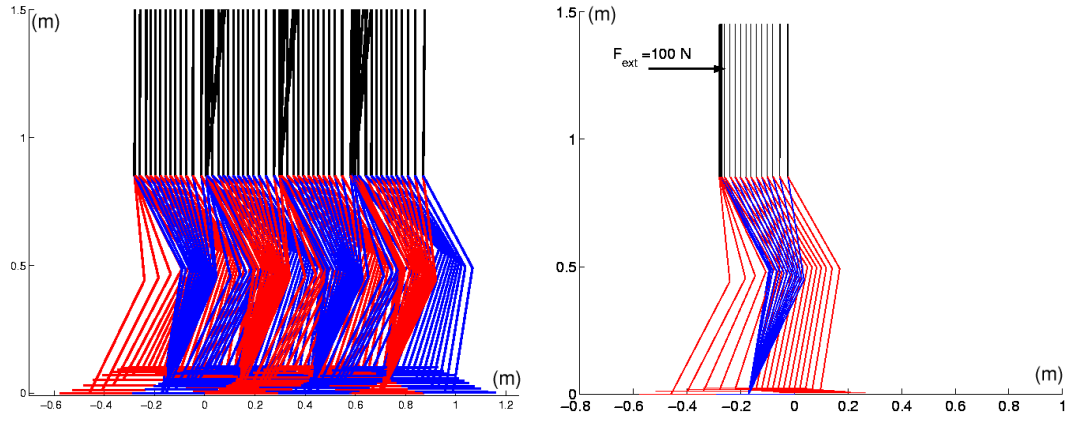
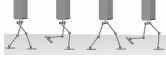


FIG. 2.19 – BIP 2D À gauche : enchaînement de pas sur sol plat - À droite : robustesse à une poussée dans le dos

⊕ Valeurs numériques

$$q_{max} = [0.5; 1.2; 0.1; 0.5; 1.2; 0.1]^T$$

$$\Gamma_{max} = [270; 270; 270; 270; 270; 270]^T$$

$$h_{pelvis_{min}} = 0.85 \text{ m}$$

$$vx_{max} = 0.6 \text{ m.s}^{-1}$$

$$T_e = 0.01 \text{ s (période échantillonnage)}$$

$$q_{min} = [-0.5; 0.2; -1.0; -0.5; 0.2; -1.0]^T \text{ rad}$$

$$\Gamma_{min} = -\Gamma_{max} \text{ Nm}$$

$$\alpha_{max} = 0.1 \text{ rad}$$

$$vx_{min} = 0.2 \text{ m.s}^{-1}$$

$$N_p = 5.T_e \text{ (horizon prédiction)}$$

⊕ Conditions initiales (CI)

$$\begin{cases} q(0) = [-0.16; 0.64; -0.48; -0.46; 0.55; -0.09]^T \\ \dot{q}(0) = [-0.30; 0.39; -0.06; 0.0; 0.64; -0.66]^T \\ \Gamma(0) = [22; -52; 34; -1; 17; 3]^T \end{cases}$$

Note : les CI satisfont les contraintes, N_p est très réduit par rapport aux simulations sur le pendule inversé, la période d'échantillonnage choisie correspond à celle des expérimentations effectuées sur BIP (page 81).

⊕ Marche sur sol plat

La figure 2.19 montre un enchaînement de plus de 4 pas sur sol plat. La vitesse moyenne de déplacement obtenue est de 1.8 km.h^{-1} .

⊕ Robustesse à une poussée dans le dos

Une force $F_{ext} = 100 \text{ N}$ a été appliquée à mi-dos du robot, sans impact, durant la simulation pendant 0.5 s . Le robot parvient à conserver son allure (fig.2.19).

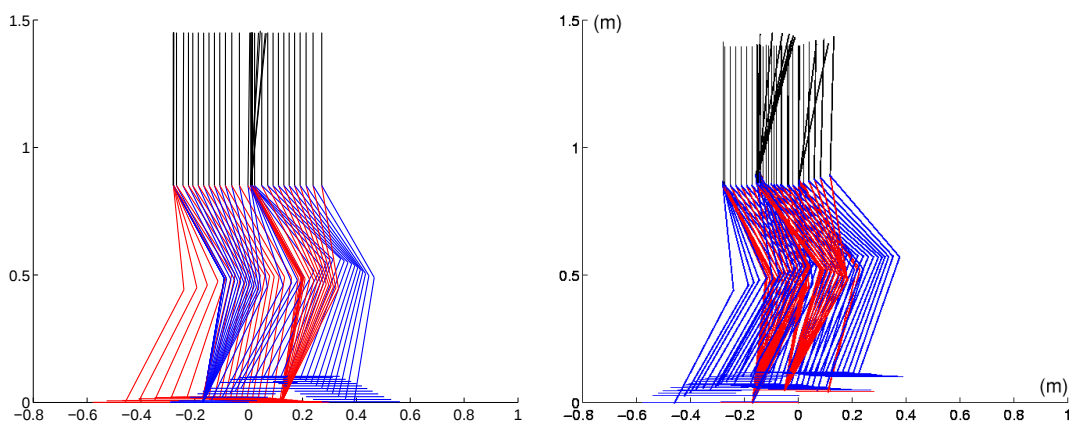
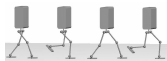


FIG. 2.20 – BIP 2D À gauche : marche avec différentes élévations du pied de vol sur sol plat - À droite : montée d'escalier

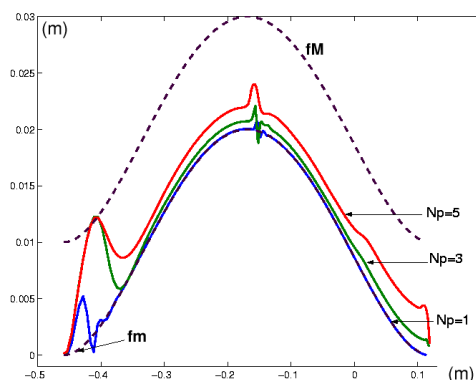


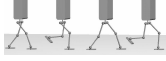
FIG. 2.21 – BIP 2D influence des contraintes selon l'horizon de prédiction

⊕ Montée d'escalier et enjambement d'obstacles

L'ensemble de contraintes utilisées pour la marche sur sol plat permet également d'évoluer dans un environnement à escaliers ou d'enjamber des obstacles.

Il suffit simplement de modifier en ligne le jeu de contraintes portant sur la trajectoire du pied libre incluant les paramètres liés au réglage du pas (variation de hauteur et longueur de l'enjambée) (fig. 2.18 et 2.20).

L'influence du choix de l'horizon de prédiction se visualise aisément sur la figure 2.21. On voit que le fait d'augmenter l'horizon permet d'atténuer la tendance de l'algorithme SQP à “coller” aux contraintes, et lisse les mouvements du robot.



2.5.2 Marche dynamique : BIP 2D

Nous avons réalisé des simulations de marche sagittale dynamique sur le robot BIP 2D. Nous avons utilisé un modèle où la référence est située au niveau des hanches et est libre de se déplacer (page 55). On a ainsi 6 articulations à commander (hanches, genoux, chevilles) et 9 degrés de liberté (fig.2.22). Pour chaque pied, on considère deux points de contact avec le sol : les orteils et le talon.

L'environnement de programmation utilisé est `matlab` et l'algorithme d'optimisation est de type SQP.

⊕ Critère

Nous avons utilisé un critère composé d'une partie quadratique en la commande et d'un terme comportant la distance selon Ox entre la position du bassin et celle du pied de vol :

$$\min_{U_k} U_k^T U_k + p(x_{chevol} - x_{bassin})^2$$

avec : $U_k = [\Gamma_{k+1} \dots \Gamma_{k+N_p}]^T$ et $\dim(\Gamma_i) = 6$, $p = 10^5$ est le poids associé à la contrainte.

⊕ Récapitulatif des contraintes

★ Dynamique :	Continue :	$\begin{cases} \min_{\ddot{q}} \frac{1}{2} \ddot{q}^T M(q) \ddot{q} + \ddot{q}^T (-\Gamma + N(q, \dot{q}) \dot{q} + G(q)) \\ C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0 \\ C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0 \end{cases}$
	Impact :	$\begin{cases} \min_{\dot{q}_+} \frac{1}{2} (\dot{q}_+ - \dot{q}_-)^T M(q) (\dot{q}_+ - \dot{q}_-) \\ C_n(q) \dot{q}_+ = 0 \\ C_t(q) \dot{q}_+ = 0 \end{cases}$
★ Équilibre dynamique :		$ \lambda_t \leq \alpha_f \lambda_n$ (respect du cône de frottement) $y_{tal_{sol}} = 0.0$ $y_{ort_{sol}} = 0.0$
★ Contr. fonct. :	Butées :	$q_{min} \leq q \leq q_{max}$ $\dot{q}_{min} \leq \dot{q} \leq \dot{q}_{max}$
	Saturation actionneurs :	$\Gamma_{min} \leq \Gamma \leq \Gamma_{max}$
★ Posture érigée :		$y \geq h_{bassin_{min}}$
★ Rythme de progression :		$vx_{min} \leq vx_{bassin} \leq vx_{max}$ $vx_{gen_{sol}} \geq 0$
★ Jambe libre :		$-s.vy_{chevol} \leq 0$ $\alpha_{pied_{vol}} = 0$ $y_{chevol} \geq f_m$

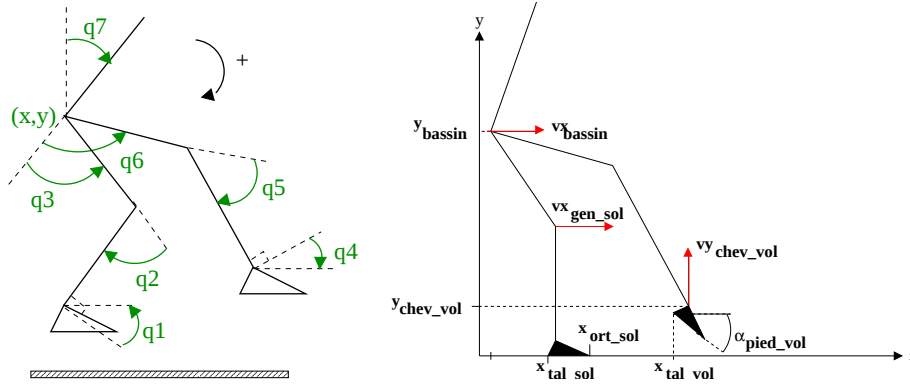
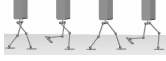


FIG. 2.22 – BIP 2D : À gauche degrés de liberté - À droite contraintes pour la locomotion dynamique

où :

- λ_t et λ_n représentent respectivement l'amplitude des forces tangentielle et normale engendrées par le contact,
 - α_f est le coefficient de frottement.
 - les autres termes sont identiques à ceux utilisés pour la marche statique page 122.
- La contrainte portant sur les forces de support nous permet d'assurer l'équilibre dynamique du robot et de satisfaire aux hypothèses de validité du modèle utilisé pour la dynamique du robot.

L'ajout du terme $(x_{chevol} - x_{bassin})^2$ dans le critère permet au pied de vol de rattrapper le mouvement du bassin (contraint de se déplacer vers l'avant). Lorsque la cheville se retrouve sous le bassin, elle a tendance à s'y maintenir.

L'optimisation quadratique liée à la dynamique complète du robot avec contraintes unilatérales et non glissement ainsi que la gestion des impacts est traitée en utilisant la fonction *quadprog*. L'intégration de la dynamique est réalisée avec la fonction *ode45* couplée à une recherche d'évènements permettant de détecter les instants d'impacts et de décollage des pieds à tout instant en arrêtant l'intégration.

⊕ Valeurs numériques

$$q_{max} = [\pi/2; \pi; \pi; \pi/2; \pi; \pi; 0.01]^T$$

$$\dot{q}_{max} = [2; 2; 2; 2; 2; 2; 2]^T$$

$$\Gamma_{max} = [250; 250; 250; 250; 250; 250]^T$$

$$h_{bassin_{min}} = 0.84 \text{ m}$$

$$vx_{max} = 2$$

$$T_e = 0.01 \text{ s (période échantillonnage)}$$

$$q_{min} = [-\pi/2; 0; -\pi; -\pi/2; 0; -\pi; 0]^T \text{ rad}$$

$$\dot{q}_{min} = -\dot{q}_{max} \text{ rad}$$

$$\Gamma_{min} = -\Gamma_{max} \text{ Nm}$$

$$\alpha_f = 0.6$$

$$vx_{min} = 1 \text{ m.s}^{-1}$$

$$N_p = 3.T_e \text{ (horizon prédiction)}$$

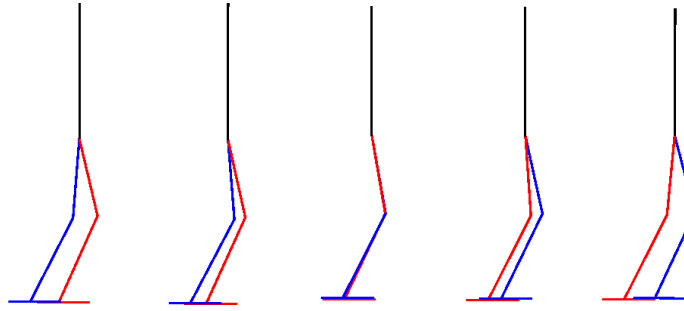
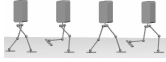


FIG. 2.23 – BIP2D - allure d'un pas dynamique

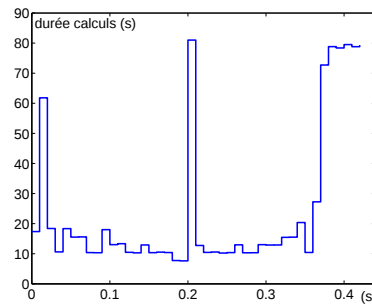


FIG. 2.24 – BIP2D - évolution du temps de calcul de la commande

⊕ **Conditions initiales**

$$\begin{cases} q(0) = [-0.45; 0.7; -0.25; -0.505; 0.42; 0.085; 0.0]^T \\ \dot{q}(0) = [0.0; 0.0; 0.0; 0.0; 0.2; -1; 0.0]^T \\ \Gamma(0) \text{ compensation de la gravité} \end{cases}$$

⊕ **Marche sur sol plat**

Nous présentons figure 2.23 l'allure d'un pas. L'évolution des temps de calcul de l'optimisation est présenté sur la figure 2.24. Le pas obtenu a une longueur de 33 *cm* et une hauteur de 7 *mm*. Ce pas est réalisé en 0.44 *s*. L'allure des couples et des forces de support sont disponibles figure 2.25.

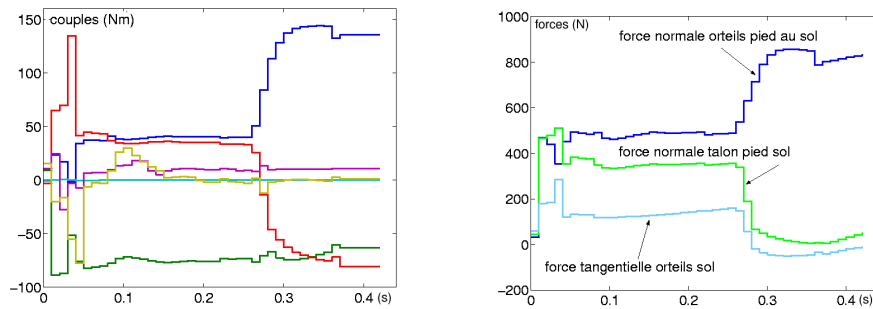
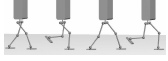


FIG. 2.25 – BIP2D - À gauche : évolution des couples articulaires - À droite : évolution des forces de support



Nous avons arrêté la simulation juste avant l'impact du pied de vol. Des problèmes numériques liés aux contacts multiples nous ont en effet posé des difficultés.

2.5.3 Équilibre : BIP 2D

Le robot doit être capable de se mouvoir mais également se maintenir dans une posture donnée. Le modèle est identique à celui utilisé pour la marche dynamique, nous avons rajouté un actionnement au niveau du tronc afin de se rapprocher du prototype BIP 3D. On a ainsi 7 articulations à commander (hanches, genoux, chevilles) et 9 degrés de liberté.

L'environnement de programmation utilisé est `matlab` et l'algorithme d'optimisation est de type SQP.

Nous avons réalisé une série de simulations dans lesquelles le robot a pour mission de maintenir son équilibre malgré l'application d'une poussée dans son dos. Cette force est supposée mesurée par les capteurs du robot. L'objectif du robot est de rester debout, à l'arrêt, dans sa position initiale.

⊕ Critère

Comme nous l'avons expliqué dans la première partie, chez l'homme, différents critères sont utilisés suivant l'objectif poursuivi. Nous proposons le critère suivant dans le but de maintenir la posture initiale :

$$\frac{1}{2} \sum_{j=k+1}^{j=k+N_p} (q - q_{ini})^T (q - q_{ini}) \quad (2.5)$$

⊕ Contraintes

★ Dynamique : Continue :	{	$\min_{\ddot{q}} \frac{1}{2} \ddot{q}^T M(q) \ddot{q} + \ddot{q}^T (-\Gamma + N(q, \dot{q}) \dot{q} + G(q))$
		$C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0$
		$C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0$
Impact :	{	$\min_{\dot{q}_+} \frac{1}{2} (\dot{q}_+ - \dot{q}_-)^T M(q) (\dot{q}_+ - \dot{q}_-)$
		$C_n(q) \dot{q}_+ = 0$
		$C_t(q) \dot{q}_+ = 0$
★ Équilibre dynamique :		$ \lambda_t \leq \alpha_f \lambda_n$
★ Contr. fonct. : Butées :		$q_{min} \leq q \leq q_{max}$
Saturation actionneurs :		$\Gamma_{min} \leq \Gamma \leq \Gamma_{max}$
★ Posture érigée :		$y \geq h_{bassin_{min}}$

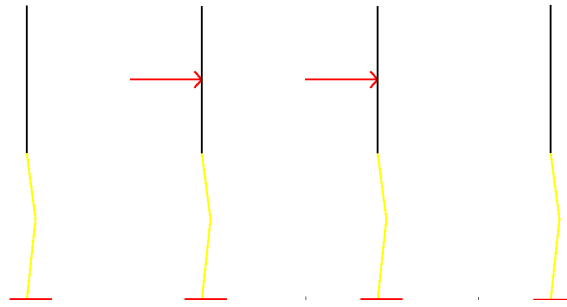
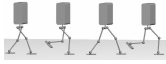


FIG. 2.26 – BIP 2D : application d'une force de 100 N pendant $0,25\text{ s}$ à mi-dos.

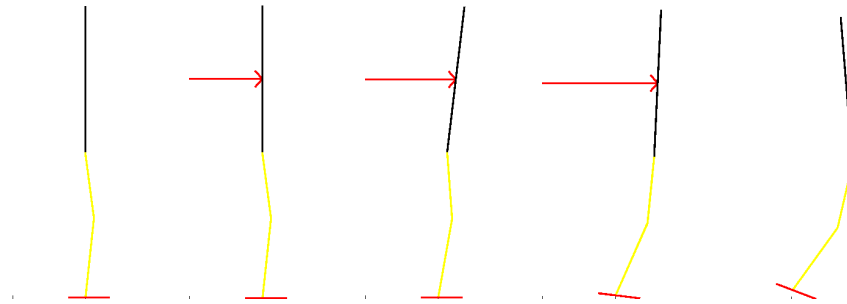


FIG. 2.27 – BIP 2D : application d'une force de 200 N pendant $0,5\text{ s}$ à mi-dos.

⊕ Poussée continue sans impact

- ★ Une force de 100 N est appliquée pendant $0,25\text{ s}$

Le robot est capable de résister à cette force et de maintenir sa position de départ (fig.2.26).

- ★ Une force de 200 N est appliquée pendant $0,5\text{ s}$

Le robot est incapable de résister à cette force, mais lutte jusqu'à la chute (fig.2.27).

Si l'on ne met pas une notion de marge de sécurité au delà de laquelle il faut réagir le robot va tenter de lutter contre la force qui lui est appliquée jusqu'à ne plus pouvoir le faire. Le robot ayant un horizon d'anticipation réduit, il ne peut donc pas élaborer une stratégie d'avancée d'un pied. D'ailleurs, le critère ne tend pas vers une telle solution. Il faudra donc modifier le critère en fonction du danger auquel le système est soumis. C'est aussi le cas chez l'homme qui évalue la perturbation subie et choisit de lutter pour maintenir sa posture ou au contraire de faire un pas en faveur du maintien de son équilibre !

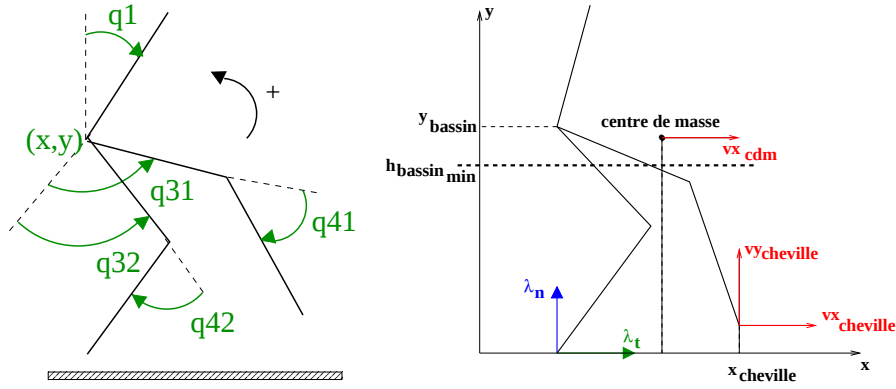
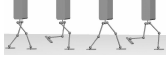


FIG. 2.28 – RABBIT : À gauche degrés de liberté - À droite contraintes

2.5.4 Marche dynamique sous-actionnée : RABBIT

Nous avons également effectué des simulations de marche dynamique sur un modèle du robot RABBIT.

⊕ Critère

Nous avons utilisé un critère composé d'une fonction quadratique en la commande :

$$\min_{U_k} U_k^T U_k$$

avec : $U_k = [\Gamma_{k+1} \dots \Gamma_{k+N_p}]^T$ et $\dim(\Gamma_i) = 4$.

⊕ Récapitulatif des contraintes

★ Dynamique :	Continue :	$\begin{cases} \min_{\ddot{q}} \frac{1}{2} \ddot{q}^T M(q) \ddot{q} + \ddot{q}^T (-\Gamma + N(q, \dot{q}) \dot{q} + G(q)) \\ C_n(q) \ddot{q} + s_n(q, \dot{q}) \geq 0 \\ C_t(q) \ddot{q} + s_t(q, \dot{q}) = 0 \end{cases}$
	Impact :	$\begin{cases} \min_{\dot{q}_+} \frac{1}{2} (\dot{q}_+ - \dot{q}_-)^T M(q) (\dot{q}_+ - \dot{q}_-) \\ C_n(q) \dot{q}_+ = 0 \\ C_t(q) \dot{q}_+ = 0 \end{cases}$
★ Équilibre dynamique :		$ \lambda_t \leq \alpha_f \lambda_n$
★ Contr. fonct. :	Butées :	$q_{min} \leq q \leq q_{max}$
	Saturation actionneurs :	$\Gamma_{min} \leq \Gamma \leq \Gamma_{max}$
★ Posture érigée :		$y_{bassin} \geq h_{bassin_{min}}$
★ Rythme de progression :		$vx_{cdm_{min}} \leq vx_{cdm} \leq vx_{cdm_{max}}$
★ Pied libre :		$vx_{chev_{vol}} \geq 0$
		$s.vy_{chev_{vol}} \geq 0$

avec : $s = 1$ tant que la cheville libre n'est pas passée devant la cheville au sol, puis $s = -1$.

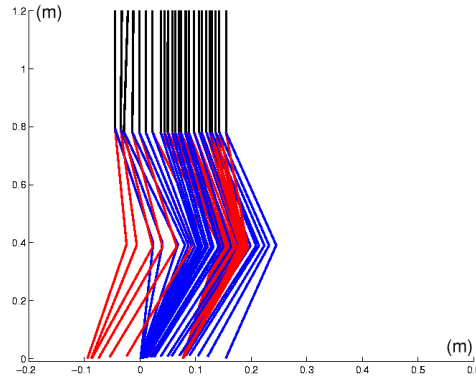


FIG. 2.29 – RABBIT : marche sur sol plan

⊕ **Conditions initiales**

$$\begin{cases} q(0) = [0; -2.97; -0.23; -3.1; -0.23; -0.04]^T \\ \dot{q}(0) = [0; 0; 0; 0; 0; 0.6; 0.05]^T \\ \Gamma(0) = [0; 0; 0; 0]^T \end{cases}$$

⊕ **Valeurs numériques**

$$\begin{aligned} q_{max} &= [0.0; -\frac{\pi}{2}; -0.1; -\frac{\pi}{2}; -0.1]^T & q_{min} &= [-0.1; -\frac{3\pi}{2}; -\frac{\pi}{2}; -\frac{3\pi}{2}; -\frac{\pi}{2}]^T \text{ rad} \\ \Gamma_{max} &= [150; 150; 150; 150]^T & \Gamma_{min} &= -\Gamma_{max} \text{ Nm} \\ h_{bassin_{min}} &= 0.78 \text{ m} & \alpha_f &= 0.6 \\ vx_{max} &= 0.6 \text{ m.s}^{-1} & vx_{min} &= 0.2 \text{ m.s}^{-1} \\ T_e &= 0.01 \text{ s (période échantillonnage)} & N_p &= 3.T_e \text{ (horizon prédiction)} \end{aligned}$$

⊕ **Marche sur sol plat**

La figure 2.29 présente l'allure lors de l'enchaînement de deux pas. La figure 2.30 montre les couples articulaires nécessaires au déplacement pendant ce déplacement, ainsi que l'allure des forces de support. Les forces normales et tangentielles sont tout à fait satisfaisantes en amplitude et vérifient bien la contrainte de non-glissement. Le double appui correspond à 7,5% du premier pas et 9% du second pas. L'évolution des temps de calcul de la commande optimale à chaque pas d'échantillonnage est visible sur la figure 2.31. Le temps moyen du calcul est de 3,5 s, la valeur minimale étant de 0,82 s et la valeur maximale 49 s. Les valeurs maximales correspondent aux changements de pied de support. Les simulations ont été effectuées sur un micro-ordinateur PC standard, sans optimisation ni simplification des calculs.

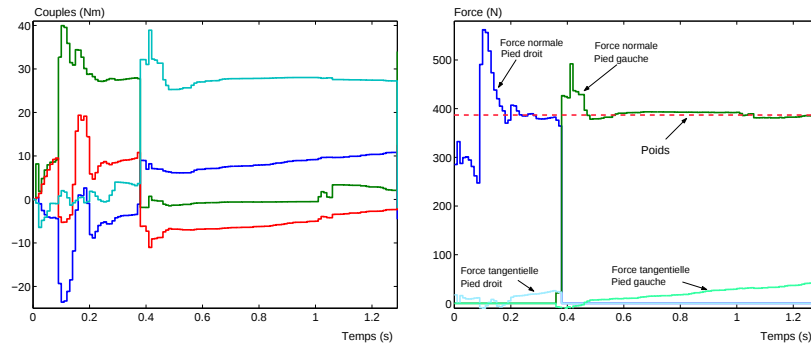
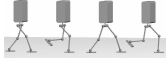


FIG. 2.30 – RABBIT : À gauche couples articulaires - À droite forces de contact

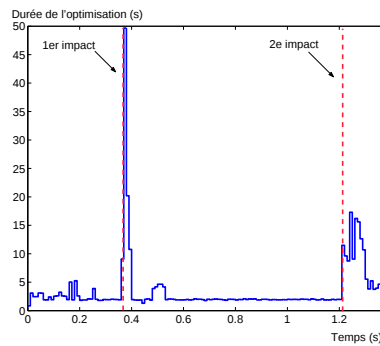
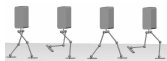


FIG. 2.31 – RABBIT : évolution du temps de calcul de la commande optimale

2.6 Conclusion

La stratégie de commande à horizon fuyant proposée pour la locomotion bipède est simple dans son concept et sa mise en œuvre. Les difficultés principales résident dans le choix judicieux des contraintes et le choix de la technique d'optimisation. L'analyse du pendule inversé nous a permis de mettre en évidence l'influence du choix du problème d'optimisation et du choix des arguments de l'algorithme sur le comportement dynamique du système, sa stabilité et sa faisabilité. L'application au cas des robots bipèdes met en évidence la difficulté quant au choix de contraintes et aux choix de l'horizon de prédiction vis-à-vis de temps de calculs acceptables en vue d'une implantation temps réel. Mais, notre objectif était avant tout de démontrer que notre stratégie permettait effectivement de générer des mouvements de marche bipède sans trajectoire de référence. Nous avons mis en évidence l'utilisabilité pour des allures statiquement stables mais également dynamiquement stables. Nous avons aussi montré que cette technique pouvait se généraliser à différents types de robots bipèdes, sans modification de la structure de l'algorithme.



Chapitre 3

Vers la mise en œuvre de la commande

3.1 Introduction

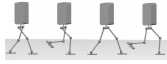
Dans l'élaboration de la stratégie de commande, il a été fait abstraction des problèmes de temps de calcul afin de mettre la priorité sur la définition de la commande et l'évaluation de ses potentialités. Pour que notre méthode soit utilisable en pratique, les temps de calcul devront être améliorés. Plusieurs pistes étant possibles, nous proposons ici quelques idées.

3.2 Choix du problème d'optimisation

Nous avons pu observer dans le cas du pendule inversé que le choix de l'algorithme d'optimisation influence de façon importante les temps de calculs. Une étude approfondie devra être menée en ce sens afin de trouver un compromis entre performance et rapidité de la recherche de solution optimale ou sous optimale mais satisfaisant les contraintes. Les techniques d'optimisation par arithmétiques d'intervalles sont une piste à explorer pour résoudre ce problème de la locomotion exprimé à l'aide de contraintes c'est-à-dire d'intervalles possibles d'évolution, dans la mesure où la précision n'est pas une performance déterminante (Jaulin et al., 2001 ; Merlet, 2000).

3.3 Auto-apprentissage

Si l'homme ne fait pas de poursuite de trajectoire, il a néanmoins en mémoire des structures de mouvements apprises au fur et à mesure. Ainsi, il est tout à fait possible d'imaginer de conserver l'historique des couples calculés au cours d'un pas du robot et d'initialiser l'optimisation des pas suivants avec ces couples. En effet, en l'absence de perturbations, les pas seront sensiblement identiques au cours du déplacement. Une seconde idée est d'associer notre approche à celle proposée par P.-B. Wieber qui utilise des familles de trajectoires de référence pré-calculées (Wieber, 2000). Le robot posséderait ainsi quelques trajectoires de base dont il pourrait s'inspirer pour



affronter les situations déjà rencontrées, pour initialiser l'optimisation par exemple, ou changer de stratégie pour maintenir sa stabilité en cas de perturbations.

3.4 Limitation des itérations

Pour que notre méthode soit implémentable, la durée du calcul de la commande à appliquer doit être bornée. Le nombre d'itérations de l'algorithme d'optimisation doit être limité, ainsi que leur durée. Nous avons vu sur le cas du pendule inversé, qu'une limitation raisonnable du nombre d'itérations permettait en effet de réduire fortement la durée de l'optimisation.

3.5 Calcul sur des périodes supérieures à T_e

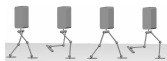
Il est possible d'envisager une approche multi-fréquence où le calcul de la commande est effectué à une fréquence inférieure à la fréquence d'échantillonnage. Un réajustement peut alors être réalisé à des points de coïncidence (Richalet, 1993).

3.6 Simplification du modèle

Le modèle utilisé dans la boucle d'optimisation peut être simplifié. Par exemple, si les heurts avec le sol sont de faible amplitude, les calculs d'impact ne seront pas indispensables. Par ailleurs, nous avons utilisé un calcul simplifié qui donne des résultats satisfaisants pour l'intégration de la dynamique dans le cas des bipèdes, ce qui nous a permis un gain de temps important pour des écarts entre modèle et réalité faibles.

3.7 Programmation en “dur”

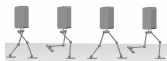
Il est envisageable de coder l'algorithme en langage machine, ce qui améliorerait nettement les temps de calcul. Les techniques de calcul parallèle (grappes de PC) et le passage d'ORCCAD en multi-fréquence ouvrent également des perspectives très intéressantes. De plus, les performances des calculateurs sont en constante augmentation du fait de l'évolution extrêmement rapide des technologies liées aux microprocesseurs (loi de Moore). Nous avons évalué les temps de calcul sur deux générations de machines (pentium3 et pentium4) et observé un rapport allant du simple au double !

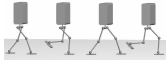


Conclusion

La contribution essentielle de cette partie de notre travail a été de proposer une nouvelle approche ou stratégie pour le contrôle de la marche sans trajectoire de référence comme ce peut être le cas chez l'homme. L'idée d'utiliser une commande optimale avec horizon de prédiction nécessite de déterminer un jeu de contraintes s'adaptant à toutes les situations qui pourraient se présenter. Le but n'est pas de modifier les paramètres en fonction du déroulement du déplacement du robot, mais de proposer des contraintes qui évoluent et s'adaptent à l'environnement et aux perturbations. Les simulations effectuées sur la marche bipède permettent de valider notre approche et son utilisabilité. L'application de la méthode aux cas des robots BIP et Rabbit met en avant la portabilité de l'algorithme. Malgré les points qui restent encore à approfondir pour envisager la mise en œuvre sur un prototype bipède (temps de calcul, faisabilité, robustesse...), les résultats de simulation offrent des perspectives intéressantes.

Les travaux pour porter cette stratégie sur un prototype sont en cours (F. Lydoire LIRMM). Les perspectives plus lointaines concernent la marche dynamique 3D de BIP et l'utilisation de capteurs extéroceptifs pour mettre à jour dynamiquement les contraintes. Il nous semble en outre que la robotique bipède n'aura d'avenir que si l'on apprend aux machines à gérer les chutes qui sont inévitables en locomotion bipède.



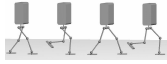


Pour finir...

La première contribution de cette thèse a été de mener une **réflexion sur la locomotion bipède et les interactions entre les domaines des sciences du vivant et de la robotique**. Nous avons développé principalement la piste de l'exploitation des données humaines pour la robotique anthropomorphe. Contrairement à ce qui se fait habituellement, nous n'avons pas cherché à imiter la marche humaine, mais plutôt à la comprendre afin de souligner les aspects importants du contrôle du mouvement. Ce positionnement a servi de fil conducteur à cette thèse volontairement tournée vers la simulation et l'expérimentation.

Comme nous l'avons souligné à plusieurs reprises au cours de ce mémoire, deux approches sont actuellement en développement au sein du projet BIP : le suivi de trajectoire choisie en ligne parmi une famille de mouvements précalculés et la stratégie proposée en quatrième partie de ce mémoire et qui repose sur une commande à horizon fuyant sans trajectoire de référence.

Dans le cadre de la première approche plus classiquement utilisée, nous avons proposé des **modèles pour différentes versions du robot BIP 2D et pour le robot RABBIT**. Nous avons mis en place une technique pour la **génération de marches statiques planes** pour le robot BIP 2D à partir des travaux de P.-B. Wieber qui avait proposé au cours de sa thèse une méthode pour la génération de mouvements posturaux du robot BIP 3D en appui sur un pied à partir d'une interpolation entre postures clefs intermédiaires. Sur cette base nous avons également développé une nouvelle approche pour la **génération de marches statiques 3D** en s'affranchissant de la définition de postures intermédiaires. Notre stratégie utilise une optimisation sous contraintes pour déterminer des marches symétriques à partir d'une posture de départ donnée. Une autre contribution a été le **développement et la mise en œuvre d'un contrôleur pour le suivi de mouvements de référence** pour le robot BIP dans ces deux versions, sur la base du logiciel ORCCAD. Ce travail a permis de mettre en place deux plates-formes expérimentales et de réaliser les premiers tests sur les robots BIP 2D et BIP 3D. Un résultat important a été l'**expérimentation de mouvements posturaux 3D de grande amplitude**, à partir des trajectoires générées par la méthode de P.-B. Wieber. À notre connaissance ce type de réalisation était une première mondiale. Sur la base des allures de marche générées par notre technique, nous sommes parvenus à l'**expérimentation**



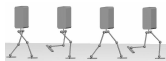
de marches statiques 2D. Ces deux résultats ont fait l'objet d'un film pour le pavillon français de l'exposition universelle de Hanovre en 2000. Une version du contrôleur a été fournie au LMS de Poitiers pour leur prototype BIP 3D. Nous avons ensuite procédé à une **campagne expérimentale d'allures de marche sur le robot BIP 2D.** L'exploitation des données mesurées à cette occasion a conduit à proposer une **analyse comparative des démarches**, en prenant pour référence la marche humaine normale. L'évaluation de l'influence des paramètres des pas réalisés nous a permis de mettre en évidence un certain nombre de principes de base pour la définition des déplacements.

Enfin, la dernière contribution originale de ce travail de thèse a été la **proposition d'une stratégie de contrôle de la locomotion sans trajectoire de référence** à partir de l'observation des comportements humains. Nous proposons une commande basée sur une approche commande à horizon fuyant permettant de générer des mouvements non prédéfinis tout en anticipant le comportement futur du système. Afin de mieux les comprendre, nous avons procédé à une **analyse systématique des approches de commande prédictive sur un exemple académique** : le pendule inversé. L'**application de la stratégie aux marches statiques et dynamiques sur les robots BIP 2D et RABBIT** a conduit à des résultats de simulation encourageants. Le point faible de notre approche concerne les temps de calcul, mais nous faisons quelques **propositions pour l'implantation temps réel.**

Les perspectives liées à ce travail sont multiples. À court terme, ce sera un approfondissement des aspects liés à la locomotion humaine qui sera mené au cours d'un post-doc dans le laboratoire de neurosciences DPM (Développement et Pathologie du Mouvement) à Marseille. Il s'agira de mieux comprendre les origines du mouvement et de les utiliser ultérieurement pour le contrôle de robots bipèdes, mais également chez l'homme, dans le cadre de la restauration du mouvement chez les patients paraplégiques par électro-stimulation fonctionnelle. Cette approche est séduisante dans la mesure où la formulation de la commande à horizon fuyant permet de prendre en compte naturellement les aspects liés notamment à la fatigue, phénomène le plus critique pour l'autonomie de ces patients.

La mise en œuvre expérimentale de notre stratégie de commande sera par ailleurs poursuivie par F. Lydoire au cours de sa thèse au LIRMM (Montpellier).

L'ensemble de ce travail a fait l'objet de plusieurs publications dont le détail est proposé page 139.



Publications de l'auteur

- ★ C.Azevedo, P.Poignet, B.Espiau (2002), Moving Horizon Control for Biped Robots without Reference Trajectory, ICRA¹
- ★ C.Azevedo, P.Poignet, B.Espiau (2002), On line Optimal Control for Biped Robots, IFAC²
- ★ F.Lydoire, C.Azevedo, B.Espiau, P.Poignet (2002), 3D parametrized gaits for biped walking, CLAWAR³
- ★ C.Azevedo, P.Poignet (2002), Commande prédictive pour la marche d'un robot bipède sous actionné, CIFA⁴
- ★ C.Azevedo, N.Andreff, S.Arias, B.Espiau (2002), Experimental BIPedal walking, ISER⁵
- ★ C.Azevedo, N.Andreff, S.Arias (2002), BIPedal walking : from gait design to experimental analysis, Mechatronics⁶ (article soumis)
- ★ C.Azevedo (2001), On the Interaction between the Human and the Robot in Bipedal Walking, CLAWAR
- ★ C.Azevedo, N.Andreff (2000), Étude expérimentale des premières démarches du robot BIP2000, rapport de recherche INRIA 4017
- ★ C.Azevedo, R.Pissard-Gibollet(2000), Le contrôleur du robot BIP2000, rapport technique INRIA 0249
- ★ C.Azevedo (2000), Algorithmes et architecture de la commande du Robot Bipède Anthropomorphe BIP2000, JJCR⁷
- ★ C.Azevedo and the BIP team (2000), Control Algorithms and Architecture of the Anthropomorphic Biped Robot Bip2000, CLAWAR

¹ICRA : IEEE International Conference on Robotics and Automation

²IFAC : International Federation of Automatic Control World Congress

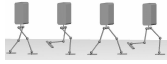
³CLAWAR : Climbing And Walking Robots Conference

⁴CIFA : Conférence Internationale Francophone d'Automatique

⁵ISER : International Symposium on Experimental Robotics

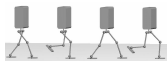
⁶Mechatronics : journal Elsevier Science

⁷JJCR : Journées des Jeunes Chercheurs en Robotique

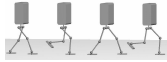


Références

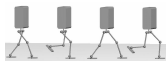
- Abernethy, B., Hanna, A., et Plooy, A. : 2002, *The attentional demands of preferred and non-preferred gait patterns*, Vol. 15, Gait and Posture
- Adams, B., Breazeal, C., Brooks, R., et Scassellati, B. : 2000, *Humanoid Robots : A New Kind of Tool*, Vol. 15-(4), IEEE Intelligent Systems and Their Applications : Special Issue on Humanoid Robotics
- Alamir, M. et Bornard, G. : 1994, *On the Stability of Receding Horizon Control of Nonlinear Discrete-Time Systems*, Vol. 23, Syst. Control Letter
- Alamir, M. et Bornard, G. : 1995, *Stability of a Truncated Infinite Constrained Receding Horizon Scheme : the General Discrete Nonlinear Case*, Vol. 31-(9), Automatica
- Allard, P. et Blanchi, J. : 1996, *Analyse du mouvement humain par la biomécanique*, Décarie Éditeur, Québec, Canada
- Allgöwer, F., Badgwell, T., Qion, J., Rawlings, J., et Wright, S. : 1999, *Nonlinear Predictive Control and Moving Horizon Estimation - an introduction overview*, European Control Conference ECC
- Armstrong-Hélouvry, B. : 1991, *Control of Machines with Friction*, Kluwer Academic Publishers
- Asimo : 2000, Honda, <http://www.honda.co.jp/ASIMO/>
- Assaïante, C. : 2000, *Construction du répertoire des référentiels posturaux : maturation et adaptation au cours de l'enfance*, HDR en Neurosciences, Université de la Méditerranée, Faculté des Sciences de Luminy
- Assaïante, C. : 2001, *Bipédie humaine : la conquête de l'équilibre dynamique au cours de l'ontogénèse*, Vol. 4, Primatologie
- Azevedo, C. : 2001, *On the Interaction between the Human and the Robot in Bipedal Walking*, Climbing And Walking Robots
- Azevedo, C. et al. : 2000, *Control Architecture and Algorithms of the Anthropomorphic Biped Bip2000*, Climbing And Walking Robots
- Azevedo, C. et Andreff, N. : 2000, *Étude expérimentale des premières démarches du robot BIP2000*, rapport de recherche INRIA 4017
- Azevedo, C., Andreff, N., et Arias, S. : 2002a, *BIPedal walking : from gait design to experimental analysis*, Mechatronics (article soumis)
- Azevedo, C., Andreff, N., Arias, S., et B.Espiau : 2002b, *Experimental BIPedal walking*, International Symposium on Experimental Robotics
- Azevedo, C. et Pissard-Gibollet, R. : 2000, *Le contrôleur du robot BIP2000*, rapport



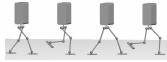
- technique INRIA 0249
- Azevedo, C. et Poignet, P. : 2002, *Commande prédictive pour la marche d'un robot bipède sous actionné*, Conférence Internationale Francophone d'Automatique
- Azevedo, C., Poignet, P., et Espiau, B. : 2002c, *Moving Horizon Control for Biped Robots without Reference Trajectory*, ICRA'02
- Azevedo, C., Poignet, P., et Espiau, B. : 2002d, *On line Optimal Control for Biped Robots*, IFAC'02
- Basseville, M. et Nikiforov, I. : 1993, *Detection of Abrupt Changes : Theory and Applications*, P T R Prentice Hall, Englewood Cliffs, New Jersey
- Berry, G. et Gonthier, G. : 1992, *The Esterel Synchronous Programming Language : Design, Semantics, Implementation*, Vol. 19-(2), Science Of Comp. Prog.
- Berthoz, A. : 1997, *Le sens du mouvement*, Éditions Odile jacob
- Bischoff, R. : 2001, *System Reliability and Safety Concepts of the Humanoid Service Robot HERMES*, Vol. I-2, Proc. IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environments
- Bitmead, R., Gevers, M., Ptersen, I., et Kaye, R. : 1990, *Adaptative Optimal Control : The thinking man's GPC*, Prentice Hall
- Boucher, P. et Dumur, D. : 1996, *La Commande Prédictive*, Éd. Technip.
- Brogliato, B. : 1996, *Nonsmooth impact Mechanics*, Vol. 220 of *Lecture Notes In Control and Information Sciences*, Springer-Verlag, London, Great Britain
- Brooks, R. : 2001, *The relationship between matter and life*, Vol. 409, Nature
- Brooks, R. et Stein, L. : 1993, *Building Brains for Bodies*, Vol. 1439, MIT AI Lab Memo
- Cabodevilla, G., Chaillet, N., et Abba, G. : 1996, *Energy-minimized Gait for a Biped Robot*, *Autonom. Mobile System*, Springer-Verlag
- Cappozzo, A. : 1984, *Gait Analysis Methodology*, Vol. 3, Human Movement Science
- Channon, P., Hopkins, S., et Pham, D. : 1996, *Optimal Control fot an n-legged robot*, Vol. 210, *Journal of Systems and Control Engineering*
- Chen, C. et Allgower, F. : 1998, *A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability*, Vol. 34(10), *Automatica*
- Chen, C. et Shaw, L. : 1982, *On Receding Horizon Feedback Control*, Vol. 18(3), *Automatica*
- Chessé, S. et Bessonnet, G. : 2001, *Optimal Dynamics of Constrained Multibody Systems. Application to Bipedal Walking System Synthesis*, IEEE International Conference on Robotics & Automation
- Chevallereau, C., Formal'sky, A., et Perrin, B. : 1997, *Control of a Walking Robot with Feet Following a Reference Trajectory Derived from Ballistic Motion*, IEEE



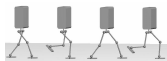
- International Conference on Robotics & Automation
- Chisci, L., Lombardi, A., et Mosca, E. : 1996, *Dual-receding Horizon Control of Constrained Discrete Time Systems*, Vol. 2, European Journal of Control
- Clarac, F. : 1997, *Neurosciences : quoi de neuf ?*, Communiqué de presse CNRS
- Clarke, D., Mohtadi, C., et Tuffs, P. : 1987, *Generalized Predictive Control*, Vol. 23(2), Automatica
- Courtial, E. : 1996, *Commande prédictive et estimation d'état des systèmes non linéaires*, PhD Thesis, Université Claude Bernard Lyon 1, France
- Craig, L., Zhou, J. L., et Tits, A. L. : 1997, *User's guide for CFSQP Version 2.5 : A C code for solving (large scale) constrained nonlinear (minmax) optimization problems, generating iterates satisfying all inequality constraints*, Electrical Engineering Department and Institute for System Research, University of Maryland
- Cutler, C. et Ramaker, B. : 1979, *Dynamic Matrix Control. A Computer Control Algorithm*, AIChE National Meeting
- DeNicolao, G., Magni, L., et Scattolini, R. : 1997, *Stabilizing receding-horizon control of nonlinear time-varying systems*, European Control Conference ECC
- DeNicolao, G., Magni, L., et Scattolini, R. : 2000, *Stability and Robustness of Nonlinear Receding Horizon Control*, Vol. 26, Nonlinear Model Predictive Control, F.Allgöwer, A. Zheng - Progress in Systems and Control Theory
- Denk, J. et Schmidt, G. : 2001a, *Synthesis of a Walking Primitive Database for a Humanoid Robot using Optimal Control Techniques*, Proceedings of IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS2001)
- Denk, J. et Schmidt, G. : 2001b, *Walking Primitive Synthesis for an Anthropomorphic Biped using Optimal Control Techniques*, Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR)
- Donatelli, R. : 1990, *The biomechanics of the foot and ankle*, F.A. Davis, Philadelphia, USA
- Donelan, J., Kram, R., et Kuo, A. : 2001, *Mechanical and metabolic determinants of the preferred step width in human walking*, Vol. 268, Proc. Royal Society Lond.
- Dufour, P. : 2000, *Contribution à la commande prédictive des systèmes à paramètres répartis non linéaires*, PhD Thesis, Université Claude Bernard Lyon 1
- Duysens, J., Van de Crommert, H., et Smits-Engelsman B.C.M., Van der Helm, F. : 2002, *A Walking Robot Called Human : lessons to be learned from neural control of locomotion*, Vol. 35, Journal of Biomechanics
- Espiau, B. et Bip team, t. : 1997, *BIP : A Joint Project for the Development of an Anthropomorphic Biped Robot*, Proc. Int. Conf. on Advanced Robotics
- Espiau, B. et Génot, F. : 2002, *Les robots marcheurs bipèdes*, La robotique mobile



- Hermes
- Espiau, B. et Sardain, P. : 2000, *The Anthropomorphic Biped Robot BIP2000*, Proc. Int. Conf. on Robotics and Automation
- Findeisen, R. et Allgöwer, F. : 2000, *Nonlinear Model Predictive Control for Index-one DAE Systems*, Vol. 26, Nonlinear Model Predictive Control, F.Allgöwer, A. Zheng - Progress in Systems and Control Theory
- Fletcher, R. : 1987, *Practical Methods of Optimization*, John Wiley and Sons
- Floreano, D. et Urzelai, J. : 2000, *Evolutionary Robots with on-line self-organization and behavioral fitness*, Vol. 13, Neural Networks
- France, L., Girault, A., et Espiau, B. : 1999, *Generation of Walk Transient Trajectories for a Biped Robot*, International Conference on Advanced Robotics
- Furusho, J. et Sano, A. : 1990, *Sensor-Based Control of a Nine-link Biped*, Vol. 9-(2), The International Journal of Robotics Research
- Gage, J. : 1990, *An overview of normal walking*, Vol. XXXVIII, American Academy of Orthopaedic Surgeons, Instructional Course Lectures
- Garcia, C., Prett, D., et Morari, M. : 1989, *Model Predictive Control : theory and practice - a survey*, Vol. 25, Automatica
- Garcia, M., Chatterjee, A., Ruina, A., et Coleman, M. : 1997, *The Simplest Walking Model : Stability and Scaling*, Vol. 120, ASME Journal of Biomechanical Engineering
- Gienger, M. and Loeffler, K. et Pfeiffer, F. : 2001, *Towards the Design of a Biped Jogging Robot*, Proc. Int. Conf. on Robotics and Automation
- Giménez y Ribotta, M., Provencher, J., Feraboli-Lohnherr, D., Rossignol, S., Privat, A., et Orsal, D. : 2000, *Activation of Locomotion in Adult Chronic Spinal Rats Is Achieved by Transplantation of Embryonic Raphe Cells Reinnervating a Precise Lumbar Level*, Vol. 20-(13), The Journal of Neuroscience
- Génot, F. : 1998, *Contributions à la modélisation et à la commande des systèmes mécaniques de corps rigides avec contraintes unilatérales*, PhD Thesis-INPG, France
- Goswami, A. : 1999, *Postural Stability of Biped Robots and Foot-Rotation Indicator (FRI) Point*, The International Journal of Robotic Research
- Goswami, A., Espiau, B., et Keramane, A. : 1996, *Limit Cycles and their Stability in Passive Bipedal Gait*, IEEE International Conference on Robotics & Automation
- Grishin, A. et Formal'sky, A. : 1990, *Control of Bipedal Walking Robot by means of Impulses of Finite Amplitude*, Vol. 25-(2), AN SSSR Mekhanika Tverdogo Tela
- Grizzle, J., Abba, G., et Plestan, F. : 1999, *Proving Asymptotic Stability of walking for a biped robot*, Int. Symp. on Mobile, Climbing and Walking Robots

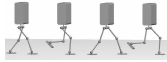


- Guihard, M. et Gorce, P. : 2001, *Dynamic Control of a Biomechanical Inspired Robot : BIPMAN*, Int. Symp. on Mobile, Climbing and Walking Robots
- Guiraud, D., Divoux, J., et Rabischong, P. : 2002, *Identification of a First Order Model of Implanted Electrode on the First SUAW Patient*, IFESS2002, Ljubljana, Slovénie
- Guiraud, D., Pacetti, A., Meola, E., Divoux, J., et Rabischong, P. : 2001, *One year implanted patients follow up : SUAW project first results*, IFESS2001
- Hanneton, S. : 1999, *Le contrôle de la dynamique des mouvements - Application à l'étude d'une tâche de poursuite visuo-manuelle*, Thèse de doctorat - Université Paris 6
- Hemami, H. et Farnsworth, R. : 1977, *Postural and Gait Stability of Planar Five Link Biped by Simulation*, Vol. 22, IEEE Transactions on Automatic Control
- Hirai, K., Hirose, M., Haikawa, Y., et Takenaka, T. : 1998, *The development of the Honda humanoid robot*, Proc. Int. Conf. on Robotics and Automation
- Hollerbach, J. : 1982, *Computers, Brains and the control of movement*, Vol. 5, Trends in Neuroscience
- Hurmuzlu, Y. : 1993a, *Dynamics of Bipedal Gait : Part1*, Vol. 60, Journal Of applied Mechanics
- Hurmuzlu, Y. : 1993b, *Dynamics of Bipedal Gait : Part2- Stability Analysis of a Planar Five-Link Biped*, Vol. 60, Journal Of applied Mechanics
- Hurmuzlu, Y., Génot, F., et Brogliato, B. : 2001, *Modeling, Stability and Control of Biped Robots - A General Framework*, Vol. 4290, Rapport de Recherche INRIA
- Jaulin, L., Kieffer, M., Didrit, O., et Walter, E. : 2001, *Applied interval analysis : with examples in parameter and state estimation, robust control and robotics*, Springer-Verlag
- Kato, R. et Mori, M. : 1984, *Control Method of Biped Locomotion giving Asymptotic Stability of Trajectory*, Vol. 20-(4), Automatica
- Keerthi, S. et Gilbert, E. : 1988, *Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems : stability and moving horizon approximations*, Vol. 57(2), Journal Opt. Theory and Applications
- Khalil, H. : 1996, *Nonlinear Systems*, Prentice Hall Inc.
- Khalil, W. et Kleinfinger, J. : 1986, *A new geometric notation for open and closed loop robots*, IEEE International Conference on Robotics & Automation
- Knoll, A. : 2002, *Editorial : Cui Bono Robo Sapiens ?*, Vol. 12, Autonomous Robots
- Kouvaritakis, B., Cannon, M., et J.A., R. : 2000, *Stability, feasibility, optimality and the degrees of freedom in constrained predictive control*, Vol. 26, Nonlinear Model Predictive Control, F.Allgöwer, A. Zheng - Progress in Systems and Control

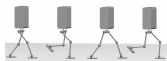


Theory

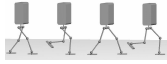
- Kun, A. et Miller, W. : 1996, *Adaptive Dynamic Balance of A Biped Robot Using Neural Networks*, Vol. 1, Proceedings of the 1996 IEEE International Conference on Robotics and Automation
- Lee, E. et Markus, L. : 1967, *Foundations of Optimal Control Theory*, John Wiley & Sons
- LIRMM, INRIA, Propara, et (centre de rééducation fonctionnelle) : 2002, *Contrôle du mouvement du membre inférieur humain paralysé sous stimulation électrique*, Projet Interdisciplinaire de recherche CNRS - ROBEA
- Loeb, G. : 1983, *Finding common ground between robotics and physiology*, Vol. 6, Trends in Neuroscience
- Lötstedt, P. : 1982, *Mechanical Systems of Rigid Bodies subject to Unilateral Constraints*, Vol. 42, SIAM J. Appl. Math.
- Lydoire, F. : 2001, *Simulation d'un robot marcheur et génération de démarches paramétrées*, Rapport de DEA (encadré par C. Azevedo)
- Lydoire, F., Azevedo, C., Espiau, B., et P. Poignet : 2002, *3D Parametrized Gaits for Biped Walking*, Climbing And Walking Robots
- Manamani, N., M'Sirdi, N., et Nadjar-Gauthier, N. : 1997, *Sliding Mode Control of Pneumatic Robot Leg*, Proc. of the European Control Conference
- Manamani, N., M'Sirdi, N., et Nadjar-Gauthier, N. : 1998, *Methodology for Control of a Legged Robot with Fast Dynamics*, RoManSy 12
- Mayne, D. et Michalska, H. : 1990, *Receding Horizon Control of Nonlinear Systems*, Vol. 35, IEEE Trans. Automat. Contr.
- Mayne, D., Rawlings, J., Rao, C., et Sokaert, P. : 2000, *Constrained model predictive control : Stability and optimality*, Vol. 36(6), Automatica
- McGeer, T. : 1990, *Passive Dynamic Walking*, Vol. 9-(2), International Journal of Robotics Research
- Meadows, E. et Rawlings, J. : 1993, *Receding Horizon Control with an Infinite Horizon*, Proc. Amer. Contr. Conf.
- Menzel, P. et D'Alusio, F. : 2000, *Evolution of a new species Robosapiens*, Material World books
- Merlet, J.-P. : 2000, *ALIAS : an interval analysis based library for solving and analyzing system of equations*, SEA Toulouse
- Michalska, H. et Mayne, D. : 1993, *Robust Receding Horizon Control of Constrained Nonlinear Systems*, Vol. 38(11), IEEE Trans. Automat. Contr.
- Minoux, M. : 1983, *Programmation mathématique : théorie et algorithmes*, Vol. T1, Dunod



- Newell, K. : 1995, *Continuité et discontinuité dans l'acquisition du mouvement*, H.Ripoll, J.Bilard, M.Durand, J.Keller and M.Lévêque (Eds) ; Psychologie du sport : questions actuelles - Parisé : Éditions Revue EPS
- Nubar, Y. et Contini, R. : 1961, *A Minimal Principle in Biomechanics*, Bulletin of Mathematical Biophysics
- Park, J. et Chung, H. : 1999, *ZMP Compensation by On-Line Trajectory Generation for Biped Robots*, Proceedings of IEEE Conference on Systems, Man and Cybernetics (SMC'99)
- Park, J. et Kim, K. : 1998, *Biped Robot Walking Using Gravity Compensated Inverted Pendulum Mode and Computed-Torque Control*, IEEE International Conference on Robotics & Automation
- Patla, A. : 1991, *Understanding the Control of Human Locomotion : A Prologue*, Elsevier Science Publishers
- Pfeiffer, F. et Glocker, C. : 1996, *Multibody Dynamics With Unilateral Contacts*, Wiley-Interscience Publication, John Wiley and Sons Inc., New York, NY, USA
- Popovic, M., Keller, T., Pappas, I., Dietz, V., et Morari, M. : 2001, *Surface-Stimulation technology for Grasping and Walking Neuroprostheses*, IEEE Engineering in Medicine and Biology
- Pratt, G. : 1998, *Legged Robots at MIT : what's new since Raibert*, Vol. 7-(3), IEEE Robotics & Automation Magazine
- Pratt, J. et Pratt, G. : 1998, *Intuitive Control of a Planar Bipedal Walking Robot*, IEEE International Conference on Robotics & Automation
- Propoi, A. : 1963, *Use of linear programming methods for synthesizing sampled data automatic systems*, Vol. 24(7), Autom. Remote Control
- Quintern, J. : 1998, *Application of Functional Electrical Stimulation in Paraplegic Patients*, Vol. 10, NeuroRehabilitation
- Rabishong, P. : 1996, *Robionique et contrôle de l'appareil moteur chez l'homme*, Revue Française de Mécanique
- Rawlings, J. et Muske, K. : 1993, *The stability of constrained receding horizon control*, Vol. 38, IEEE Trans. Automat. Contr.
- Richalet, J. : 1993, *Pratique de la commande prédictive*, Hermès science publications
- Richalet, J., Rault, A., Testud, J., et Papon, J. : 1978, *Model predictive heuristic control : Applications to industrial processes*, Vol. 14(2), Automatica
- Riener, R. : 1998, *Patient-Driven Control of FES-Supported Standing Up : A Simulation Study*, Vol. 6-(2), IEEE Transactions on Rehabilitation Engineering
- Rose, C., Guenter, B., Bondenheimer, B., et Cohen, M. : 1996, *Efficient generation of motion transitions using spacetime constraints*, Computer Graphics

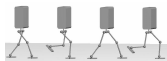


- Rose, J. et Gamble, J. : 1994, *Human Walking*, William and Wilkins, Baltimore, USA
- Rostami, M., Bessonnet, G., et Sardain, P. : 1998, *Optimal Gait Synthesis of a Planar Biped*, IFAC int. Workshop on Motion Control
- Roussel, L. : 1998, *Génération de trajectoires optimales de marche pour un robot bipède*, PhD Thesis-INPG, France
- Sano, A. et Furusho, J. : 1990, *Realization of a Natural Dynamic Walking Using Angular Momentum Information*, IEEE International Conference on Robotics & Automation
- Sardain, P., Rostami, M., et Bessonnet, G. : 1997, *An Anthropomorphic Biped Robot : Dynamic Concepts and Technological Design*, Vol. 28-6, IEEE Trans. on Systems, Man, and Cybernetics
- Sardain, P., Rostami, M., et Bessonnet, G. : 1998, *An Anthropomorphic Biped Robot : Dynamic Concepts and Technological Design*, Vol. 28-6, Proc. IEEE Trans. on Syst. Man and Cybernetics
- Scokaert, P., Mayne, D., et Rawlings, J. : 1999, *Suboptimal Model Predictive Control (Feasibility Implies Stability)*, Vol. 44(3), IEEE Trans. Automat. Contr.
- Scokaert, P., Rawlings, J., et Meadow, E. : 1997, *Discrete-time with Perturbations : Application to Model Predictive Control*, Vol. 33(3), Automatica
- Sdr-3x : 2000, Sony, [http ://www.sony.co.jp/en/SonyInfo/News/Press/200011/00-057E2](http://www.sony.co.jp/en/SonyInfo/News/Press/200011/00-057E2)
- Shih, C., Gruver, W., et Zhu, Y. : 1991, *Fuzzy Logic Force Control for a Biped Robot*, Vol. b, IEEE International Symposium on Intelligent Control
- Slotine, J. et Li, W. : 1991, *Applied Nonlinear Control*, Prentice Hall, New Jersey
- Slotine, J. J. : 2002, *À l'heure des neurosciences*, Vol. 350, La Recherche
- Takanishi, A. et Lim, H. : 1990, *Realization of Dynamic Biped Walking Stabilized By Trunk Motion On a Sagittally Uneven Surface*, IEEE International Workshop on Intelligent Robots and Systems
- TheOrccadTeam : 1998, *An Integrated and Modular Approach for the Specification, the Validation and the Implementation of Complex Robotics Missions*, Vol. 17-(4), International Journal of Robotics Research, Special Issue on Integrated Architectures for Robot Control and Programming
- Thuilot, B., Goswami, A., et Espiau, B. : 1997, *Bifurcation and Chaos in a Simple Passive Bipedal Gait*, IEEE International Conference on Robotics & Automation
- Unuma, M., Anjyo, K., et Takeuchi, R. : 1995, *Fourier Principles for emotion-based human figure animation*, Computer Graphics
- VanDerKooij, H. : 2000, *Human Balance Control in standing and walking*, PhD

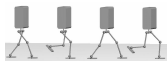


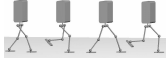
Thesis-Twente University

- Vaughan, C., Davis, B., et O'Connor, J. : 1992, *Dynamics of Human Gait*, Human Kinetics
- Vickers, M. : 2000, *Exoskeleton Operational Concepts : Lessons from Future War-games*, Exoskeletons for Human Performance Augmentation (EHPA) Workshop
- Vidyasagar, M. : 1993, *Nonlinear System Theory*, Englewood Cliffs, NJ, Prentice Hall
- Viéville, T., Romann, F., Hotz, B., Mathieu, H., Buffa, M., Robert, L., et Facao, P. : 1993, *Autonomous navigation of a Mobile Robot using Inertial and Visual Cues*, Intelligent Robots and Systems, M. Kikode, T. Sato and K. Tatsuno ed., Yokohama
- Vukobratović, M., Borovav, B., Surla, D., et Stokić, D. : 1990, *Biped Locomotion : Dynamics, Stability, Control and Application*, Springer-Verlag, London, Great Britain
- Wagner, N., Mulder, M., et Hsu, M. : 1988, *A knowledge based control strategy for a biped*, Vol. 3, IEEE International Conference on Robotics & Automation
- Weiss, P. : 2001, *Dances with Robots*, Vol. 159-(26), Science News
- Whittle, M. : 1996, *Clinical gait analysis : a review*, Vol. 15, Human Movement Science
- Wieber, P. : 2000, *Modélisation et Commande d'un Robot marcheur Anthropomorphe*, PhD Thesis-École Nationale Supérieure des Mines de Paris, France
- Winter, D. : 1990, *Biomechanics and motor control of human movement*, John Wiley, New York, USA
- Winter, D., Patla, A., et Frank, J. : 1990, *Assessment of balance control in humans*, Vol. 16, Medical Progress through Technology
- Yamaguchi, G. et Zajac, F. : 1990, *Restoring Unassisted Natural Gait to Paraplegics via Functional Electrostimulation : A Computer Simulation Study*, IEEE Transactions on Biomedical Engineering
- Yamaguchi, J., Soga, E., Inoue, S., et Takanishi, A. : 1999, *lopment of a Bipedal Humanoid Robot - Control Method of Whole Body Cooperative Dynamic Biped Walking*, Proc. Int. Conf. on Robotics and Automation
- Yamasaki, F., Matsui, T., Miyashita, T., et Kitano, H. : 2001, *PINO The Humanoid : A Basic Architecture*, Vol. 0-2019, Lecture Notes in Computer Science
- Yang, J. : 1994, *A Control Study of a Kneeless Biped Locomotion System*, Vol. b-(331), Journal of the Franklin Institute
- Zatsiorsky, V., Werner, S., et Kaimin, M. : 1994, *Basic Kinematics of Walking*, Vol. 34-(2), The Journal of Sports Medicine and Physical Fitness



Zheng, Y. et Hemami, H. : 1984, *Impact Effects of Biped Contact with the Environment*, Vol. 14-3, IEEE Trans. Syst. Man Cyber.





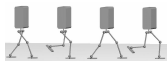
Annexe : Définitions

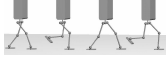
Nous proposons ci-dessous quelques définitions utilisées au cours du mémoire.

- un algorithme décrit par une application multivoque $\mathcal{A}$ est **globalement convergent** si quelque soit le point de départ x^0 choisi, la suite x^k engendrée par $x^{k+1} \in \mathcal{A}(x^k)$ converge vers un point satisfaisant une condition nécessaire d'optimalité (Minoux, 1983)
- Soient deux ensembles $\mathcal{E}$ et $\mathcal{F}$, on appelle **graphe d'une correspondance** entre $\mathcal{E}$ et $\mathcal{F}$ un sous-ensemble Γ du produit cartésien $\mathcal{E} \times \mathcal{F}$.
- La **correspondance** (*mapping*) est l'opérateur f qui, à tout élément x de l'ensemble de départ $\mathcal{E}$ pour lequel il existe au moins un couple $(x, y) \in \Gamma$, associe son **image** $f(x)$ dans $\mathcal{F}$:

$$f(x) = \{y \mid (x, y) \in \Gamma\}$$

- une correspondance telle que pour tout $x \in \mathcal{E}$, il existe au moins un couple $(x, y) \in \Gamma$ est appelée une **application multivoque**.





Annexe : Faisabilité - mode dual

Nous reprenons ici les étapes de l'approche mode dual de la sous optimalité présenté par M. Scokaert et al. (Scokaert et al., 1999).

Les contraintes associées au problème sont :

$$\begin{aligned} u_k &\in U \\ x_k &\in X \\ x_{j+1} &= f(x_j, u_j) \end{aligned} \tag{1}$$

Si l'on ajoute une contrainte de stabilité :

$$x_{k+N} \in W \tag{2}$$

où W est un sous ensemble compact convexe de X qui contient l'origine et dans lequel il existe un retour d'état linéaire h_L .

Si $W \neq 0$, on choisit une fonction de coût qui vérifie :

$$\begin{aligned} L(x, u) &= 0, \forall x \in W \\ L(x, u) &= x^T \cdot x + u^T \cdot u, \forall x \notin W \end{aligned} \tag{3}$$

Théorème : Soit F l'ensemble des états pour lesquels il existe une séquence de commandes qui satisfait 1, 2. La stratégie à horizon fini définie en mode dual est asymptotiquement stabilisante.

Preuve :

1) La stabilité est une propriété locale garantie par la propriété de stabilisation de la commande choisie dans W .

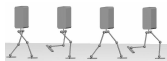
2) La convergence en temps fini vers W se démontre comme suit.

Si le critère s'écrit :

$$J_k = \sum_{j=k}^{k+N-1} u_j^T \cdot u_j + x_j^T \cdot x_j \tag{4}$$

À l'instant k et à l'issue de l'optimisation, on obtient J_k^f :

$$J_k^f = \sum_{j=k}^{k+N-1} u_j^{fT} \cdot u_j^f + x_j^{fT} \cdot x_j^f \tag{5}$$



En initialisant à l'itération suivante la séquence de commande par :

$$\begin{aligned} u_{k+i}^0 &= u_{k+i}^f, \forall i \in [2, N-1] \\ u_{k+N}^0 &= h_L(x_{k+N-1}) \end{aligned} \quad (6)$$

En s'appuyant sur le fait que : $u_{k+N}^T u_{k+N} + x_{k+N}^T x_{k+N} = 0$ puisque $x_{k+N} \in W$, on obtient :

$$J_{k+1}^0 = J_k^f - u_k^{f^T} u_k^f \quad (7)$$

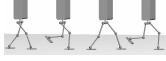
après minimisation :

$$J_{k+1}^f \leq J_{k+1}^0 \quad (8)$$

Alors :

$$J_{k+1}^f \leq J_k^f - u_{k+1}^{f^T} u_{k+1}^f \quad (9)$$

Le critère est décroissant et l'on peut garantir que l'on converge vers W en un temps fini.



Annexe : Calcul de P

On peut adjoindre une contrainte d'état terminal égalitaire dans le critère pour forcer le système à atteindre cet état en un temps fini. De la même façon que dans le cas d'une approche horizon quasi infini (page 102), ici :

$$J = \frac{1}{2} U_{k+1}^T U_{k+1} + X_{N_p}^T P X_{N_p} \quad (10)$$

Un retour d'état linéaire est utilisé pour déterminer la matrice de pénalité terminale P (Chen et Allgower, 1998).

L'objectif est de ramener le pendule en son point d'équilibre ($q = 0, \dot{q} = 0, \Gamma = 0$).

★ La première étape consiste à linéariser le système autour de l'origine :

$$\dot{x} = Ax + B\Gamma \quad (11)$$

$$A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & -\frac{f}{ml^2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 19.62 & -0.04 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ \frac{1}{ml^2} \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$$

★ on s'assure alors que le système est contrôlable :

$$rg(C_o) = \begin{bmatrix} B & AB \end{bmatrix} = 2$$

★ on recherche alors un retour d'état linéaire $\Gamma = -Kx$ tel que la fonction de coût $J = \int x' Q x dt$ soit minimisée.

$$K = \begin{bmatrix} 8.81 & 2.2047 \end{bmatrix}$$

$$A_K = A - BK = \begin{bmatrix} 0 & 1 \\ -19.62 & -8.8590 \end{bmatrix}$$

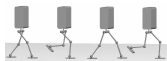
★ la partie réelle de la plus grande valeur propre de A_K est

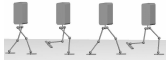
$$\lambda_{A_K} = -4.4495$$

On choisit la constante $k = 4.0$ et l'on résoud l'équation de Lyapunov :

$$(A_K + k.Id).P + P.(A_K + k.Id)^T = -K^T K$$

$$\text{soit : } P = 10^5 \begin{bmatrix} 0.0792 & -0.3173 \\ -0.3173 & 1.2814 \end{bmatrix}$$





Annexe : Étude d'un pendule inverse

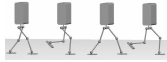
Sommaire

★ Problématique

étudier l'influence des paramètres et réglages de la commande prédictive sur un cas simple

★ Simulation

présentation de l'étude systématique d'un cas d'école : le pendule inversé



Présentation du pendule

On considère le pendule inversé (fig.1) dont la dynamique s'écrit :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{g}{l} \cdot \sin(x_1) - \frac{f}{ml^2} x_2 + \frac{1}{ml^2} \cdot \Gamma \end{cases} \quad (12)$$

avec $x_1 = q$ et $x_2 = \dot{q}$.

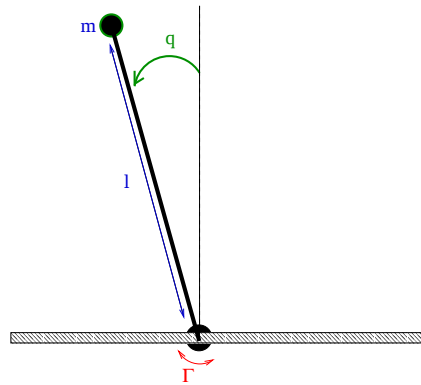
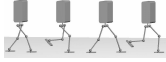


FIG. 1 – Pendule inversé

Nous avons utilisé le jeu de valeurs numériques suivant :

Masse	Frottement	Longueur	constante de gravité
$m=1,0 \text{ kg}$	$f=0,01 \text{ m}^{-1} \text{ s}^{-1} \text{ s}$	$l=0,5 \text{ m}$	$g=9,81 \text{ Nm}^{-1}$

Nous avons effectué une série de simulations afin d'évaluer l'influence des différents paramètres et réglages sur le comportement du système.



Problème 1 - optimisation de la somme quadratique des couples

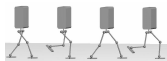
Le problème consiste à :

$$\begin{aligned} \text{résoudre :} \quad & \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} \\ \text{initialisé avec :} \quad & U_{k+1}^O = [\Gamma_{k+1|k}^*; \dots \Gamma_{k+N_p-1|k}^*; \Gamma_{k+N_p-1|k}^*]^T \end{aligned}$$

Le problème est un problème d'optimisation sans contraintes, on utilise un algorithme QN : *fminunc* sous Matlab.

Description des simulations :

- *simu*_{0₁} - le système est initialisé en sa position d'équilibre
- *simu*_{0₂} - le système est initialisé en dehors de sa position d'équilibre

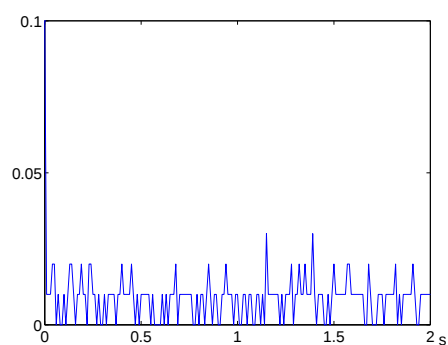


$simu_{0_1}$ - conditions initiales = position d'équilibre

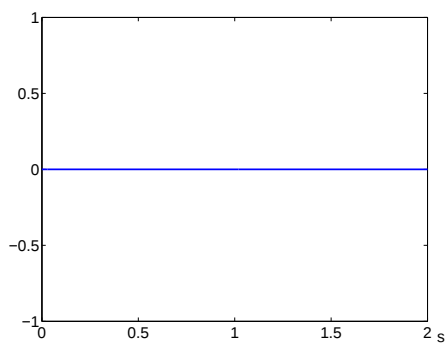
Critère à minimiser :	$\frac{1}{2}U^T U$
Contraintes à satisfaire :	
Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
Algorithme d'optimisation :	quasi-newton
Environnement :	matlab
Conditions initiales :	$q_0 = 0, \dot{q}_0 = 0, \Gamma_0 = 0$
Horizon de prédiction :	10



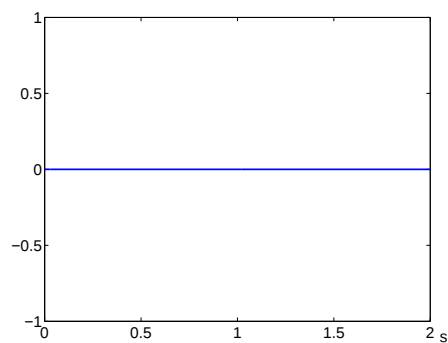
comportement du pendule



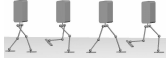
temps de calcul optimisation (s)



fonction objectif final

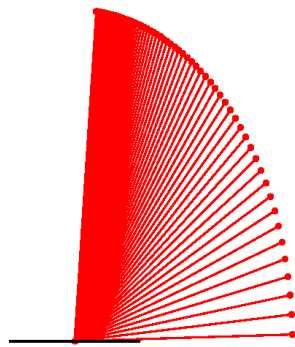


couple articulaire Γ (Nm)

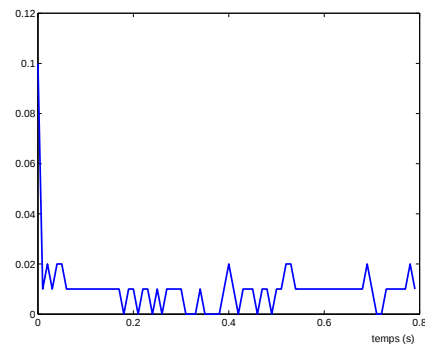


*simu*₀₂ - conditions initiales $\neq$ position d'équilibre

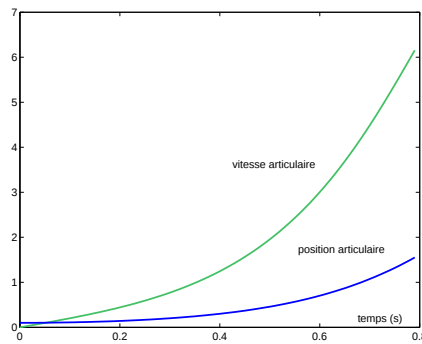
	Critère à minimiser :	$\frac{1}{2}U^T U$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	10



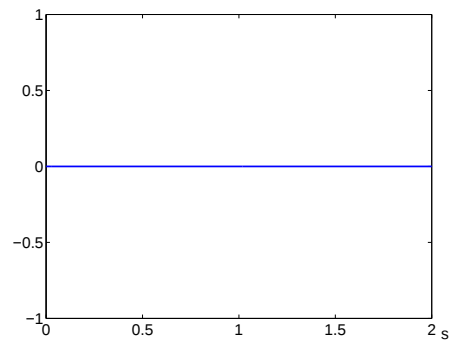
comportement du pendule



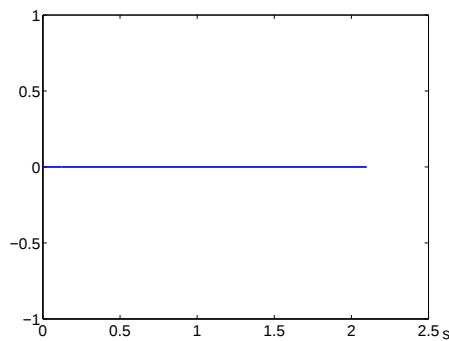
temps de calcul optimisation (s)



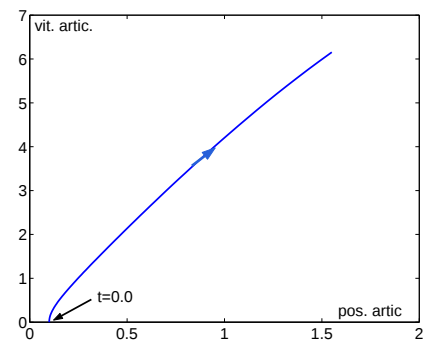
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



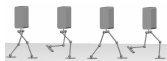
couple articulaire Γ (Nm)



fonction objectif final



plan de phase



Problème 2 - optimisation de la somme quadratique des couples et de l'état final

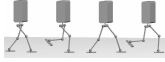
Le problème consiste à :

$$\begin{aligned} \text{résoudre :} \quad & \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} + X'_{N_p} P X_{N_p} \\ \text{initialisé avec :} \quad & U_{k+1}^O = [\Gamma_{k+1|k}^*; \dots \Gamma_{k+N_p-1|k}^*; \Gamma_{k+N_p-1|k}^*]^T \end{aligned}$$

Le problème est un problème d'optimisation sans contraintes, on utilise un algorithme QN : *fminunc* sous Matlab.

Description des simulations :

- *simu₁₁* - le système est initialisé en sa position d'équilibre
- *simu₁₂* - le système est initialisé en dehors de sa position d'équilibre
- *simu₁₃* - on reprend *simu₁₂* en faisant passer l'horizon de prédiction de 10 à 30
- *simu₁₄* - on reprend *simu₁₃* en faisant passer l'horizon de prédiction de 30 à 60
- *simu₂₁* - on reprend *simu₁₃* en modifiant la matrice P dans le critère. On fait passer le paramètre de réglage k de 4 à 1
- *simu₃₁* - on reprend *simu₁₃* en modifiant les conditions initiales du système, on passe de $q_0 = 0.1$ à $q_0 = -\pi/2$
- *simu₃₂* - on reprend *simu₁₃* en modifiant les conditions initiales du système, on passe de $\dot{q}_0 = 0$ à $\dot{q}_0 = 1$
- *simu₃₃* - on reprend *simu₁₃* en modifiant les conditions initiales du système, on passe de $\dot{q}_0 = 0$ à $\dot{q}_0 = -1$

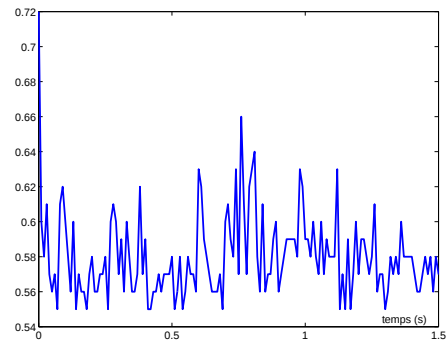


$simu_{1_1}$ - conditions initiales = position d'équilibre

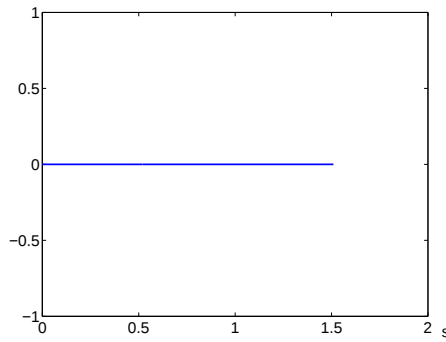
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	10



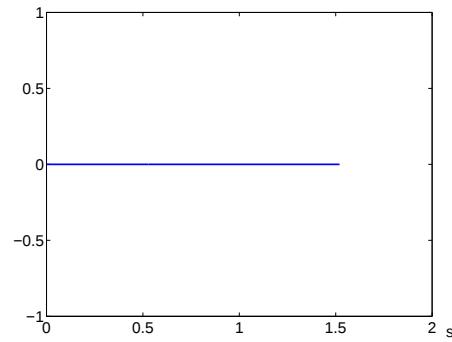
comportement du pendule



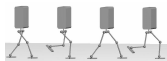
temps de calcul optimisation (s)



fonction objectif final



couple articulaire Γ (Nm)

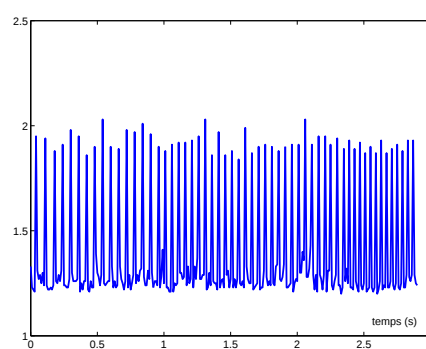


$simu_{12}$ - conditions initiales $\neq$ position d'équilibre

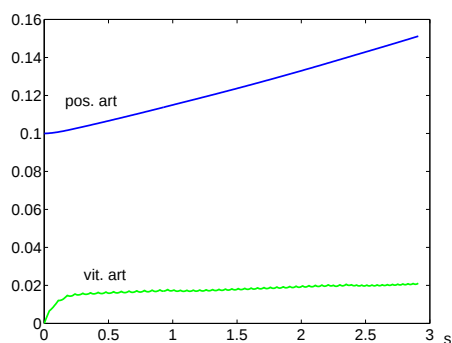
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	10



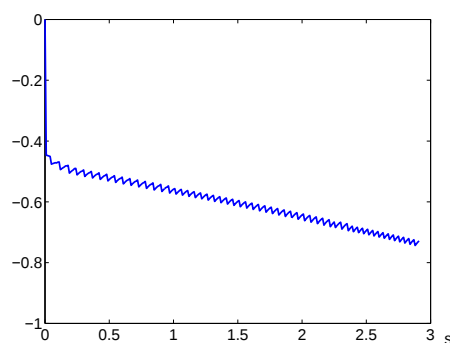
comportement du pendule



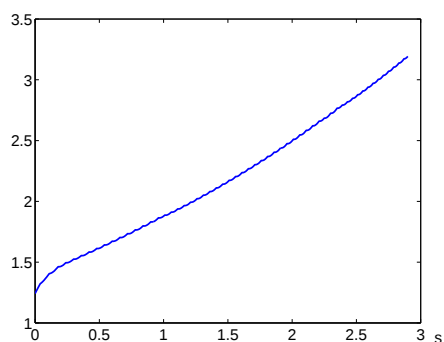
temps de calcul optimisation (s)



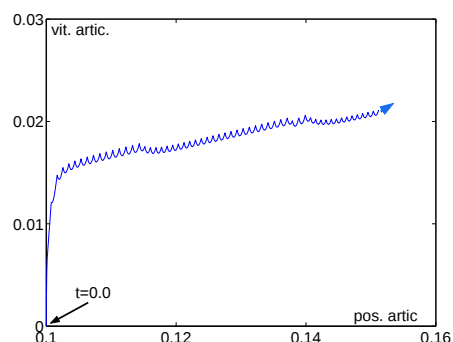
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



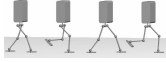
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

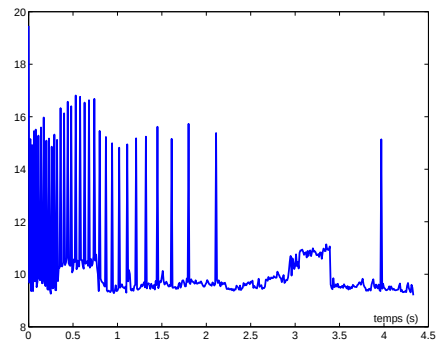


*simu*₁₃ - **augmentation** N_p ($10 \rightarrow 30$)

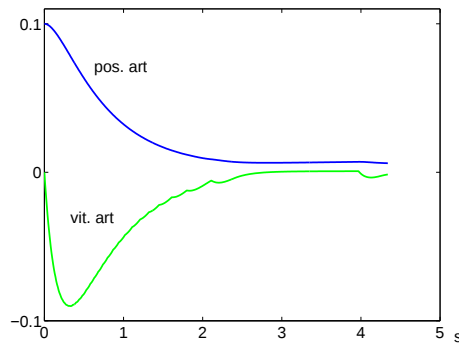
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



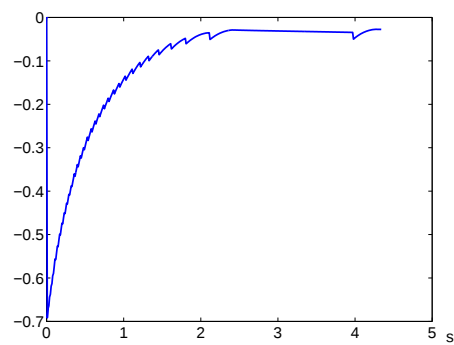
comportement du pendule



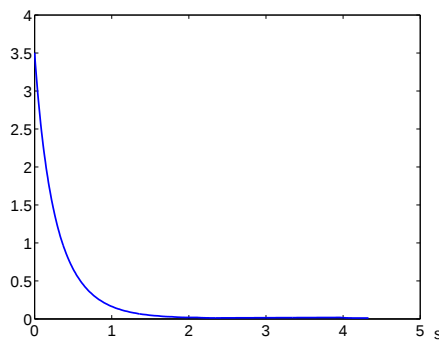
temps de calcul optimisation (s)



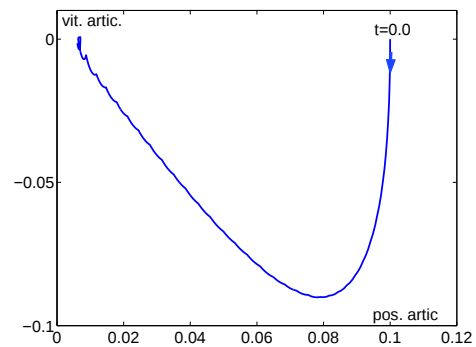
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



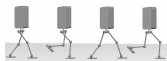
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

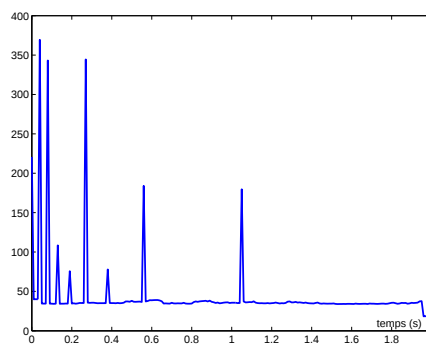


$simu_{14}$ - augmentation N_p ($30 \rightarrow 60$)

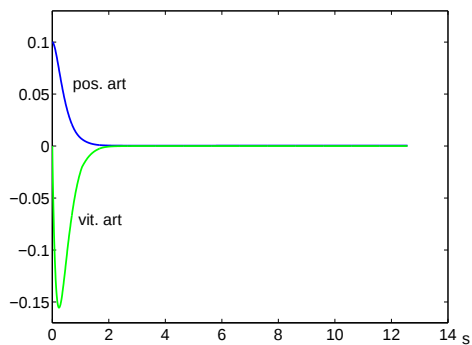
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
★	Horizon de prédiction :	60



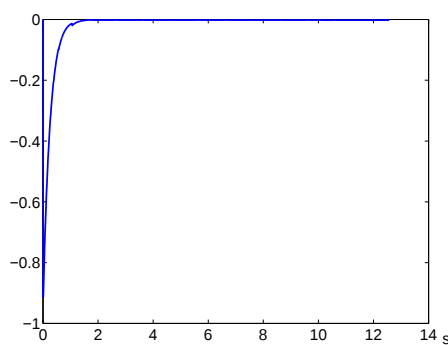
comportement du pendule



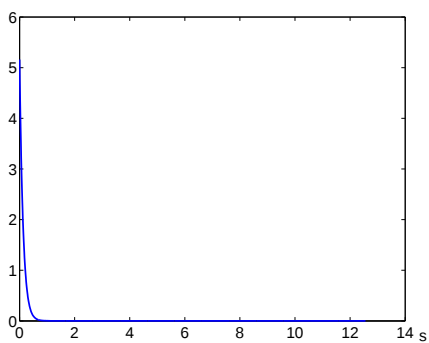
temps de calcul optimisation (s)



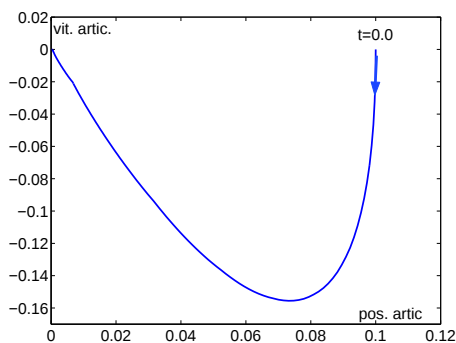
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



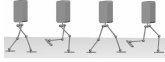
couple articulaire Γ (Nm)



fonction objectif final



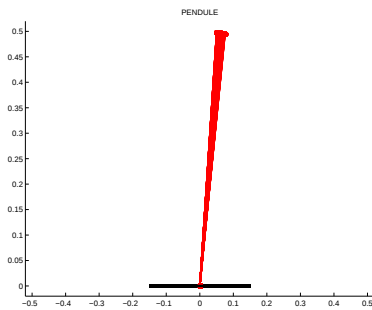
plan de phase



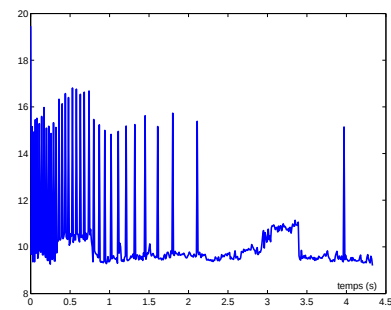
$simu_{21}$ - choix de P ($k = 4 \rightarrow 1$)

(pour les détails du calcul voir p.155)

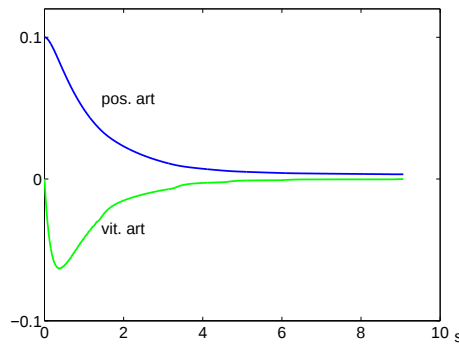
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P_2 X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



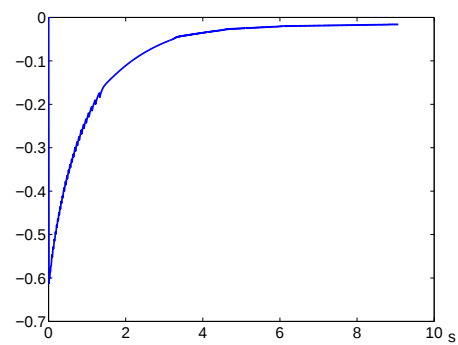
comportement du pendule



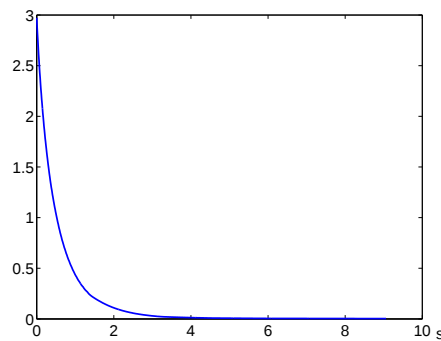
temps de calcul optimisation (s)



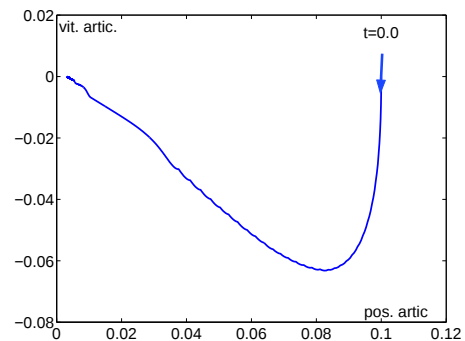
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



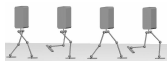
couple articulaire Γ (Nm)



fonction objectif final

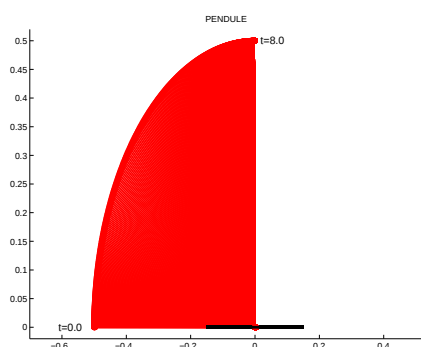


plan de phase

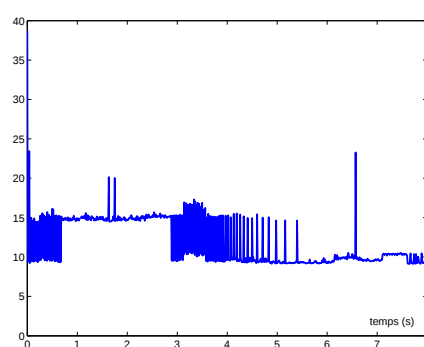


*simu*₃₁ - position initiale très défavorable

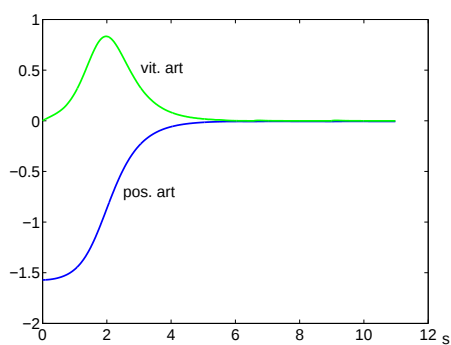
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P_2 X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = -\pi/2, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



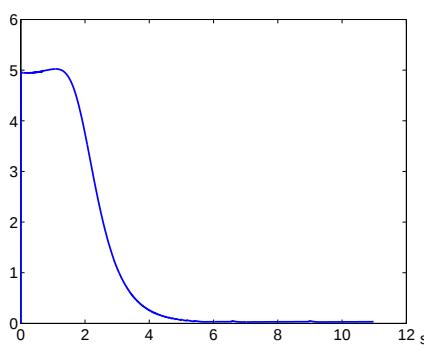
comportement du pendule



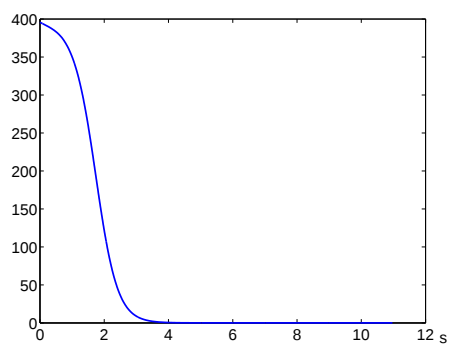
temps de calcul optimisation (s)



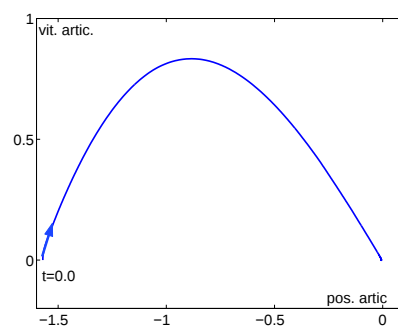
position q (rad) et vitesse $\dot{q}$ (rad.s^{-1})



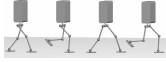
couple articulaire Γ (Nm)



fonction objectif final

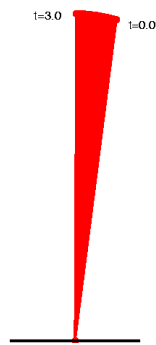


plan de phase

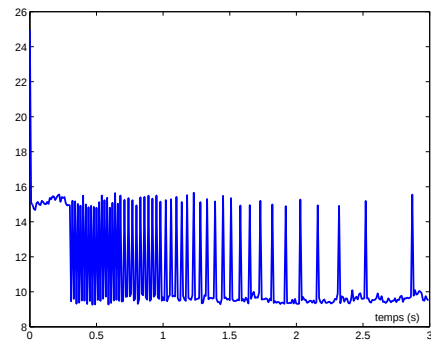


*simu*₃₂ - vitesse initiale très défavorable

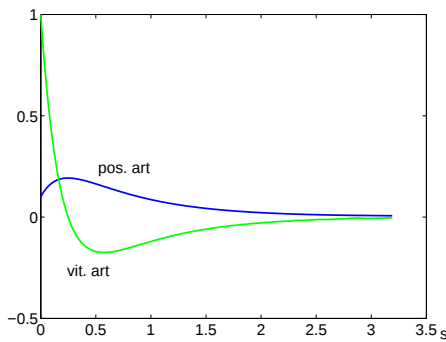
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P_2 X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 1, \Gamma_0 = 0$
	Horizon de prédiction :	30



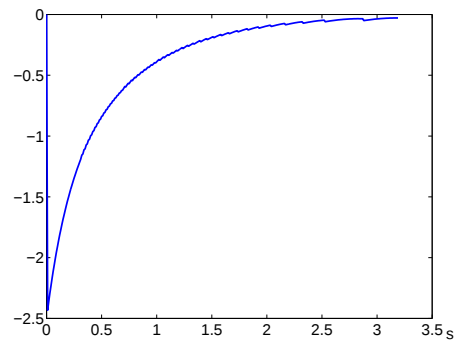
comportement du pendule



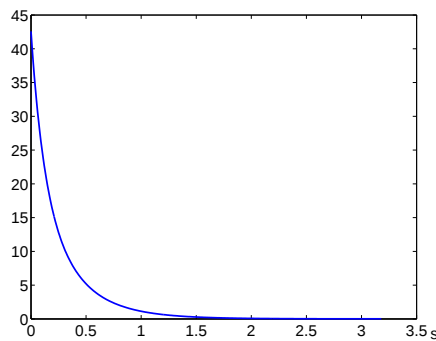
temps de calcul optimisation (s)



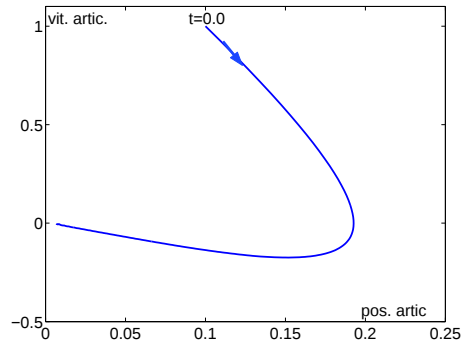
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



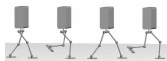
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

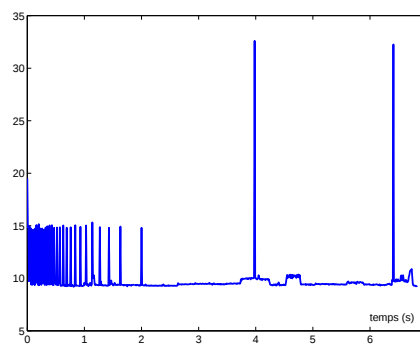


$simu_{33}$ - vitesse initiale favorable

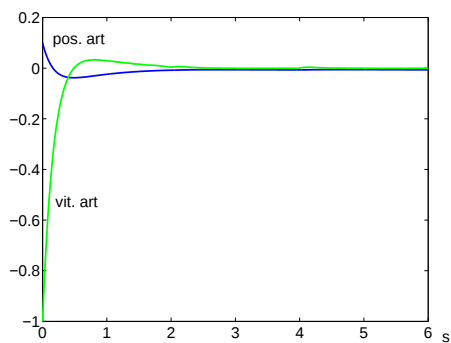
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P_2 X_{N_p}$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = -1, \Gamma_0 = 0$
	Horizon de prédiction :	30



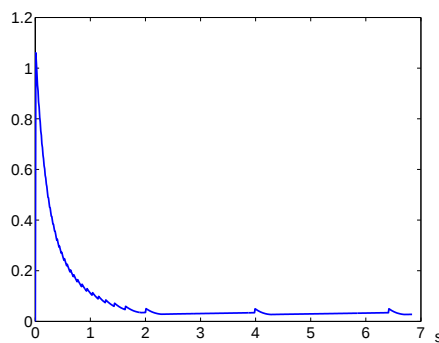
comportement du pendule



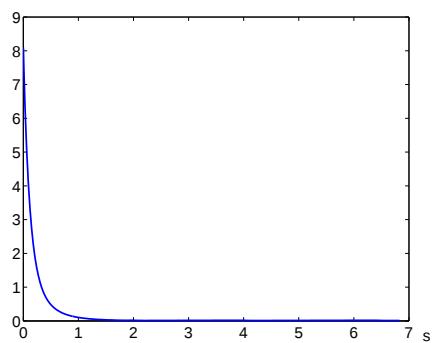
temps de calcul optimisation (s)



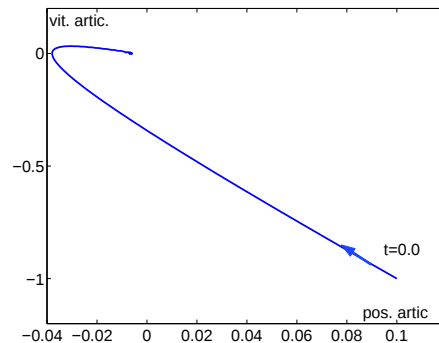
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



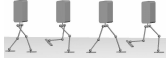
couple articulaire Γ (Nm)



fonction objectif final



plan de phase



Problème 3 - optimisation de la somme quadratique des couples sous contraintes

Le problème consiste à :

$$\begin{aligned}
 &\text{résoudre :} && \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} \\
 &\text{soumis à :} && q_{min} \leq q \leq q_{max} \\
 &&& \dot{q}_{min} \leq \dot{q} \leq \dot{q}_{max} \\
 &&& \Gamma_{min} \leq \Gamma \leq \Gamma_{max} \\
 &\text{initialisé avec :} && U_{k+1}^O = [\Gamma_{k+1|k}^*; \dots \Gamma_{k+N_p-1|k}^*; \Gamma_{k+N_p-1|k}^*]^T
 \end{aligned}$$

avec : $q_{min} = -0.2$, $q_{max} = 0.2$, $\dot{q}_{min} = -2$, $\dot{q}_{max} = 2$, $\Gamma_{min} = -100$, $\Gamma_{max} = 100$.

Le problème est un problème d'optimisation sous contraintes, on utilise un algorithme SQP : *fmincon* sous Matlab ou FSQP sous Scilab.

Description des simulations :

– Résolution avec SQP :

$simu_{4_1}$ - les conditions initiales du système satisfont les contraintes

$simu_{4_2}$ - les conditions initiales du système ne satisfont pas les contraintes

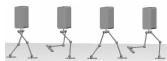
– Résolution avec FSQP :

$simu_{4_1^b}$ - les conditions initiales du système satisfont les contraintes

$simu_{4_2^b}$ - l'horizon de prédiction passe de 10 à 30

$simu_{4_3^b}$ - les conditions initiales du système ne satisfont pas les contraintes

$simu_{4_4^b}$ - les conditions initiales du système ne satisfont pas les contraintes

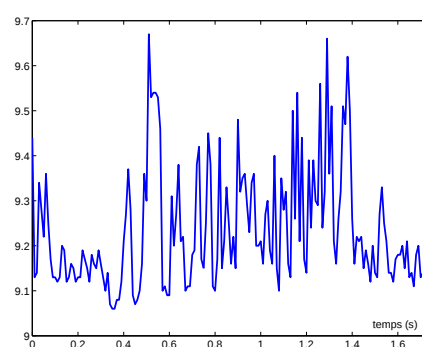


$simu_{4_1}$ - conditions initiales $\in$ domaine admissible

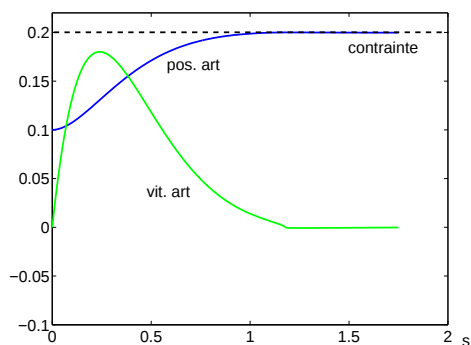
Critère à minimiser :	$\frac{1}{2}U^T U$
Contraintes à satisfaire :	$ q \leq 0.2, \dot{q} \leq 2, \Gamma \leq 100$
Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
Algorithme d'optimisation :	quasi-newton
Environnement :	matlab
Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
Horizon de prédiction :	30



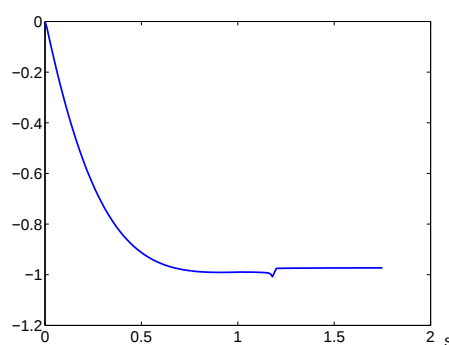
comportement du pendule



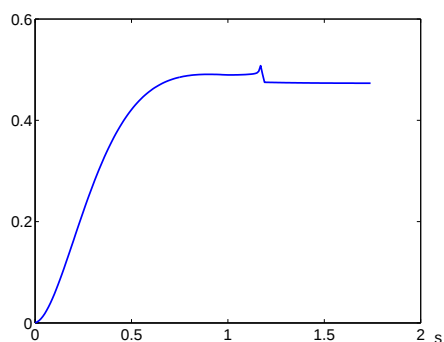
temps de calcul optimisation (s)



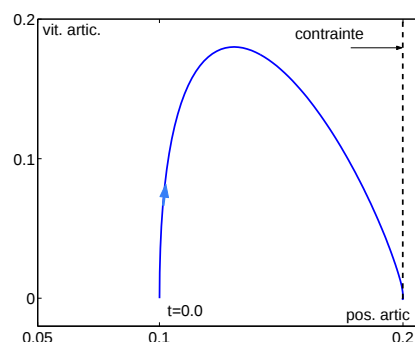
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



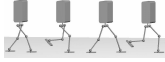
couple articulaire Γ (Nm)



fonction objectif final

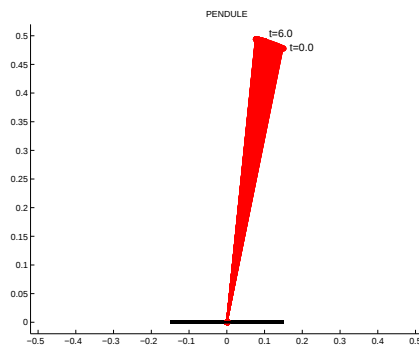


plan de phase

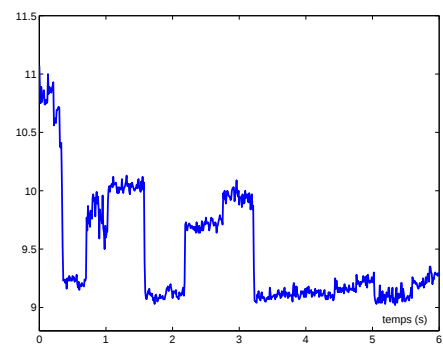


$simu_{4_2}$ - conditions initiales $\notin$ domaine admissible

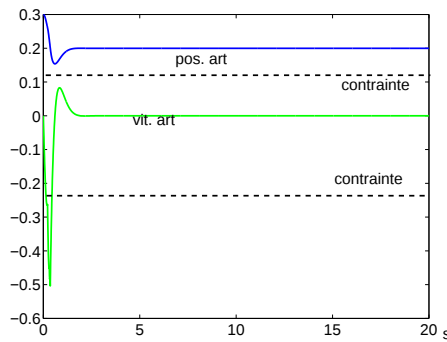
★	Critère à minimiser :	$\frac{1}{2}U^T U$
	Contraintes à satisfaire :	$ q \leq 0.2, \dot{q} \leq 2, \Gamma \leq 100$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



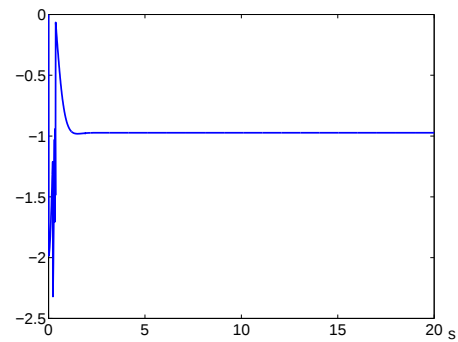
comportement du pendule



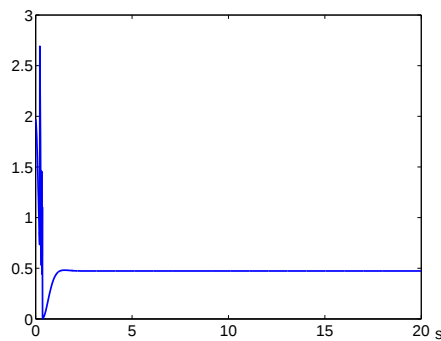
temps de calcul optimisation (s)



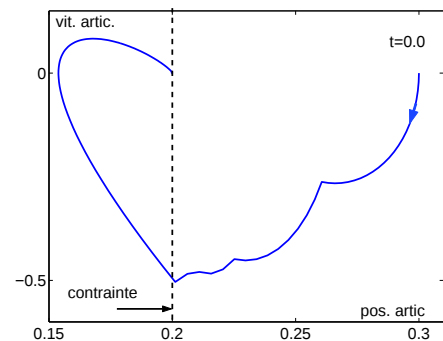
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



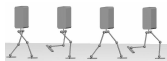
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

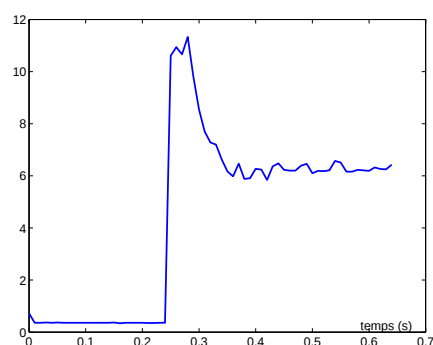


$simu_{41}^b$ - FSQP : $CI \in \text{domaine admissible}$

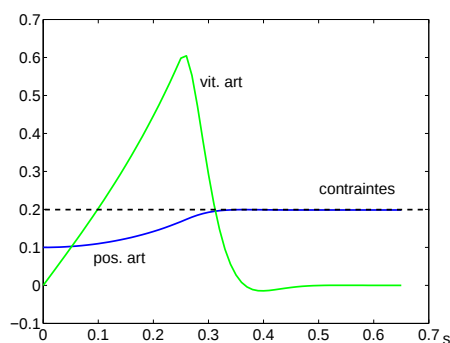
	Critère à minimiser :	$\frac{1}{2}U^T U$
★	Contraintes à satisfaire :	$ q \leq 0.2$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
★	Algorithme d'optimisation :	FSQP
★	Environnement :	scilab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	10



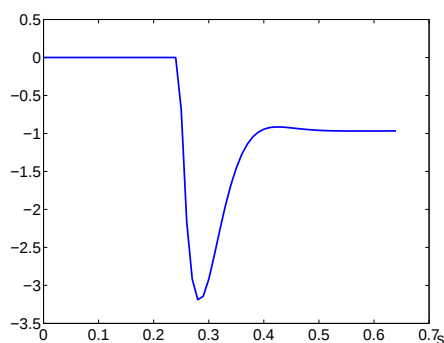
comportement du pendule



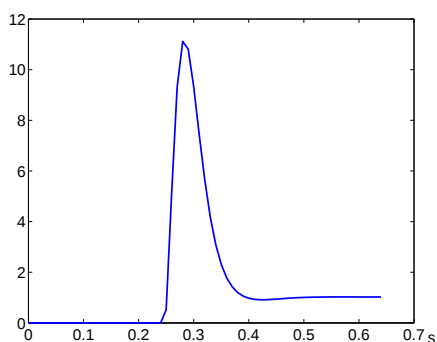
temps de calcul optimisation (s)



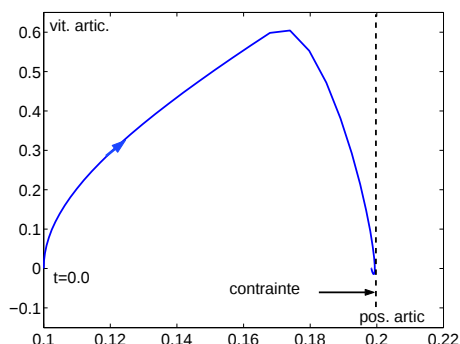
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



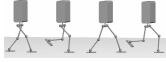
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

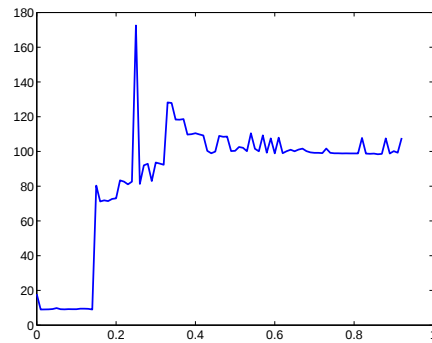


$simu_{42}^b$ - **augmentation** N_p $10 \rightarrow 30$

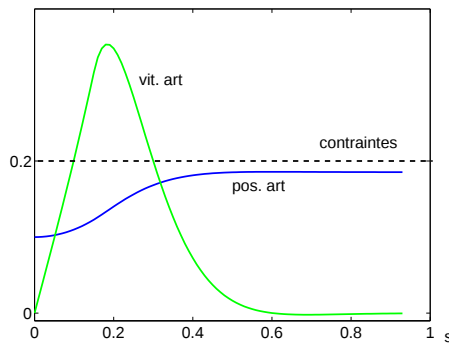
	Critère à minimiser :	$\frac{1}{2}U^T U$
	Contraintes à satisfaire :	$ q \leq 0.2$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	FSQP
	Environnement :	scilab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
★	Horizon de prédiction :	30



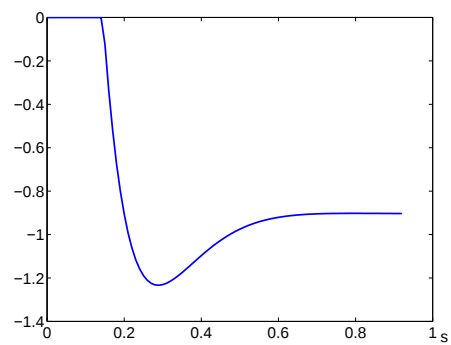
comportement du pendule



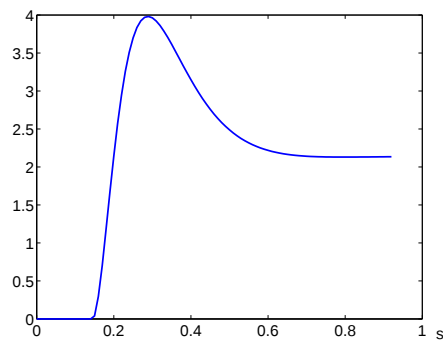
temps de calcul optimisation (s)



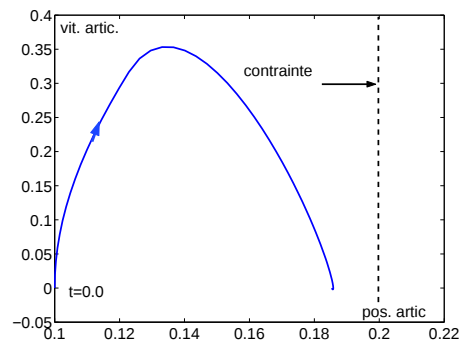
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



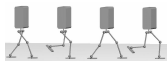
couple articulaire Γ (Nm)



fonction objectif final

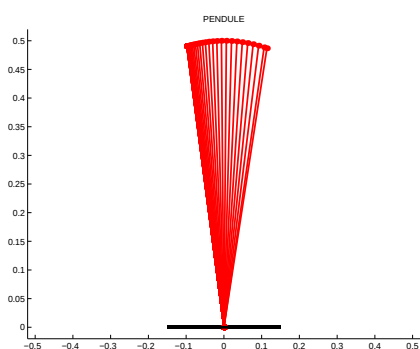


plan de phase

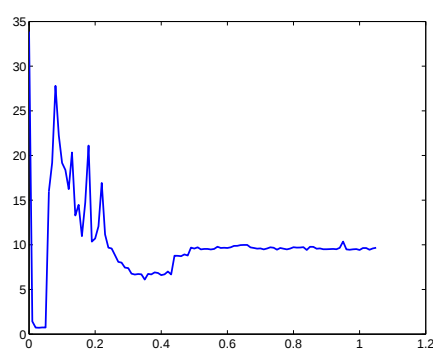


$simu_{43}^b$ - conditions initiales $\notin$ domaine admissible

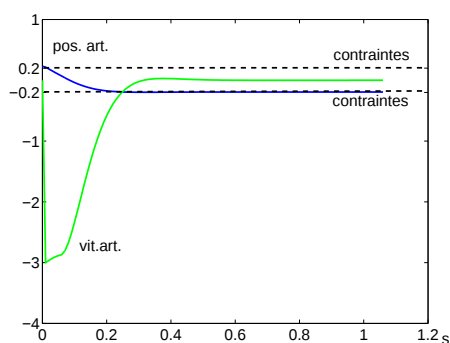
	Critère à minimiser :	$\frac{1}{2}U^T U$
	Contraintes à satisfaire :	$ q \leq 0.2$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	FSQP
	Environnement :	scilab
★	Conditions initiales :	$q_0 = 0.23, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



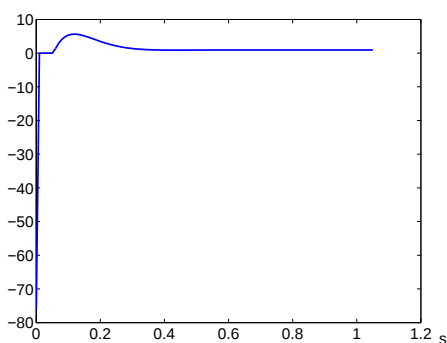
comportement du pendule



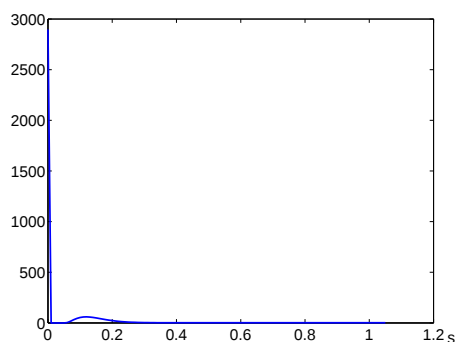
temps de calcul optimisation (s)



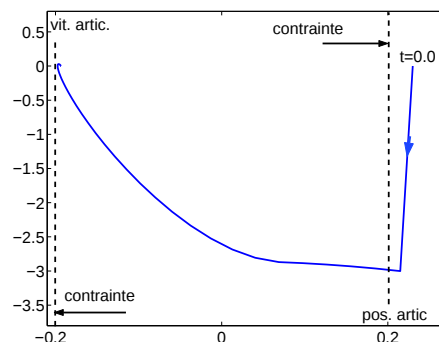
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



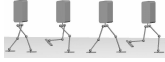
couple articulaire Γ (Nm)



fonction objectif final

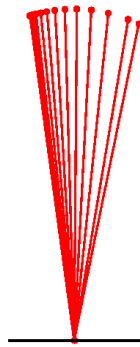


plan de phase

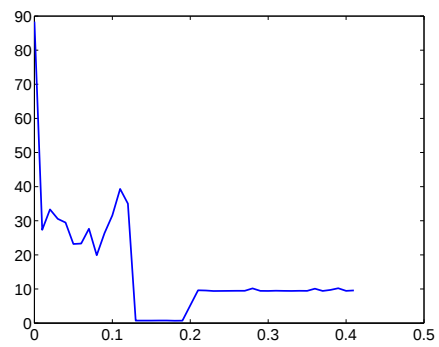


$simu_{44}^b$ - conditions initiales $\notin$ domaine admissible

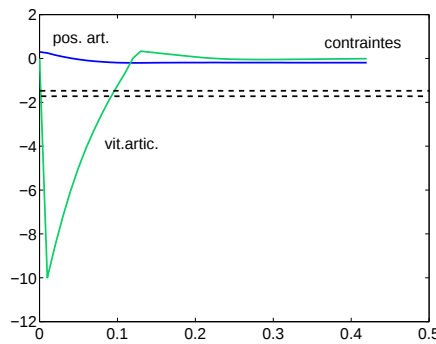
	Critère à minimiser :	$\frac{1}{2}U^T U$
	Contraintes à satisfaire :	$ q \leq 0.2$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	FSQP
	Environnement :	scilab
★	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



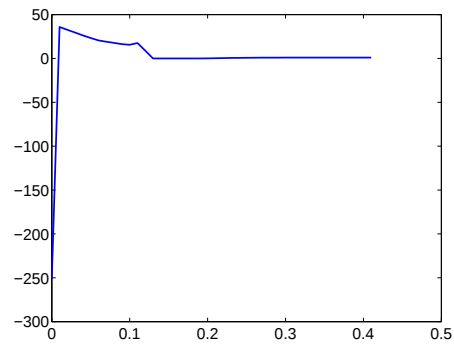
comportement du pendule



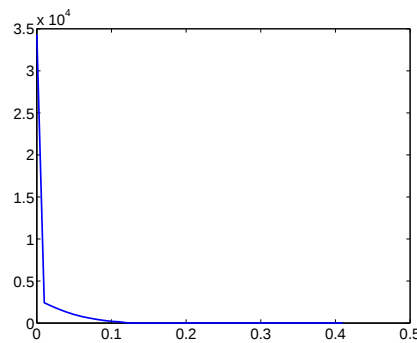
temps de calcul optimisation (s)



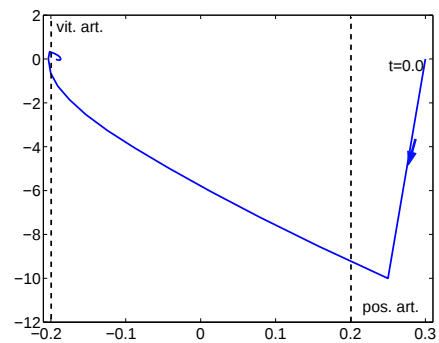
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



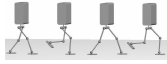
couple articulaire Γ (Nm)



fonction objectif final



plan de phase



Problème 4 - optimisation de la somme quadratique des couples et de l'état final sous contraintes

Le problème consiste à :

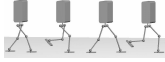
$$\begin{aligned}
 &\text{résoudre :} && \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} + X_{N_p}^T P X_{N_p} \\
 &\text{soumis à :} && q_{min} \leq q \leq q_{max} \\
 &&& \dot{q}_{min} \leq \dot{q} \leq \dot{q}_{max} \\
 &&& \Gamma_{min} \leq \Gamma \leq \Gamma_{max} \\
 &\text{initialisé avec :} && U_{k+1}^O = [\Gamma_{k+1|k}^* \dots \Gamma_{k+N_p-1|k}^*; \Gamma_{k+N_p-1|k}^*]^T
 \end{aligned}$$

avec : $q_{min} = -0.2$, $q_{max} = 0.2$, $\dot{q}_{min} = -2$, $\dot{q}_{max} = 2$, $\Gamma_{min} = -100$, $\Gamma_{max} = 100$.

Le problème est un problème d'optimisation sous contraintes, on utilise un algorithme SQP : *fmincon* sous Matlab.

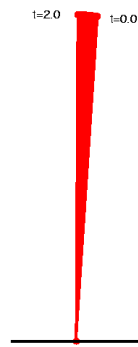
Description des simulations :

- *simu₄₃* - les conditions initiales du système satisfont les contraintes
- *simu₄₄* - les conditions initiales du système ne satisfont pas les contraintes

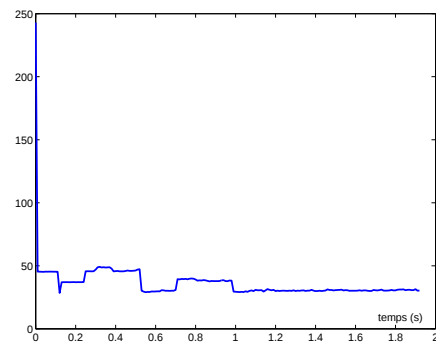


*simu*₄₃ - conditions initiales ∈ domaine admissible

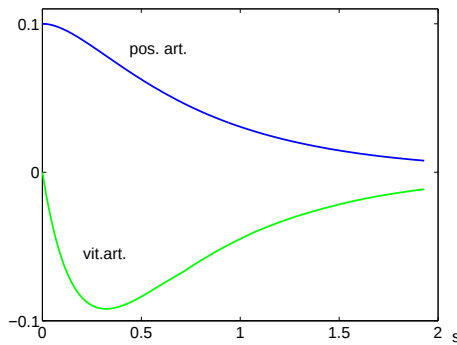
★	Critère à minimiser :	$\frac{1}{2}U^TU + X_{N_p}^TPX_{N_p}$
	Contraintes à satisfaire :	$ q \leq 0.2, \dot{q} \leq 2, \Gamma \leq 100$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	SQP
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



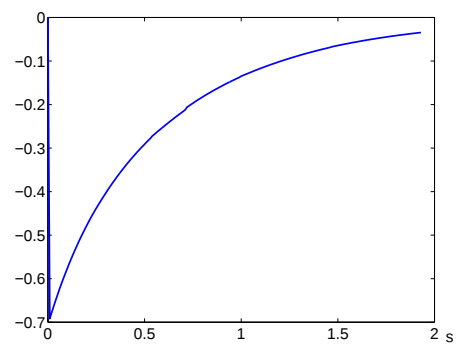
comportement du pendule



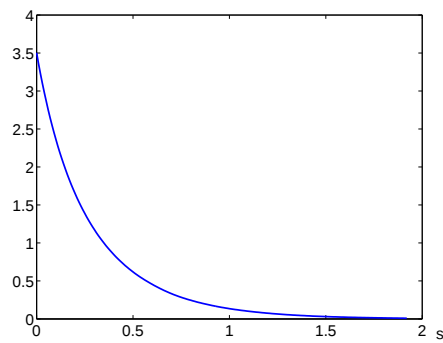
temps de calcul optimisation (s)



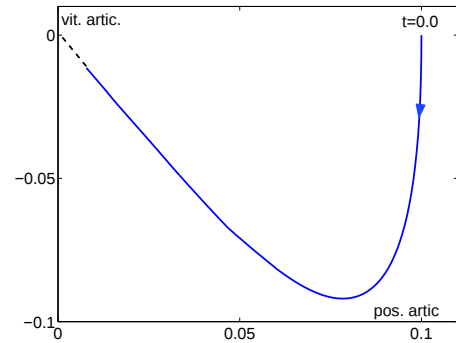
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



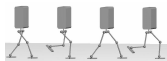
couple articulaire Γ (Nm)



fonction objectif final

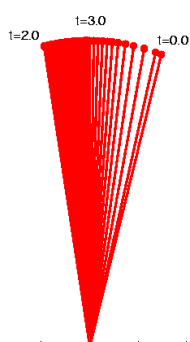


plan de phase

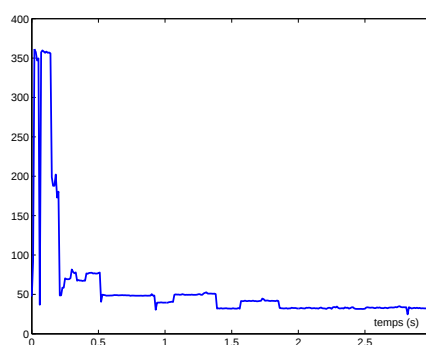


$simu_{44}$ - conditions initiales $\notin$ domaine admissible

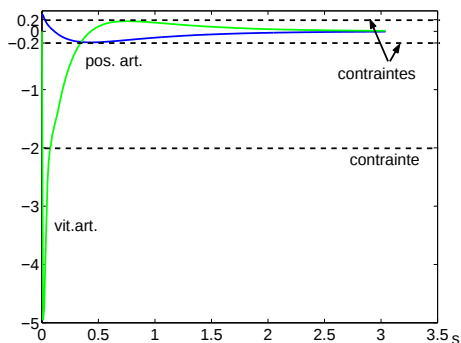
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p}$
	Contraintes à satisfaire :	$ q \leq 0.2, \dot{q} \leq 2, \Gamma \leq 100$
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	SQP
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



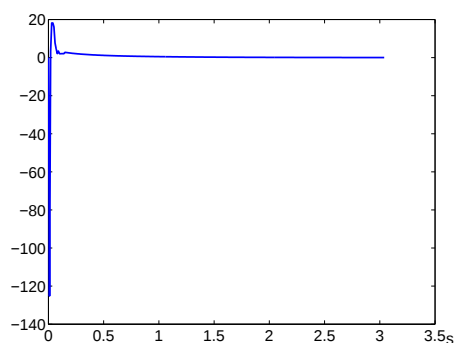
comportement du pendule



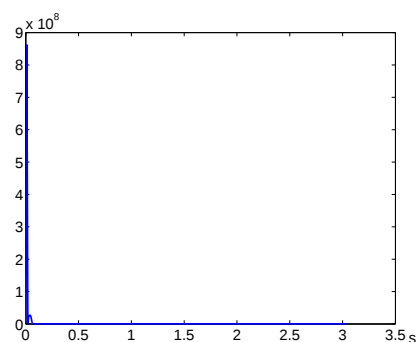
temps de calcul optimisation (s)



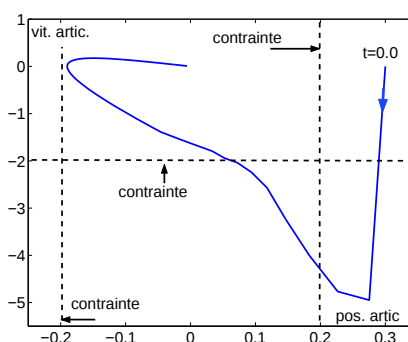
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



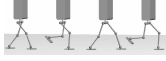
couple articulaire Γ (Nm)



fonction objectif final



plan de phase



Problème 5 - optimisation de la somme quadratique des couples avec pénalités

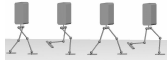
Le problème consiste à :

$$\begin{aligned}
 \text{résoudre :} \quad & \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} - p \sum \frac{1}{(-U_{i|k+1} + U_{min})} - p \sum \frac{1}{(U_{i|k+1} - U_{max})} \\
 & - p \sum \frac{1}{(-q_{i|k+1} + q_{min})} - p \sum \frac{1}{(q_{i|k+1} - q_{max})} \\
 & - p \sum \frac{1}{(-\dot{q}_{i|k+1} + \dot{q}_{min})} - p \sum \frac{1}{(\dot{q}_{i|k+1} - \dot{q}_{max})} \\
 \text{initialisé avec :} \quad & U_{k+1}^O = [\Gamma_{k+1|k}^* ; \dots ; \Gamma_{k+N_p-1|k}^* ; \Gamma_{k+N_p-1|k}^*]^T
 \end{aligned}$$

Le problème est un problème d'optimisation sans contraintes, on utilise un algorithme QN : *fminunc* sous Matlab.

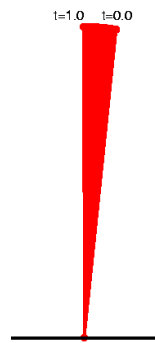
Description des simulations :

- *simu_{5₁}* - les conditions initiales du système satisfont les contraintes
- *simu_{5₂}* - les conditions initiales du système ne satisfont pas les contraintes

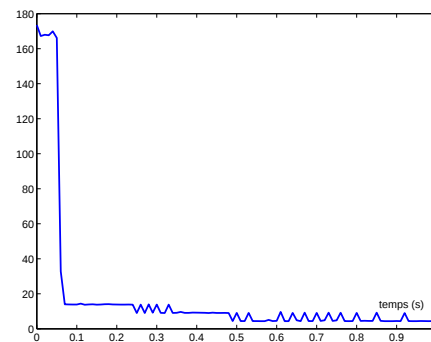


$simu_{51}$ - conditions initiales $\in$ domaine admissible

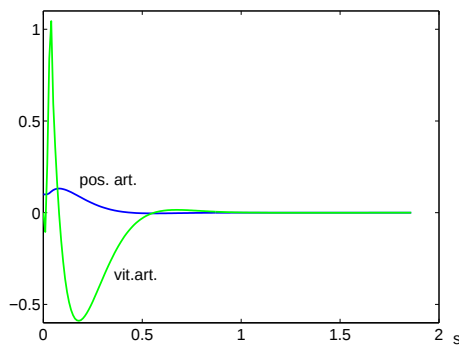
★	Critère à minimiser :	$\frac{1}{2}U^T U - 1. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



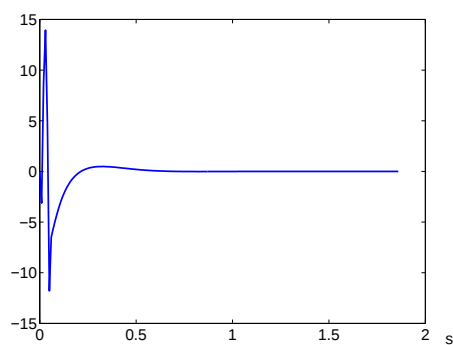
comportement du pendule



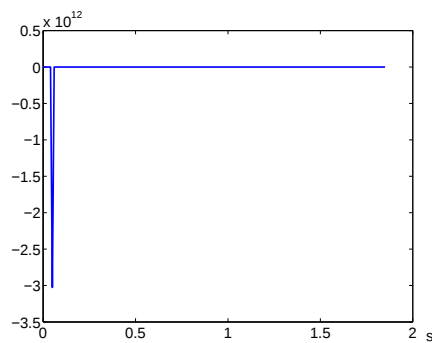
temps de calcul optimisation (s)



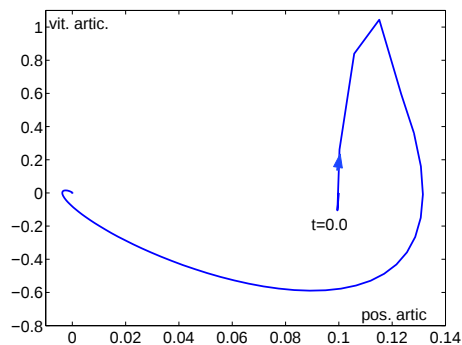
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



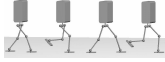
couple articulaire Γ (Nm)



fonction objectif final

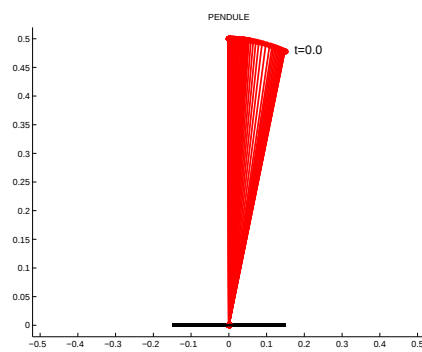


plan de phase

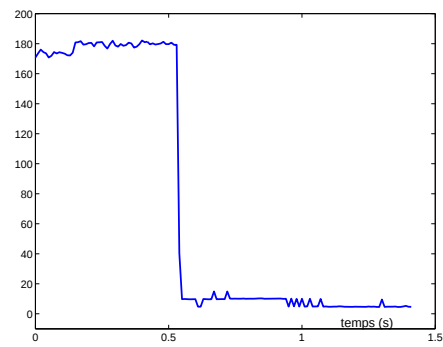


*simu*₅₂ - conditions initiales $\notin$ domaine admissible

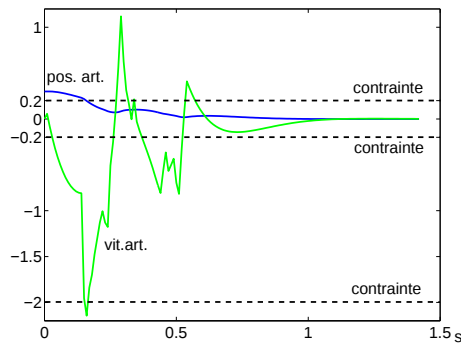
	Critère à minimiser :	$\frac{1}{2}U^T U - 1. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



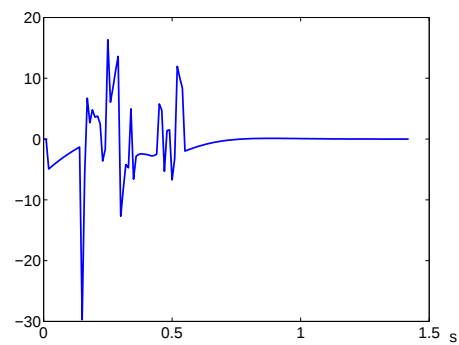
comportement du pendule



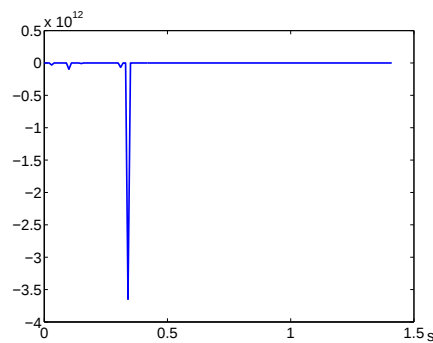
temps de calcul optimisation (s)



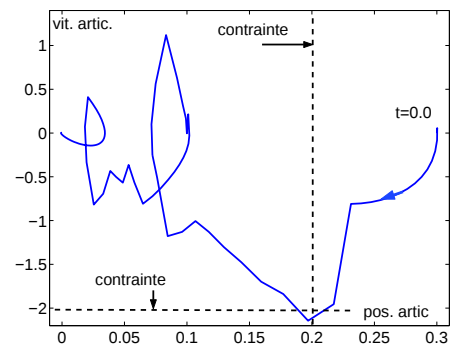
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



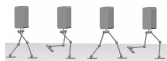
couple articulaire Γ (Nm)



fonction objectif final



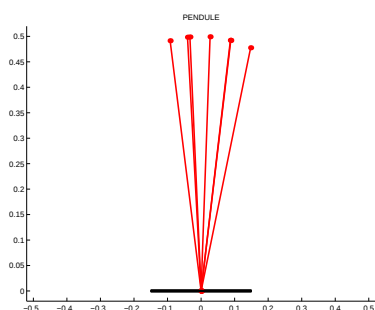
plan de phase



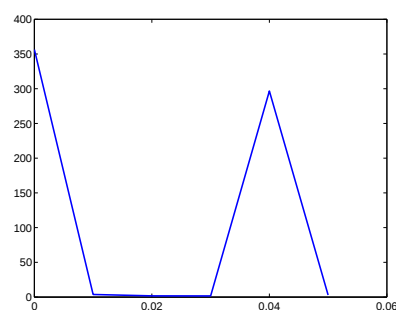
*simu*₅₃ - on fait évoluer le poids p dynamiquement

on multiplie p par 10 jusqu'à ce que les contraintes soient satisfaites.

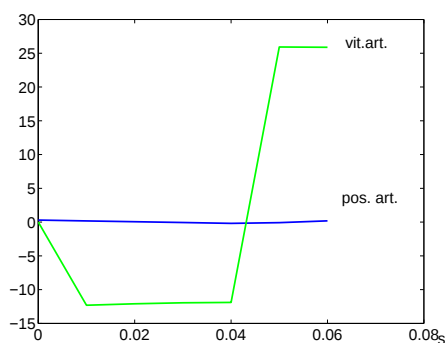
★	Critère à minimiser :	$\frac{1}{2}U^T U - p(t) \cdot \sum_{i=1}^2 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



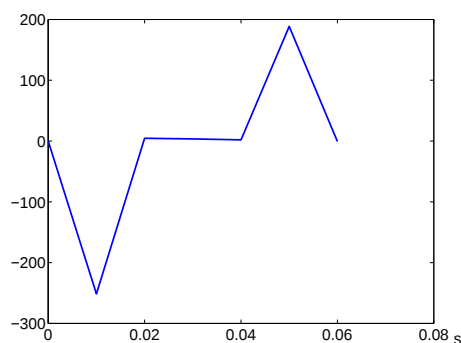
comportement du pendule



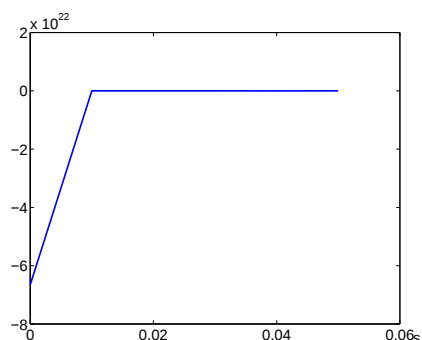
temps de calcul optimisation (s)



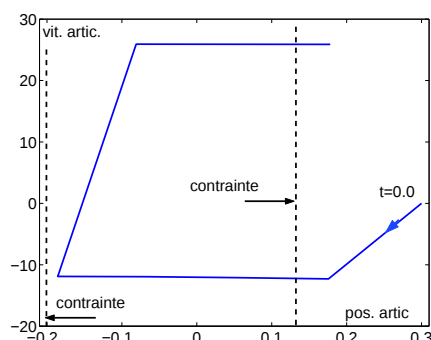
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



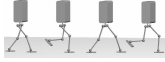
couple articulaire Γ (Nm)



fonction objectif final



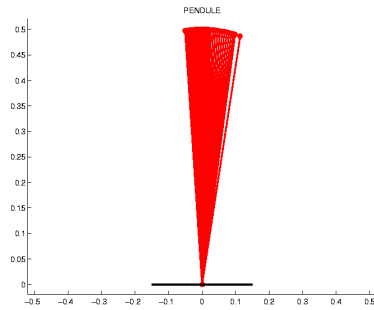
plan de phase



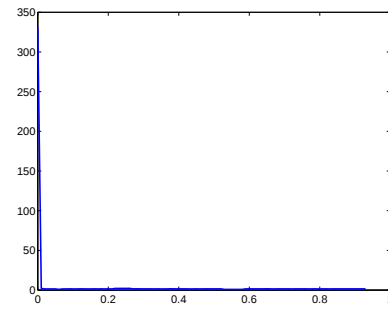
*simu*₅₄ - on fait évoluer le poids p dynamiquement

on multiplie p par 1000 jusqu'à ce que les contraintes soient satisfaites.

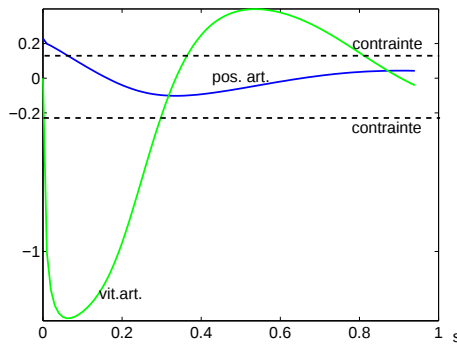
★	Critère à minimiser :	$\frac{1}{2}U^T U - p(t) \cdot \sum_{i=1}^2 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.23, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



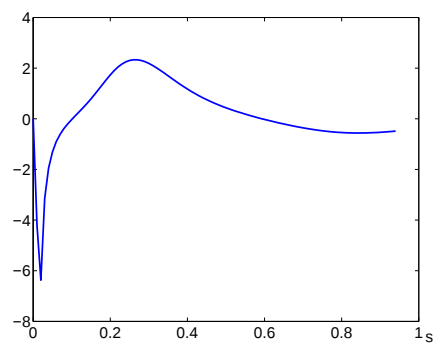
comportement du pendule



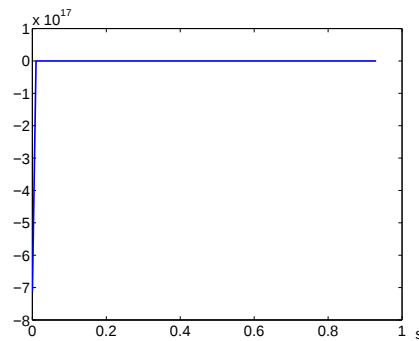
temps de calcul optimisation (s)



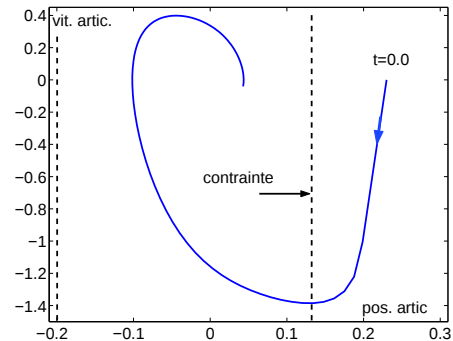
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



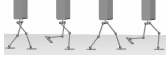
couple articulaire Γ (Nm)



fonction objectif final



plan de phase



Problème 6 - optimisation de la somme quadratique des couples et de l'état final avec pénalités

Le problème consiste à :

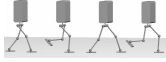
$$\begin{aligned} \text{résoudre :} \quad & \min_{U_{k+1}} \frac{1}{2} U_{k+1}^T U_{k+1} + X_{N_p}^T P X_{N_p} - p_1 \sum \frac{1}{(-U_{i|k+1} + U_{min})} - p_1 \sum \frac{1}{(U_{i|k+1} - U_{max})} \\ & - p_2 \sum \frac{1}{(-q_{i|k+1} + q_{min})} - p_2 \sum \frac{1}{(q_{i|k+1} - q_{max})} \\ & - p_3 \sum \frac{1}{(-\dot{q}_{i|k+1} + \dot{q}_{min})} - p_3 \sum \frac{1}{(\dot{q}_{i|k+1} - \dot{q}_{max})} \\ \text{initialisé avec :} \quad & U_{k+1}^O = [\Gamma_{k+1|k}^* \dots \Gamma_{k+N_p-1|k}^* \Gamma_{k+N_p-1|k}^*]^T \end{aligned}$$

Le problème est un problème d'optimisation sans contraintes, on utilise un algorithme QN : *fminunc* sous Matlab.

Description des simulations :

- *simu*₆₁ - les conditions initiales du système satisfont les contraintes
- *simu*₆₂ - les conditions initiales du système ne satisfont pas les contraintes
- *simu*₆₃ - le poids des pénalités passe de $p_i = 1$ à $p_i = 10$
- *simu*₇₁ - l'algorithme d'optimisation est limité à 100 itérations
- *simu*₇₂ - l'algorithme d'optimisation est limité à 10 itérations
- *simu*₈₁ - l'algorithme d'optimisation est initialisé avec $U_{k+1}^O = [0 \dots 0]$
- *simu*₈₂ - l'algorithme d'optimisation est initialisé avec $U_{k+1}^O = [\Gamma_k; \dots \Gamma_k; \Gamma_k]^T$
- *simu*₉₁ - le coefficient de frottement f est estimé 10 fois plus petit que le coefficient réel.
- *simu*₉₂ - on reprend *simu*₉₁ en faisant passer N_p de 30 à 60

Le problème est un problème d'optimisation sans contraintes, on utilise un algorithme QN : *fminunc* sous Matlab.

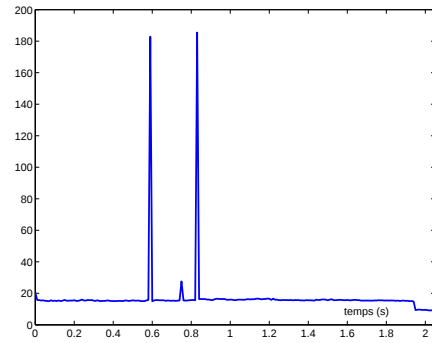


$simu_{61}$ - conditions initiales $\in$ domaine admissible

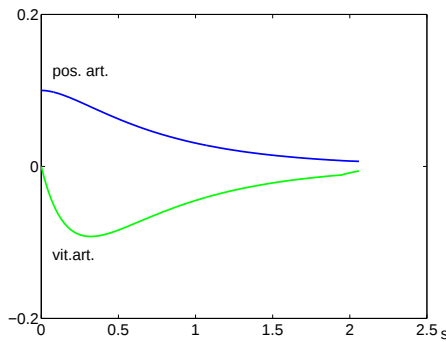
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 1. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
★	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



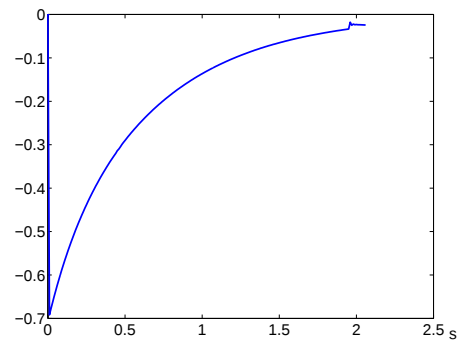
comportement du pendule



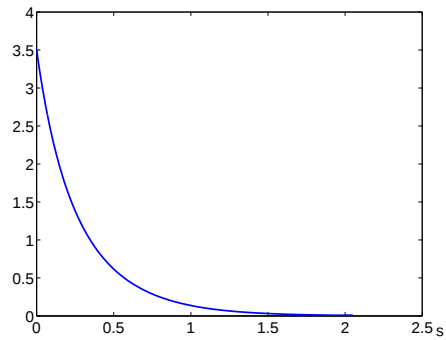
temps de calcul optimisation (s)



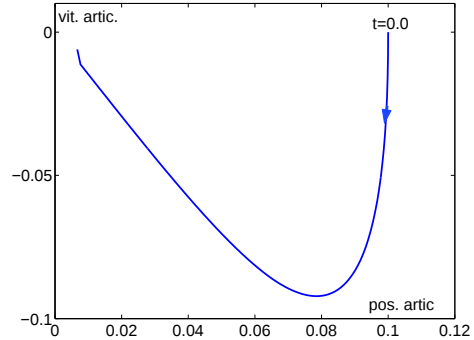
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



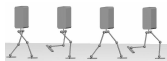
couple articulaire Γ (Nm)



fonction objectif final

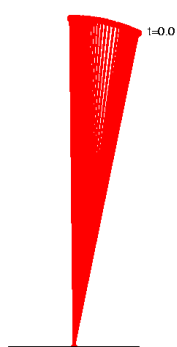


plan de phase

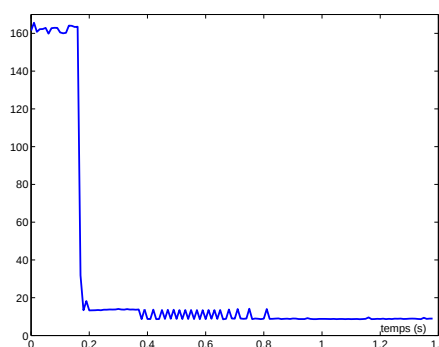


$simu_{62}$ - conditions initiales $\notin$ domaine admissible

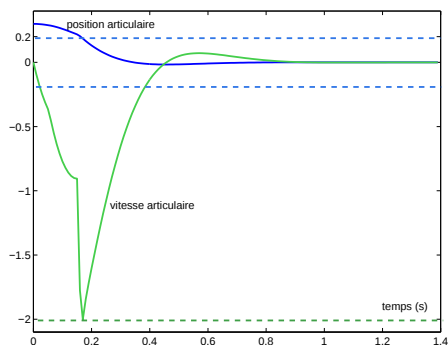
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 1. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



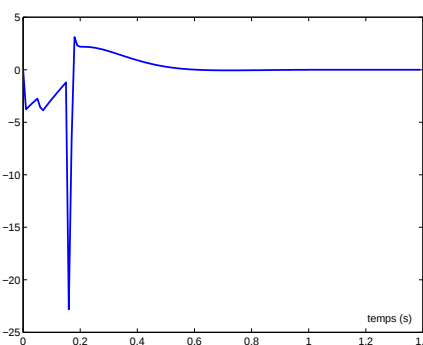
comportement du pendule



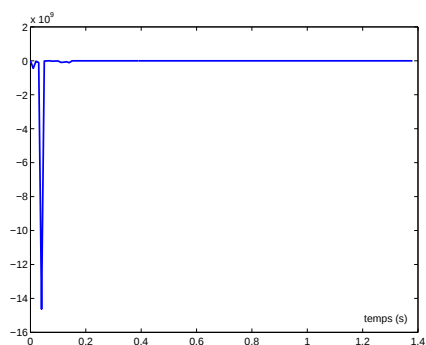
temps de calcul optimisation (s)



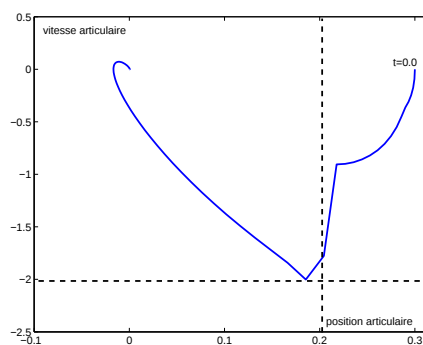
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



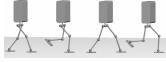
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

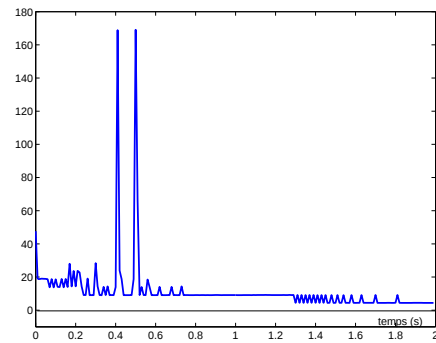


*simu*₆₃ - augmentation de p $1 \rightarrow 10$

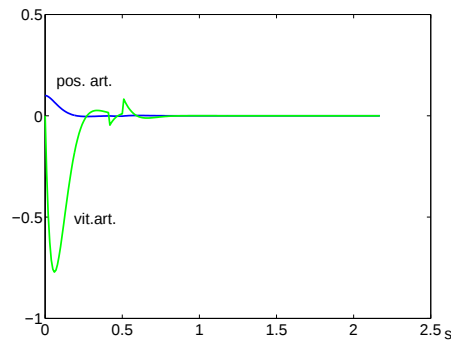
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10 \cdot \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	quasi-newton
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.3, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



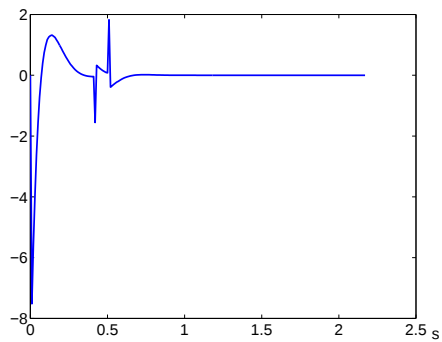
comportement du pendule



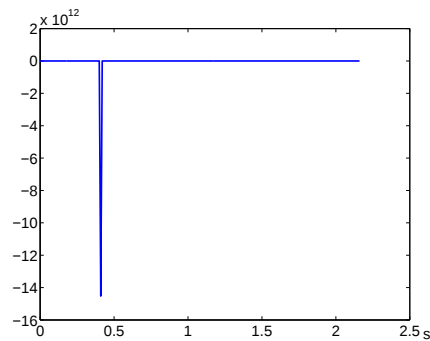
temps de calcul optimisation (s)



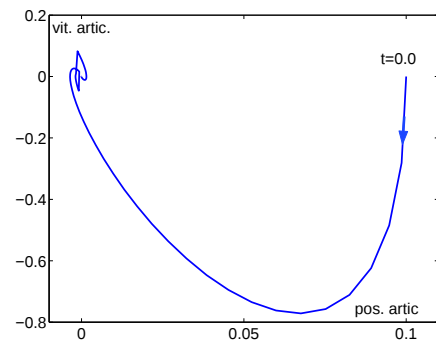
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



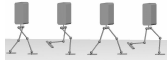
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

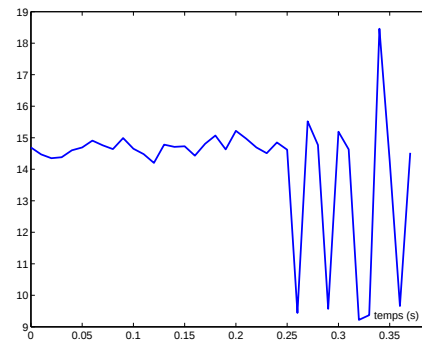


*simu*₇₁ - limitation des itérations de l'algo. *it* = 100

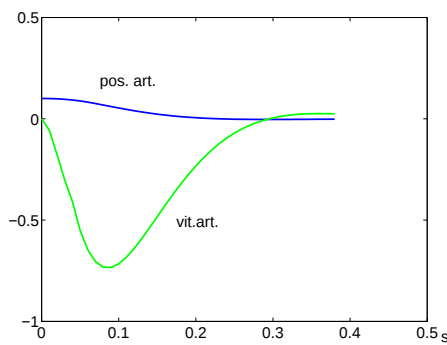
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	QN le nombre d'itération est limité à 100
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



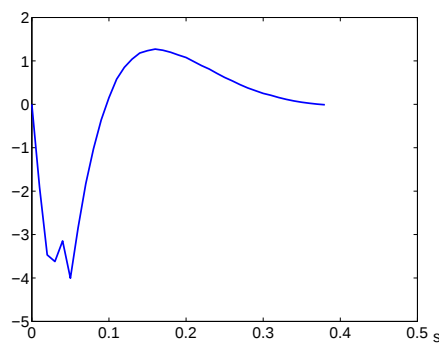
comportement du pendule



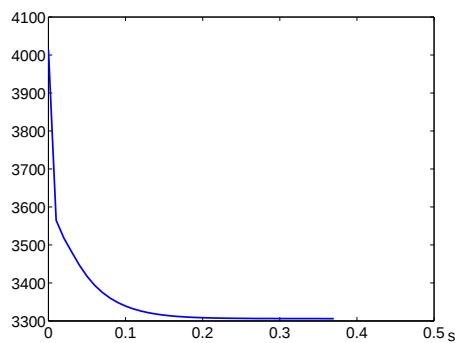
temps de calcul optimisation (s)



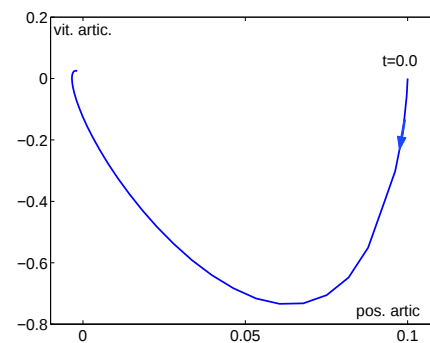
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



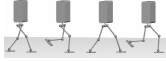
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

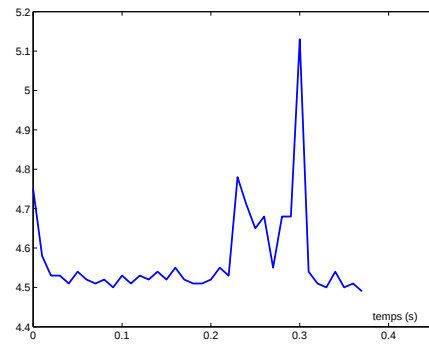


*simu*₇₂ - limitation des itérations de l'algo. *it* = 10

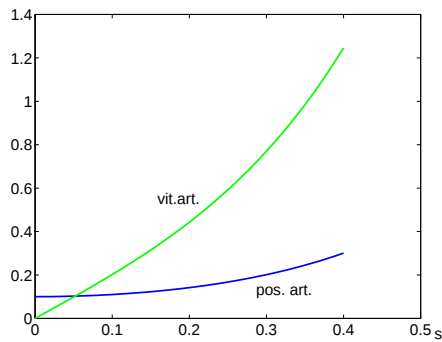
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10 \cdot \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
	Algorithme d'optimisation :	QN le nombre d'itération est limité à 10
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



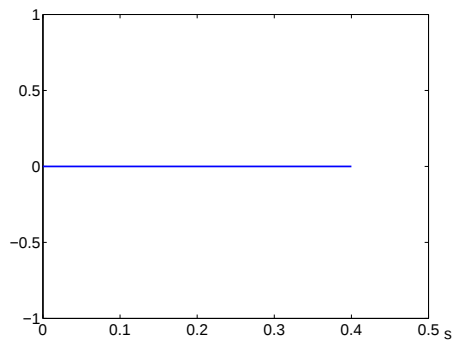
comportement du pendule



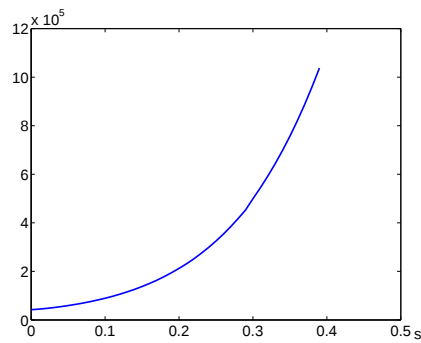
temps de calcul optimisation (s)



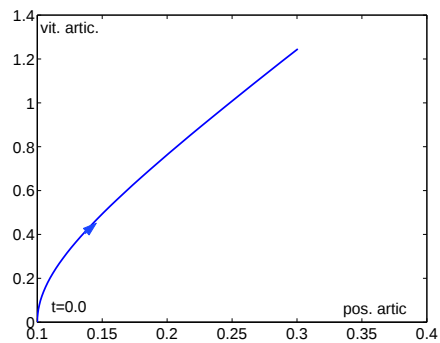
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



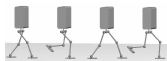
couple articulaire Γ (Nm)



fonction objectif final



plan de phase

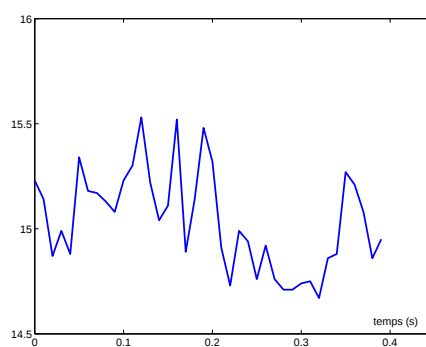


*simu*₈₁ - initialisation défavorable de l'optimisation

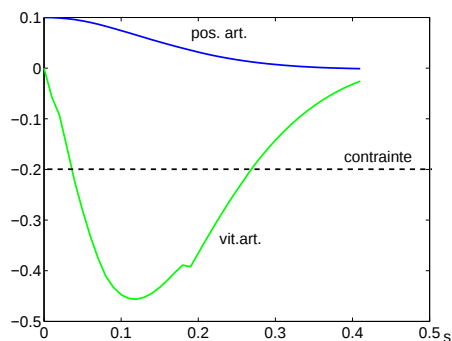
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10. \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[0 \dots 0]$
	Algorithme d'optimisation :	QN le nombre d'itération est limité à 100
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



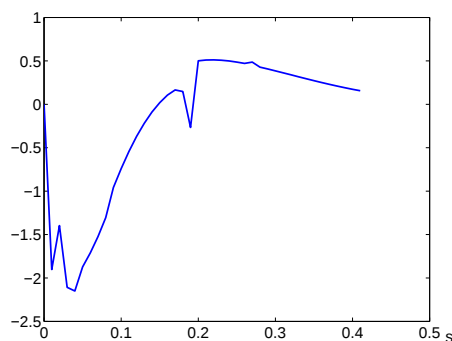
comportement du pendule



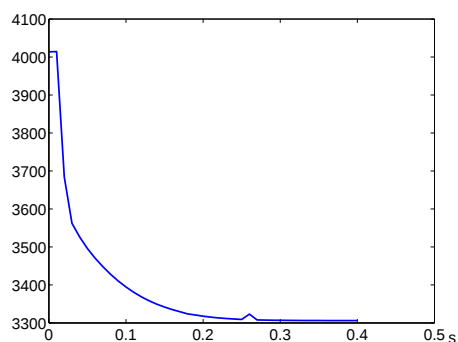
temps de calcul optimisation (s)



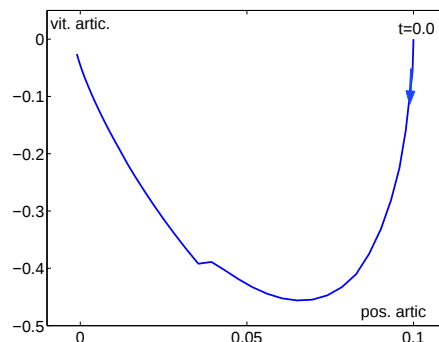
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



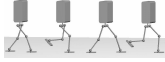
couple articulaire Γ (Nm)



fonction objectif final

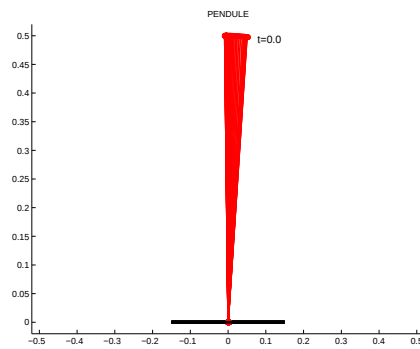


plan de phase

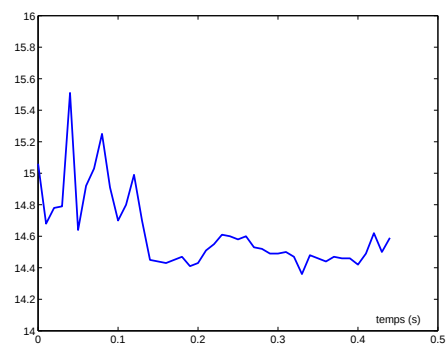


*simu*₈₂ - initialisation défavorable de l'optimisation

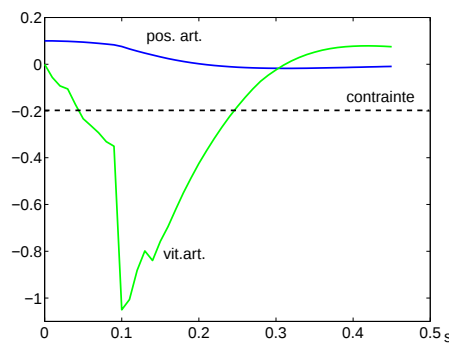
★	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10 \cdot \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_k \dots \Gamma_k]$
	Algorithme d'optimisation :	QN le nombre d'itération est limité à 100
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30



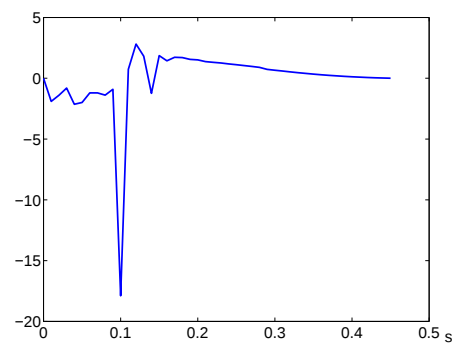
comportement du pendule



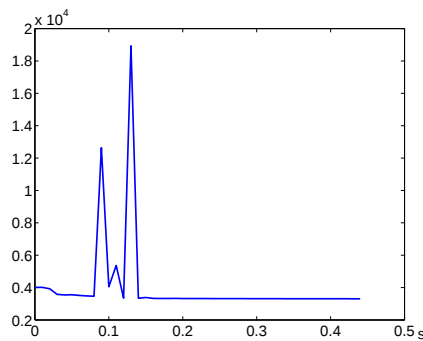
temps de calcul optimisation (s)



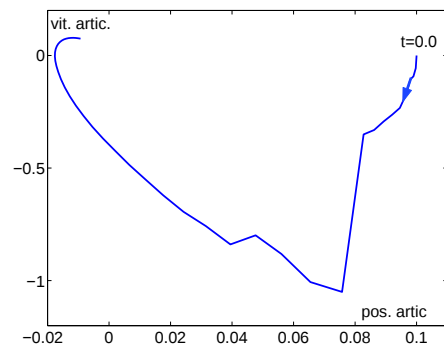
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



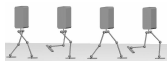
couple articulaire Γ (Nm)



fonction objectif final

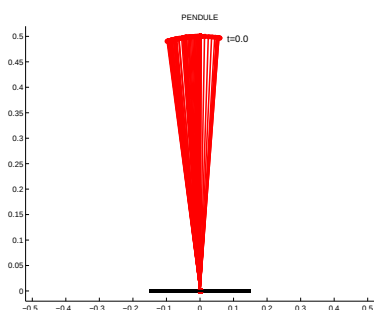


plan de phase

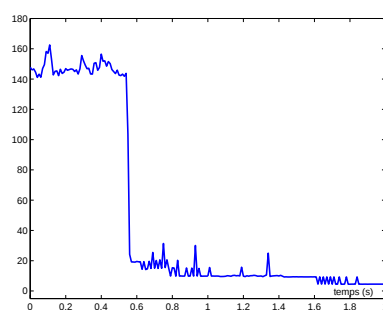


*simu*₉₁ - Erreurs de modèle $N_p = 30$

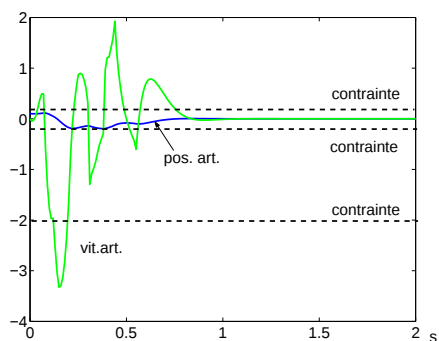
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10 \cdot \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
★	Algorithme d'optimisation :	QN le nombre d'itération est limité à 100
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
	Horizon de prédiction :	30
★	coefficient frottement réel 0.01	coefficient frottement estimé 0.001



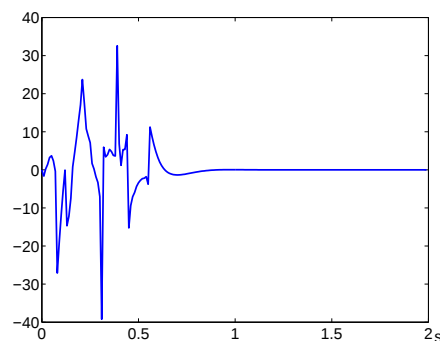
comportement du pendule



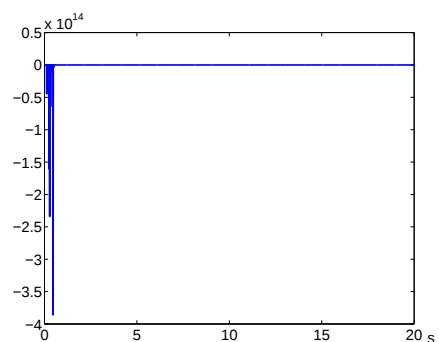
temps de calcul optimisation (s)



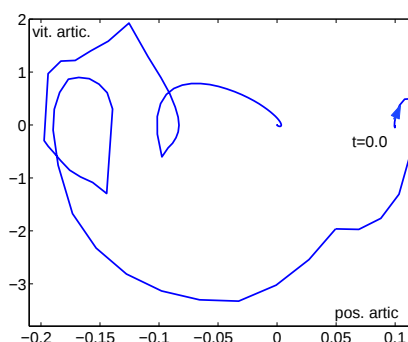
position q (rad) et vitesse $\dot{q}$ (rad.s^{-1})



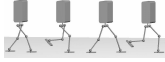
couple articulaire Γ (Nm)



fonction objectif final

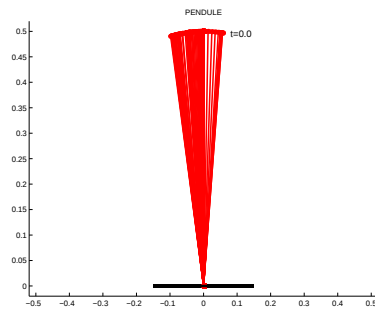


plan de phase

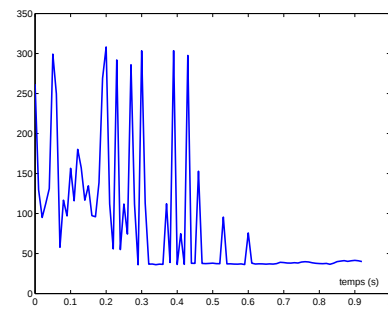


$simu_{9_2}$ - Erreurs de modèle $N_p = 60$

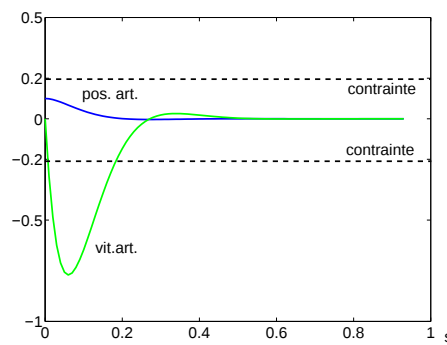
	Critère à minimiser :	$\frac{1}{2}U^T U + X_{N_p}^T P X_{N_p} - 10 \cdot \sum_{i=1}^6 1/g_i$
	Contraintes à satisfaire :	
	Initialisation de l'optimisation :	$[\Gamma_{k+1 k} \dots \Gamma_{k+N_p-1 k} \Gamma_{k+N_p-1 k}]$
★	Algorithme d'optimisation :	QN le nombre d'itération est limité à 100
	Environnement :	matlab
	Conditions initiales :	$q_0 = 0.1, \dot{q}_0 = 0, \Gamma_0 = 0$
★	Horizon de prédiction :	60
	coefficient frottement réel 0.01	coefficient frottement estimé 0.001



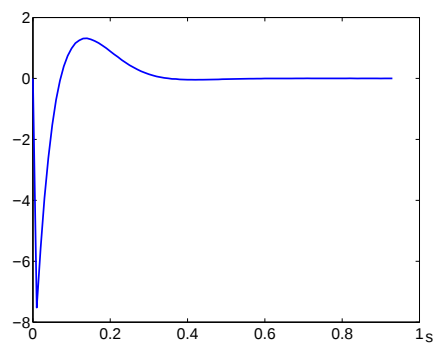
comportement du pendule



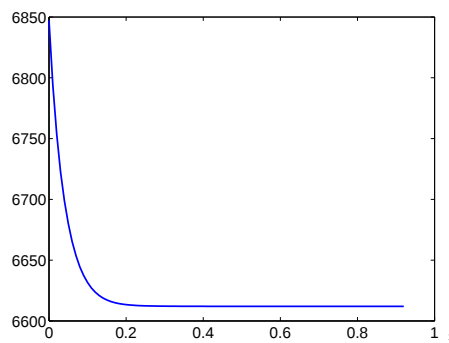
temps de calcul optimisation (s)



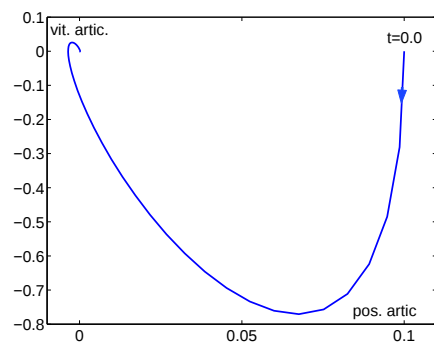
position q (rad) et vitesse $\dot{q}$ ($rad.s^{-1}$)



couple articulaire Γ (Nm)



fonction objectif final



plan de phase

Contrôle de la locomotion artificielle : de l'homme aux robots

Cette thèse est consacrée à la synthèse de la marche pour un robot bipède anthropomorphe. Ce problème est abordé en prenant comme référence la locomotion humaine. Le point de départ de notre travail est une analyse de la marche chez l'homme d'un point de vue descriptif (biomécanique) mais également explicatif (neurosciences et physiologie) ; l'objectif étant de mettre en évidence des éléments pertinents pour l'approche de la commande des robots. Les principes qui en sont déduits sont : l'absence de suivi de trajectoire articulaire ; la distinction explicite entre contrôle de la posture et commande de la marche ; l'utilisation de critères d'optimisation ; l'adaptation en ligne et l'anticipation. L'aspect postural (statique) de la locomotion a été abordé d'un point de vue expérimental sur le robot bipède BIP de l'INRIA. Un générateur de déplacements paramétrés a été développé. Un contrôleur a été réalisé et implémenté pour le suivi des mouvements générés. Ce travail a conduit aux premiers résultats de marche statique plane et de mouvements posturaux 3D. Différentes allures de marche 2D ont été exécutées et ont fait l'objet d'une analyse comparative afin d'évaluer l'influence des paramètres de marche. L'aspect dynamique de la locomotion s'est largement appuyé sur l'analyse de la locomotion humaine qui nous a conduit à proposer un schéma original, de type commande prédictive non linéaire sous contraintes : le mouvement est spécifié de façon implicite par des contraintes inégalités. Ce travail théorique et algorithmique est validé par des résultats de marche obtenus en simulation sur diverses structures de robots.

Mots-clés : robots bipèdes, trajectoires paramétrées, commande prédictive non linéaire, optimisation sous contraintes, marche artificielle.

Artificial locomotion control : from human to robots

This thesis is concerned with the synthesis of walking for an anthropomorphic bipedal robot. This problem is approached with human locomotion as a reference. The starting point of our work is an analysis of human walking from descriptive (biomechanics) as well as explicative (neurosciences and physiology) points of view ; the objective is to stress the relevant elements for the approach of robot control. The resulting control principles are : no joint trajectory tracking ; explicit distinction between postural control and walking control ; use of optimisation objectives ; on line adaptation and anticipation. The postural aspect (static) of walking is approached experimentally on the BIP robot of INRIA. A parameterized walking gait generator has been developed. A controller has been realized and implemented to track the generated displacements. This work leads to the first results of planar static walking and 3D postural movements. Several 2D walking gaits have been executed and a comparative study has been realized in order to evaluate the relative influence of gait parameters. The dynamical aspect of locomotion is widely inspired from the human walking analysis and leads to the proposition of an original scheme based on predictive non linear constrained control : the movement is specified implicitly with inequality constraints. This work is validated by simulation results obtained for several robot structures.

Keywords : biped robots, parametrized trajectories, nonlinear predictive control, optimization under constraints, artificial walking.