

Course bio-inspirée pour un robot humanoïde

Mémoire présenté par
Bruno SOMERS

pour l'obtention du grade de master en
ingénieur civil électromécanicien
Option: mécatronique

Promoteur
Renaud RONSSE

Lecteurs
Nicolas VAN DER NOOT, Paul FISETTE

Année académique 2015-2016

Remerciements

Tout d'abord, je tiens à remercier le **Professeur Renaud Ronsse**, le promoteur de ce mémoire, pour ses conseils, son encadrement et ses idées innovantes.

J'aimerais également adresser un profond remerciement à **Nicolas Van der Noot** pour son suivi tout au long de ce projet, pour ses précieux conseils, pour le partage de ses connaissances, de son expérience du simulateur et de la locomotion humanoïde en général, ainsi que pour sa disponibilité.

De plus, ce projet n'aurait pas été possible sans les moyens de calcul fournis par le **Consortium des Équipements de Calcul Intensif (CÉCI)**, financé par le Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) sous la convention n° 2.5020.11.

Enfin, je voudrais remercier ma famille et mes amis pour leur soutien et leur relecture.

Bruno Somers

Août 2016

Table d'abréviations & symboles

Abréviations générales

| | |
|----------|---|
| ZMP | Zero Moment Point |
| CPG | Central Patern Generation |
| COMAN | COmpliant HuMANoid Platform |
| CMA-ES | Covariance Matrix Adaptation Evolution Strategies |
| CECI | Consortium des Équipements de Calcul Intensif |
| SIMBICON | SIMple BIped locomotion CONtrol |
| SLIP | Spring Loaded Inverted Pendulum |
| DDL | Degrés de liberté |
| IMU | Inertial Measurement Unit |
| MPI | Message Passing Interface |

Modèle musculaire de Hill

| | |
|-----------|---|
| MTU | Muscle-Tendon Unit |
| CE | Contractile Element |
| SE | Series elastic Element |
| l_{ce} | longueur de l'élément contractile [m] |
| PE | Parallel elastic Element |
| BE | Buffer elastic Element |
| l_{mtu} | longueur de l'unité muscle-tendon [m] |
| A_m | niveau d'activation du muscle |
| l_{se} | longueur de l'élément série élastique [m] |
| S_m | niveau de stimulation du muscle |

Paramètres caractérisant un muscle

| | |
|--------------|---|
| l_{opt} | longueur optimale de fibre [m] |
| l_{slack} | longueur du tendon détendu [m] |
| F_{max} | force isométrique maximale [N] |
| v_{max} | vitesse maximale de contraction normalisée par la longueur optimale [l_{opt}/s] |
| r_0 | bras de levier constant [m] |
| ρ | facteur de pénétration $[-]$ |
| ϕ_{ref} | angle auquel $l_{mtu} = l_{opt} + l_{slack}$ [rad] |
| ϕ_{max} | angle produisant le bras de levier maximal [rad] |

Abréviations des muscles

| | |
|-----|-----------------------------|
| RF | Rectus Femoris |
| SOL | SOLeus |
| TA | Tibialis Anterior |
| VAS | VASti |
| GAS | GAStrocnemius |
| GLU | GLUteus group |
| HFL | Hip FLexor group |
| HAM | biarticular HAMstring group |
| BET | Back ExTension |
| BFL | Back FLexion |
| BTR | Back Tilt Right |
| BTL | Back Tilt Left |
| BRR | Back Rotation Right |
| BRL | Back Rotation Left |
| SET | Shoulder ExTension |
| SFL | Shoulder FLexion |
| SAB | Shoulder ABduction |
| SAD | Shoulder ADduction |
| SIR | Shoulder Internal Rotation |
| SER | Shoulder External Rotation |
| EFL | Elbow FLexion |
| EET | Elbow ExTension |

Table des matières

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Contexte | 1 |
| 1.2 | Résumé introductif | 2 |
| 1.3 | Analyse de la démarche humaine | 3 |
| 1.3.1 | Axes de référence chez l'être humain | 3 |
| 1.3.2 | Cycle de marche | 4 |
| 1.3.3 | Cycle de course | 4 |
| 2 | État de l'art | 7 |
| 2.1 | Robots humanoïdes | 7 |
| 2.2 | <i>Zero Moment Point</i> (ZMP) | 9 |
| 2.3 | Contrôleurs dynamiques | 10 |
| 2.3.1 | Cinématique inverse | 10 |
| 2.3.2 | Dynamique inverse | 10 |
| 2.3.3 | Optimisations de trajectoires | 11 |
| 2.3.4 | Contrôle <i>Task-Space</i> | 11 |
| 2.3.5 | Méthodes heuristiques | 11 |
| 2.3.6 | Méthodes de contrôle basées sur un modèle simple (<i>Simple-Model-Based</i>) . | 12 |
| 2.3.7 | Apprentissage par renforcement | 12 |
| 2.4 | Contrôleurs bio-inspirés | 12 |
| 2.4.1 | Modèle musculaire | 13 |
| 2.4.2 | Modèles de réflexes | 13 |
| 2.4.3 | <i>Central Pattern Generator</i> (CPG) | 13 |
| 2.5 | Vue d'ensemble | 14 |
| 3 | Robot COMAN & Outils | 15 |
| 3.1 | Robot COMAN | 15 |
| 3.1.1 | Présentation du robot | 15 |
| 3.1.2 | Modèles de pieds | 17 |
| 3.2 | Outils de simulation | 19 |
| 3.2.1 | Modèle Robotran | 19 |
| 3.2.2 | Partie simulateur | 20 |
| 3.2.3 | Partie contrôleur | 22 |
| 3.3 | Optimisations | 23 |
| 3.3.1 | Covariance Matrix Adaptation Evolution Strategy (CMA-ES) | 23 |
| 3.3.2 | Outils d'optimisation | 24 |
| 3.4 | Conclusion | 26 |

| | | |
|----------|--|-----------|
| 4 | Locomotion bio-inspirée | 27 |
| 4.1 | Vue générale du contrôleur | 27 |
| 4.2 | Modèle musculaire | 29 |
| 4.2.1 | Modèle musculaire de Hill | 29 |
| 4.2.2 | Activations des muscles | 30 |
| 4.2.3 | Couples appliqués aux articulations | 31 |
| 4.2.4 | Mise à échelle dynamique | 32 |
| 4.2.5 | Avantages et approximations du modèle | 32 |
| 4.3 | Muscles virtuels du COMAN | 33 |
| 4.3.1 | Muscles des jambes | 33 |
| 4.3.2 | Muscles des bras et du torse | 33 |
| 4.4 | Contrôle du torse et des bras | 35 |
| 4.5 | Réflexes bio-inspirés pour la marche et la course | 36 |
| 4.5.1 | Lois de contrôle | 37 |
| 4.5.2 | Phase d'appui | 38 |
| 4.5.3 | Phase oscillante | 40 |
| 4.5.4 | Paramètres à optimiser | 41 |
| 4.6 | Initialisation du COMAN pour la course | 41 |
| 4.7 | Fonctions d'objectif | 43 |
| 4.7.1 | Tâches de la fonction d'objectif | 43 |
| 4.7.2 | Modèle de l'énergie métabolique consommée | 44 |
| 4.8 | Conclusion | 45 |
| 5 | Résultats et analyses des simulations | 47 |
| 5.1 | Aperçu des résultats | 47 |
| 5.1.1 | Évolution des caractéristiques avec la vitesse d'avance | 48 |
| 5.1.2 | Symétrie de la locomotion | 50 |
| 5.2 | Analyse de la marche | 51 |
| 5.2.1 | Positions et couples des articulations sagittales des jambes | 52 |
| 5.2.2 | Forces de réaction du sol | 53 |
| 5.2.3 | Énergie mécanique du centre de masse | 53 |
| 5.2.4 | Énergie métabolique | 53 |
| 5.3 | Analyse de la course | 54 |
| 5.3.1 | Positions et couples des articulations sagittales des jambes | 55 |
| 5.3.2 | Contributions des différents muscles | 57 |
| 5.3.3 | Comparaison avec la marche | 58 |
| 5.4 | Marche rapide | 60 |
| 5.4.1 | Positions et couples des articulations sagittales des jambes | 60 |
| 5.4.2 | Activations des muscles | 61 |
| 5.4.3 | Forces de réaction du sol | 62 |
| 5.4.4 | Énergie mécanique du centre de masse | 62 |
| 5.4.5 | Énergie métabolique | 63 |
| 5.5 | Comparaison des modèles de pieds | 63 |
| 5.6 | Contrôleur musculaire du haut du corps | 65 |
| 5.7 | Conclusion | 66 |
| 6 | Discussion & conclusion | 69 |
| 6.1 | Évaluation des résultats | 69 |
| 6.2 | Limitations et perspectives | 70 |
| A | Améliorations du projet pour aboutir à des résultats de course | 71 |

| | |
|---|-----------|
| B Outils de simulation | 75 |
| B.1 CECI clusters | 75 |
| B.2 Algorithme CMA-ES | 75 |
| C Contrôleur | 77 |
| C.1 Relations force-longueur et force-vitesse d'un MTU | 77 |
| C.2 Caractérisation des muscles | 77 |
| C.2.1 Muscles des jambes | 77 |
| C.2.2 Muscles du haut du corps | 78 |
| C.3 Contrôleur musculaire du torse et des articulations non sagittales des bras | 79 |
| C.3.1 Bras | 79 |
| C.3.2 Torse | 79 |
| D Hill muscles characterisation | 81 |
| D.1 Goal | 81 |
| D.2 Muscles | 81 |
| D.3 Characteristics from OpenSim | 83 |
| D.4 Results for adult-like model | 85 |
| D.5 Dynamic scaling | 85 |
| D.6 Results for the COMAN | 86 |
| E Résultats et analyses | 89 |
| E.1 Contrôleur musculaire du torse et des articulations non sagittales des bras | 89 |
| Bibliographie | 91 |

Chapitre 1

Introduction

1.1 Contexte

Les robots mobiles sont de plus en plus présents dans de nombreux domaines dans la société actuelle : robots pour le transport (terrestres, aériens ou sous-marins), robots de surveillances, robots d'assistance, robots domestiques, robots de maintenance, robots guides ou d'accueil, robots militaires, etc.

L'environnement physique de la société est conçu par l'être humain. Par exemple, le dimensionnement d'un escalier ou d'une porte, la hauteur du plafond ou la largeur d'un couloir sont adaptés à la morphologie humaine ainsi qu'à la manière dont l'être humain se déplace. En conséquence, pour pouvoir évoluer dans cet environnement avec des robots mobiles, il faut soit l'adapter à ces robots, soit concevoir des robots humanoïdes qui ont la même morphologie que les humains. Les robots humanoïdes ont un grand avantage par rapport aux robots mobiles conventionnels munis de roues. L'utilisation de pieds et de jambes permet au robot de franchir des obstacles et d'évoluer sur une large variété de terrains. De plus, il n'y a pas d'animaux à roues dans la nature, contrairement aux animaux à pattes, ce qui démontre l'efficacité de ce moyen de locomotion. De même, la plupart des outils ont été créés pour être utilisés par l'homme : une chaise est adaptée à notre morphologie pour s'y asseoir, l'habitacle d'une voiture est conçu pour avoir facilement accès à toutes ses fonctionnalités, la forme d'une paire de ciseaux est adaptée aux doigts, etc.

Cependant, les robots humanoïdes sont difficiles à concevoir et à contrôler, notamment à cause de leurs nombreux degrés de liberté. Le robot utilisé dans le cadre de ce mémoire étant déjà conçu, nous nous intéresserons au contrôle de celui-ci.

En s'inspirant de la qualité de la démarche humaine, nous choisissons de suivre une démarche bio-inspiré pour concevoir le contrôleur présenté dans ce mémoire. Pour ce faire, nous nous basons sur les travaux de Nicolas Van der Noot qui a développé un contrôleur de marche dynamique bio-inspirée pour le robot COMAN, un robot d'un mètre de hauteur conçu par l'*Italian Institute of Technology* (IIT). Ce contrôleur est basé sur un modèle de réflexe des muscles introduit par Geyer et Herr [29]. Un outil de simulation a été développé pour modéliser ce robot et son environnement afin de tester le contrôleur avant d'éventuellement l'utiliser sur le vrai robot.

Le but de ce mémoire est de concevoir un contrôleur permettant d'obtenir une démarche de course semblable à celle d'un être humain pour le robot COMAN. Pour ce faire, nous nous basons sur le modèle de réflexe présenté par Wang et al dans [70]. Ce contrôleur est testé sur le simulateur conçu et implémenté par Nicolas Van der Noot.

1.2 Résumé introductif

Après la présentation du contexte et des objectifs de ce travail, cette section passe brièvement en revue ses différents chapitres.

Chapitre 2 - État de l'art

Ce chapitre permet de mettre en contexte le contrôleur utilisé dans le cadre de ce mémoire. Un bref historique des robots humanoïdes est présenté pour se rendre compte de l'évolution faite dans ce domaine. Ensuite, la différence entre la marche statique et la marche dynamique est expliquée, en y introduisant la méthode de contrôle de marche la plus connue : le *Zero moment point*. Étant donné que ce mémoire s'intéresse à la course d'un robot, les différents types de contrôleurs dynamiques sont regroupés en catégorie afin d'avoir une vue d'ensemble des différentes techniques de contrôle utilisées dans de nombreux travaux de recherches (tant en simulation que sur de vrais robots). Une attention particulière est apportée aux méthodes bio-inspirées ainsi qu'aux travaux déjà effectués sur le COMAN.

Chapitre 3 - Robot COMAN & Outils

Ce chapitre a pour but d'introduire le modèle physique du robot utilisé, le COMAN, ainsi que l'environnement de simulation. Les caractéristiques du robot sont d'abord présentées. Une contribution de ce mémoire est l'ajout de modèles de pieds en simulations, qui sont décrits dans ce chapitre.

L'environnement de simulation est ensuite introduit. Celui-ci inclut tant le modèle physique du robot que son interaction avec l'environnement extérieur. Une attention particulière est apportée aux découplages des parties simulation et contrôleur afin de pouvoir transposer le projet sur le vrai robot. La partie contrôleur ne reçoit donc que des entrées disponibles sur le vrai robot.

Le simulateur est un outil indispensable pour tester le contrôleur, car des phases d'optimisation sont utilisées pour déterminer les valeurs optimales des paramètres du contrôleur. Pour des raisons de sécurité et de gains de temps, ces optimisations peuvent difficilement être effectuées sur le vrai robot. En fin de chapitre, l'algorithme et les outils utilisés lors de ces phases d'optimisation sont présentés.

Chapitre 4 - Locomotion bio-inspirée

Ce chapitre est consacré au contrôleur bio-inspiré utilisé pour obtenir des résultats de course. Le robot est contrôlé uniquement en deux dimensions, ce qui signifie qu'il ne peut pas tomber sur le côté. Le contrôle s'effectue en envoyant des stimulations à des muscles virtuels, qui génèrent des couples aux articulations auxquelles ils sont attachés. Le but du contrôleur est donc de déterminer quelles sont les stimulations à envoyer aux muscles pour obtenir la démarche souhaitée.

Tout d'abord, le modèle musculaire utilisé est développé. On introduit ensuite les muscles virtuels utilisés sur le COMAN tant pour les jambes (muscles déjà utilisés sur le COMAN) que pour le haut du corps (contribution de ce mémoire).

Le contrôleur en tant que tel est ensuite expliqué. Il se décompose en deux parties distinctes : le contrôle des degrés de liberté des jambes, et le celui des degrés de liberté du haut du corps. L'état initial du robot est également présenté dans ce chapitre.

Enfin, la fonction d'objectif utilisée lors des phases d'optimisation est détaillée. Celle-ci permet de quantifier la démarche obtenue par rapport à celle qui est souhaitée. Cette fonction a pour but de faire converger les optimisations vers la démarche voulue.

Chapitre 5 - Résultats et analyses des simulations

Dans ce chapitre, les résultats des simulations sont présentés et analysés. Afin de comparer la course avec la marche, celle-ci est tout d'abord explicitée. Les résultats de course obtenus sont ensuite comparés avec des données humaines pour évaluer celle-ci. De plus, le rôle et les contributions de chacun des muscles dans la course sont représentés. Différentes données sont par ailleurs analysées pour déterminer les caractéristiques propres à la course : les forces de réaction du pied avec le sol, l'énergie mécanique du centre de masse, et l'énergie métabolique consommée. Enfin, une comparaison des modèles de pieds utilisés est effectuée.

1.3 Analyse de la démarche humaine

Tout d'abord, les trois axes de références chez l'être humain sont introduits dans cette section, car nous les utiliserons fréquemment dans ce mémoire. Ensuite, les différentes étapes de la locomotion humaine sont brièvement expliquées. En effet, il est utile pour la suite de ce mémoire de connaître la démarche humaine étant donné que nous suivons une approche bio-inspirée pour concevoir le contrôleur. Afin de comparer les résultats de course avec ceux de marche, les deux cycles sont présentés.

1.3.1 Axes de référence chez l'être humain

Dans la suite de ce mémoire, on découpe les articulations en fonction de leurs axes de rotation. On parlera d'articulation sagittale, frontale ou transversale, en référence aux différents plans du corps humain qui sont représentés à la figure 1.1 (une articulation sagittale est définie comme ayant un axe de rotation normal au plan sagittal).

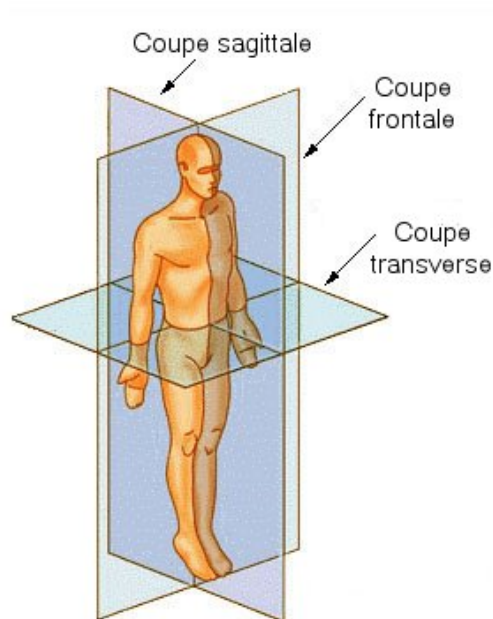


FIGURE 1.1 – Coupes et axes de référence d'un corps humain. Image issue de https://fr.wikipedia.org/wiki/Anatomie_humaine.

1.3.2 Cycle de marche

Un cycle de marche ou de course commence avec un contact initial d'un pied avec le sol, et finit par le contact initial suivant du même pied. Ce contact initial est l'impact du talon (*heel strike*). Le cycle peut être divisé en deux phases distinctes : la phase d'appui (*stance phase*) et la phase oscillante (*swing phase*). La phase d'appui est la phase où la jambe concernée est en contact avec le sol, tandis que le pied est dans les airs lors de la phase oscillante. Le texte de cette section se base sur [22].

Un cycle de marche est représenté à la figure 1.2. La phase d'appui représente environ 60% du cycle, tandis que la phase oscillante représente les 40% restants. Lorsque les deux pieds sont en contact avec le sol, on parle de double appui. Celui-ci est présent aux premiers et derniers 10% du cycle. Dès lors, le temps d'appui sur une seule jambe est égal au temps de la phase oscillante de l'autre jambe. Les différentes périodes de la marche sont également représentées à la figure 1.2. La phase de mise en charge (*loading reponse*) a pour rôle de transférer le poids vers la jambe d'appui et d'absorber les chocs. La phase de milieu d'appui (*mid-stance*) permet au corps d'avancer sur le pied d'appui. La phase de fin d'appui (*terminal stance*) permet l'avancée du corps jusqu'au contact du pied opposé avec le sol. La phase pré-oscillante (*pre-swing*) a pour but de propulser le corps vers l'avant et de transférer le poids vers la jambe en phase d'appui. Enfin, les phases de début d'oscillation (*initial swing*), de milieu d'oscillation (*mid-swing*) et de fin d'oscillation (*terminal swing*) ont pour but de permettre l'avancée du membre oscillant sans qu'il y ait de contact avec le sol.

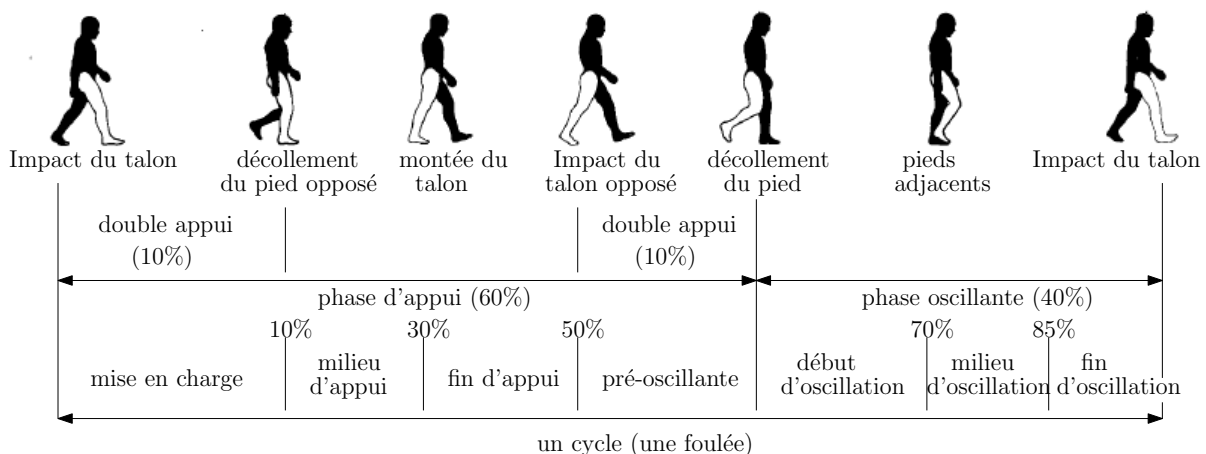


FIGURE 1.2 – Analyse d'un cycle de marche. La jambe considérée est celle de droite (blanche). Image adaptée de [1].

1.3.3 Cycle de course

Un cycle de course est représenté à la figure 1.3. Les deux différences principales avec le cycle de marche sont l'apparition de phases de vol où les pieds ne sont plus en contact avec le sol, et la disparition des phases de double appui. La phase d'appui représente moins de 50% du cycle, et plus la vitesse de course augmente, plus cette période est courte (pour les meilleurs sprinters, cette phase est inférieure à 22%) [43]. Le texte de cette section se base sur [47].

Lorsque la vitesse de course augmente, le contact initial se déplace de l'arrière du pied (le talon) à l'avant du pied. Cette différence marque la distinction entre la course et le sprint. La course est pratiquée sur une distance longue (endurance), tandis que le sprint a pour but de parcourir une petite distance le plus rapidement possible, sans se soucier de l'énergie consommée. Dans le cadre

de ce mémoire, seule la course est étudiée. Lors de la course, le contact initial s'effectue pour la majorité des coureurs (80%) avec l'arrière du pied comme pour la marche. Pour la plupart des 20% restants, l'impact du pied avec le sol s'effectue avec le milieu du pied.

Le cycle de course peut être décomposé en une alternance de périodes d'accélération et de décélération. Ces périodes ne coïncident pas avec le contact initial ou le décollement du pied. Pendant la phase d'amortissement, la hauteur du centre de masse diminue (depuis sa valeur maximale lors de la phase de vol). La vitesse horizontale du centre de masse diminue également. Durant la phase de propulsion, la position du centre de masse augmente horizontalement et verticalement.

Les différences cinématiques entre la marche et la course seront étudiées dans le chapitre 5. On remarque déjà visuellement que le genou est légèrement fléchi lors de la phase d'appui, et que les flexions de la hanche et du genou sont plus prononcées lors de la phase oscillante pendant la course.

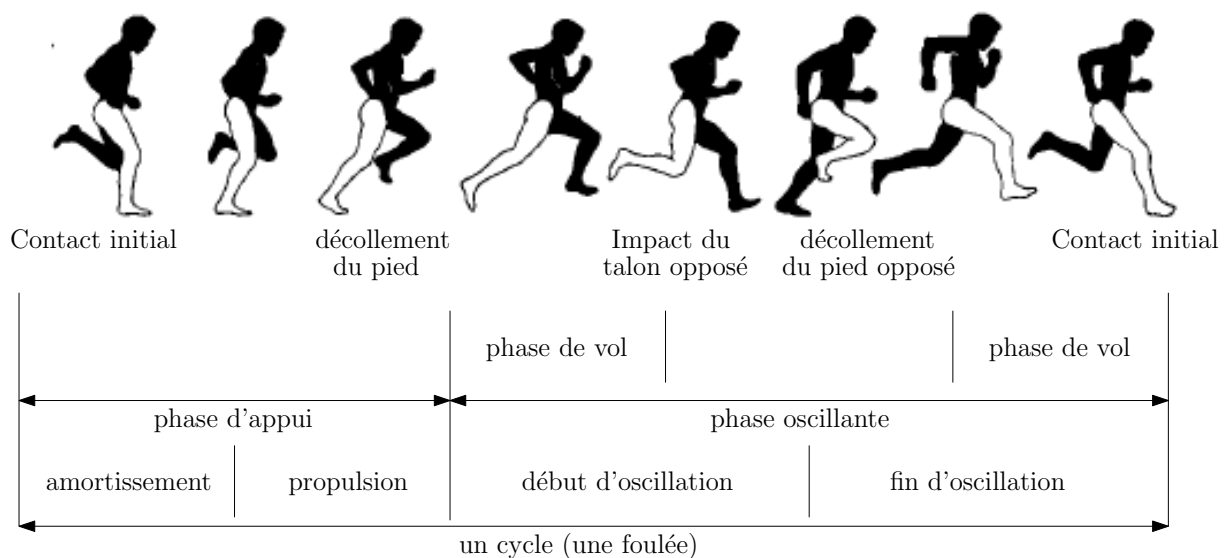


FIGURE 1.3 – Analyse d'un cycle de course. La jambe considérée est celle de droite (blanche). Image adaptée de [1].

Chapitre 2

État de l'art

Ce chapitre reprend un aperçu des différentes méthodes de contrôle développées pour la locomotion des robots humanoïdes, en se focalisant plus particulièrement sur la course afin d'avoir une vision globale de l'état de l'art en la matière. Nous commençons par un bref historique des robots humanoïdes afin de se rendre compte de l'évolution faite dans ce domaine. Le principe de stabilité basé sur le critère du *Zero Moment Point*, qui est utilisé dans la majorité des contrôleurs de marche, est brièvement expliqué. Ensuite, les différents types de contrôle dynamique sont regroupés en catégories. Le but de ce chapitre n'est pas d'expliquer en détail le contrôleur de tel ou tel robot (cela est complexe et chaque robot a ses spécificités et ses adaptations du contrôleur), mais de donner une vision de l'ensemble des méthodes utilisées. Enfin, une attention particulière sera apportée aux méthodes bio-inspirées ainsi qu'aux travaux déjà effectués sur le robot COMAN, le robot utilisé dans le cadre de ce mémoire.

2.1 Robots humanoïdes

L'un des premiers robots humanoïdes est le WABOT-1 développé par Ichiro Kato et ses collègues (Université de Waseda, Japon) en 1973 (figure 2.1a). Il était capable de marcher, reconnaître des objets, les manipuler, comprendre un langage et s'exprimer [40]. Une deuxième version du robot, WABOT-2, est apparue en 1984. Il était également capable de jouer du piano (figure 2.1b).



(a) WABOT-1 (1973)



(b) WABOT-2 (1984)

FIGURE 2.1 – Robots humanoïdes de l'université de Waseda (Japon). Images issues de http://www.humanoid.waseda.ac.jp/booklet/kato_2.html.

L'ère des robots humanoïdes n'a véritablement commencé qu'en 1996 avec le robot P2 de Honda (180 cm pour 210 kg, figure 2.2a). Il est issu d'un projet confidentiel *humanoid robot* lancé en 1986. Il est le premier robot qui peut marcher de manière stable de façon autonome. Honda présente ensuite le P3 en 1997 (160 cm pour 130 kg, figure 2.2b), puis ASIMO (*Advanced Step in Innovative Mobility*) en 2000 (120 cm pour 43kg, figure 2.2c). Il s'agit sans doute du robot humanoïde le plus connu du grand public par sa forte médiatisation. Le robot ASIMO a eu plusieurs évolutions (le dernier et sixième modèle date de 2014). Il a fallu attendre 2004 pour avoir les premiers résultats de course sur ce robot (3 km/h). Le dernier modèle peut courir à vitesse de 9km/h [12].

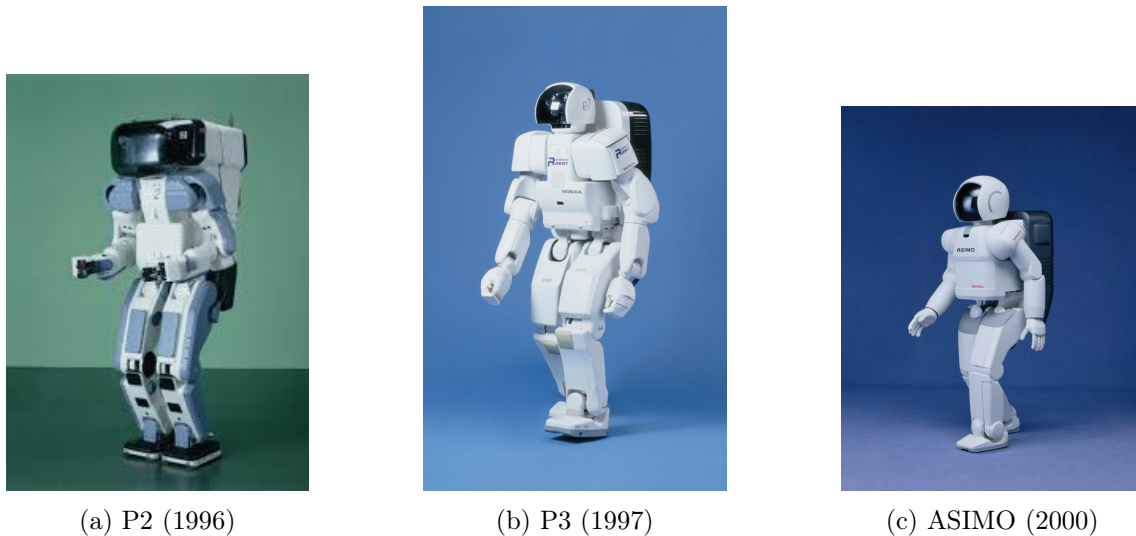


FIGURE 2.2 – Robots humanoïdes de Honda. Images issues de [12].

Les robots humanoïdes ont été l'objet de nombreuses recherches et marques d'intérêt au cours de la dernière décennie. Cela est notamment dû aux progrès technologiques effectués qui ont permis d'obtenir les déplacements et la manipulation de plus en plus proches de ceux des êtres humains. La morphologie des robots humanoïdes leur donne de nombreux avantages par rapport aux autres robots, notamment en ce qui concerne leurs interactions avec l'homme, leur capacité de travailler dans des lieux dangereux et dans tout endroit conçu pour l'être humain [71]. En effet, de nombreuses tâches quotidiennes telles que monter un escalier, ouvrir et franchir une porte ou attraper un objet en hauteur requièrent une morphologie semblable à celle de l'être humain. Un bon exemple de l'intérêt et de l'évolution des travaux est le *DARPA¹ Robotics Challenge*. Il s'agit d'une compétition qui a eu lieu de 2012 à 2015, et dont le but était de développer des robots semi-autonomes pouvant accomplir des tâches complexes dans des environnements dangereux, dégradés et construits par l'homme tels que la centrale nucléaire accidentée de Fukushima.

De nombreux travaux de recherches sont en cours, que ça soit dans le domaine privé ou au sein d'universités. Parmi les robots les plus célèbres, on citera le HRP-2 de l'université de AIST au Japon, ainsi que les robots Atlas et PETMAN développés par Boston Dynamics², destinés respectivement à tester des vêtements de protection chimique (selon les dires de ses créateurs) et à évoluer sur des terrains escarpés à l'extérieur.

1. *Defence Advanced Research Projects Agency*, <http://www.darpa.mil>

2. <http://www.bostondynamics.com>

2.2 Zero Moment Point (ZMP)

Une grande partie des robots modernes utilise des contrôleurs basés sur le respect du *Zero Moment Point* (ZMP) [68]. Pour expliquer ce concept, on utilise le polygone de sustentation. Il s'agit de l'enveloppe convexe de l'ensemble des points de contact entre les pieds et le sol (voir figure 2.3). Le ZMP existe toujours à l'intérieur du polygone de sustentation si la démarche du robot est stable et si le pied est à plat sur le sol (régulier). En cas d'équilibre statique, le ZMP se confond avec la projection du centre de masse, et peut donc être vu comme une généralisation de ce dernier dans des conditions dynamiques (à savoir lorsque les accélérations sont non nulles). Garder la projection au sol du centre de masse du robot dans le polygone de sustentation est une condition suffisante et nécessaire pour conserver l'équilibre en régime statique. L'idée des contrôleurs utilisant ce critère est de calculer des trajectoires qui satisfassent la contrainte du ZMP. Des petites perturbations peuvent être compensées en ajustant le ZMP dynamiquement pendant le mouvement. C'est ce type de contrôleur qui est utilisé pour la marche par la majorité des robots tels que ASIMO [11], HRP-2 [39] HUBO ou KHR-2 [41]. Cette méthode de contrôle possède cependant de nombreux désavantages, telles que la limitation de la vitesse, la grande énergie consommée, ou la mauvaise résistance à des perturbations extérieures.

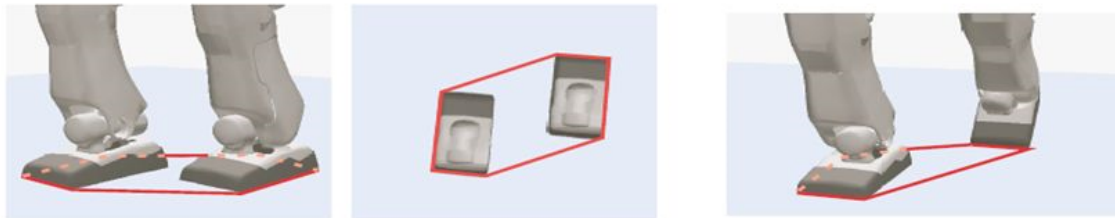


FIGURE 2.3 – Représentation du polygone de sustentation pour un contact des deux pieds (à gauche et au centre) et un contact partiel (à droite). Image issue de [38].

La grande différence entre la marche et la course est que tous les contrôleurs qui produisent une marche quasi statique ne sont pas valables pour la course. En effet, il existe deux types de marche : la marche quasi statique et la marche dynamique [38]. Contrairement à la marche quasi statique, le robot est stable lors de la marche dynamique, mais si ses articulations se figent à un moment donné, le robot chutera probablement. Garder le ZMP dans le polygone de sustentation est une condition suffisante (mais pas nécessaire) pour maintenir l'équilibre lors de la marche dynamique. Les contrôleurs dynamiques sont plus difficiles à implémenter, car les effets inertiels doivent être pris en compte.

La course possédant une phase de vol (aucun pied en contact avec le sol), il n'y a plus la présence du polygone de sustentation (à certains moments du cycle) et la démarche quasi statique n'a donc plus de raison d'être. Étant donné que ce mémoire a pour but d'obtenir une course, seules les recherches sur la locomotion dynamique seront détaillées (une très grande partie des recherches de la locomotion humaine a été basée sur la marche, notamment en se basant sur le ZMP).

De plus, reproduire une démarche humaine (démarche dynamique) n'est pas possible avec cette méthode, notamment car les pieds doivent être à plat sur le sol, la surface des pieds est trop petite par rapport à la hauteur du centre de masse, ou à cause de la différence d'un point de vue énergétique. La démarche humaine est beaucoup plus efficace. Essayer de se rapprocher le plus possible d'une démarche humaine est très compliqué. En effet, les nombreux degrés de liberté nécessaires pour avoir une mobilité et une agilité proches de celle d'un être humain rendent le contrôle de robots bipèdes ardu. Ce nombre de degrés de liberté rend souvent le problème redondant (il y a plusieurs mouvements possibles pour atteindre une configuration voulue). De

plus, la surface de contact des pieds avec le sol est limitée, ce qui rend le maintien de l'équilibre du robot difficile.

2.3 Contrôleurs dynamiques

Les idées de base des principaux contrôleurs dynamiques utilisés seront expliquées dans cette section. Nous avons regroupé les principales méthodes utilisées dans la littérature en sous-groupes pour avoir une vision globale des différentes techniques (regroupement notamment basé sur [71]). Il existe un grand nombre de contrôleurs qui se basent souvent sur une ou plusieurs des méthodes explicitées ci-dessous. Les contrôleurs purement bio-inspirés font l'objet d'une attention particulière à la section 2.4.

2.3.1 Cinématique inverse

L'utilisation de la cinématique inverse pour le mouvement des robots humanoïdes nécessite de connaître toutes les positions des articulations avant de pouvoir les imposer au robot. Étant donné le grand nombre de degrés de liberté de tels robots, il est très difficile de déterminer ces trajectoires. C'est pourquoi les données sont généralement collectées au moyen de capture de mouvements de l'être humain [51] [17]. Une fois les positions désirées aux articulations connues, les moteurs sont contrôlés pour suivre les consignes cinématiques.

L'inconvénient principal de la cinématique inverse est sa dépendance à des trajectoires articulaires qui sont déterminées dans des phases de captures de mouvements avant de les imposer sur le robot. Il est donc difficile de modifier la démarche du robot en temps réel. Ce serait un très long et fastidieux processus de composer une librairie pour encoder tous les mouvements dynamiques que le robot doit réaliser. Ce type de contrôleur dont les mouvements sont prédéfinis n'est également pas du tout robuste à des perturbations extérieures. De plus, un robot humanoïde possédant de nombreux degrés de liberté, il existe des singularités (surtout pour le genou).

2.3.2 Dynamique inverse

La démarche de la dynamique inverse a pour but de déterminer les couples de références à envoyer à chaque articulation afin d'obtenir la démarche voulue en tenant compte de la dynamique du système. Trouver la solution à ce problème n'est pas aisé pour des robots humanoïdes (base flottante), notamment à cause du sous-actionnement, de la dynamique des contraintes de l'environnement qui évolue, et des éventuelles boucles cinématiques fermées [44].

Plusieurs approches sont possibles pour résoudre ce problème de dynamique inverse. Mistry et al. [44] utilisent une décomposition orthogonale pour projeter la dynamique du robot dans un espace dimensionnel plus petit afin de trouver une solution analytique indépendante des forces de contact avec le sol. D'autres approches modifient les trajectoires désirées pour maintenir l'équilibre du robot lors de sa locomotion. Il s'agit de l'approche de Yamane et Nakamura [76] qui utilisent un filtre dynamique modifiant les trajectoires de références qui sont impossibles à atteindre physiquement (accélérations trop élevées ...).

Étant donné que des positions de références sont nécessaires afin d'effectuer la dynamique inverse, tous les inconvénients liés à la cinématique inverse sont encore d'application. Du plus, les calculs informatiques pour résoudre le problème sont très lourds compte tenu de sa complexité.

2.3.3 Optimisations de trajectoires

Plutôt que de concevoir les trajectoires articulaires de référence du corps entier manuellement ou via des captures de mouvements, l'optimisation de trajectoires permet de générer semi automatiquement un mouvement réalisable physiquement par le robot. De tels mouvements résultants d'optimisations ont été introduits par Witkin et Kass [72]. L'idée est de configurer des trajectoires (à l'aide de B-Splines ou des points de repère). Une fonction d'objectif évalue ensuite cette trajectoire pour pénaliser les violations des lois de physique et favoriser le mouvement voulu, comme la course dans [24].

La technique d'optimisation est largement utilisée dans d'autres méthodes (autres que la dynamique inverse), comme dans [69]. Le désavantage d'utiliser une technique d'optimisation est qu'elle s'effectue *offline*, car il s'agit d'un long processus demandant beaucoup de ressources. Il faut donc utiliser beaucoup d'optimisations pour obtenir différents mouvements réalisables en temps réel.

De plus, calculer les trajectoires désirées pour les articulations du corps entier est très lourd. C'est pourquoi il est plus judicieux de se concentrer sur les caractéristiques importantes lors de la locomotion plutôt que sur la position exacte de chaque articulation du corps.

2.3.4 Contrôle *Task-Space*

Le contrôle *Task-Space* permet de réduire considérablement le nombre de caractéristiques à contrôler et donc de faciliter celui-ci. Au lieu de décrire la trajectoire de toutes les articulations du robot, cette technique permet de caractériser le mouvement avec quelques paramètres, comme la position du pied et du centre de masse du robot pour la locomotion [19]. Cela permet de simplifier grandement la conception du mouvement. De plus, cela permet de contrôler en temps réel le robot. Cette méthode est souvent combinée avec une autre, comme c'est le cas par exemple avec la dynamique inverse dans [27].

Outre le centre de masse, un contrôle basé sur la quantité de mouvement centroïdal (*centroidal momentum*) est aussi utilisé. Il est défini comme la somme des quantités de mouvement, après leurs projections sur le centre de masse du robot. Il s'agit donc d'une fonction linéaire des vitesses généralisées du robot [48].

2.3.5 Méthodes heuristiques

Au lieu de contrôler directement les trajectoires du mouvement voulu, les méthodes heuristiques essaient de contrôler les caractéristiques principales du mouvement au moyen de la compréhension de la dynamique du système. Un bon exemple de cette approche est le travail de Raibert sur les robots à jambes (dès les années 80) [53] qui a réussi à produire des démarches robustes de saut et de course (et même des flips) à un robot bipède (1992) [50]. Le contrôleur utilisé se décompose en trois contrôles différents : la hauteur du saut, l'orientation sagittale du torse et la vitesse de saut. Le placement du pied en phase oscillante est la principale source du maintien de l'équilibre lors du déplacement. Cet algorithme a aussi été appliqué pour la course d'un robot bipède [54]. Hodgins et al. ont appliqué ces idées de contrôle pour des êtres humains, capables de courir et d'effectuer des voltiges en simulation [73] (1995).

Plus récemment (2007), Ying et al. ont utilisé un contrôle simple de placement de pieds pour un robot humanoïde en simulation 3D afin d'obtenir des résultats de marche et de course tout en pouvant résister à des perturbations extérieures [77]. Ce contrôleur, SIMBICON, est inspiré des travaux de Raibert et Hodgins. Pour maintenir l'équilibre, des adaptations sont effectuées en

continu lors de la locomotion, et les positions et vitesses du centre de masse sont utilisées.

Un désavantage de cette méthode est qu'il est difficile d'imaginer un contrôleur heuristique, car il est souvent basé sur une bonne intuition de la physique du problème. De plus, ces méthodes requièrent généralement l'ajustement de nombreux paramètres. Plus le nombre de paramètres est élevé, plus cela sera compliqué.

2.3.6 Méthodes de contrôle basées sur un modèle simple (*Simple-Model-Based*)

Les méthodes basées sur un modèle simple (*Simple-Model-Based*) permettent l'équilibre du robot humanoïde en contrôlant des caractéristiques importantes du mouvement. La différence avec les méthodes heuristiques est que l'approche requiert moins d'ajustements de paramètres. Le modèle de pendule linéaire inversé (LIP pour *Linear-Inverted Pendulum*) est un des modèles simples le plus utilisé dans la robotique. Ce modèle permet notamment d'établir la relation entre le ZMP et l'évolution du centre de masse afin d'obtenir des résultats de marche [37]. Des résultats de course ont été obtenus (en simulation) en combinant ce modèle avec le modèle de masse-ressort [45].

Le modèle de pendule élastique inversé (SLIP pour *Spring-Loaded Inverted Pendulum*) [8] est un bon modèle de faible ordre pour décrire la dynamique du centre de masse pour la course humaine [28] ou pour une large variété d'espèces [9]. Cependant, le modèle n'inclut pas de genou, et il ne peut donc pas être utilisé directement pour simuler les mouvements de façon précise.

2.3.7 Apprentissage par renforcement

L'apprentissage par renforcement a pour but de rendre le robot capable d'apprendre des stratégies de contrôle. Il s'agit donc de contrôleurs qui s'améliorent avec le temps. Cette technique est utilisée dans différents travaux pour la marche [46] [64]. Le grand nombre de degrés de liberté lié à la locomotion ainsi que la création de la bonne fonction d'objectif rend ce type de contrôleur difficile à implémenter. C'est pourquoi la plupart des modèles qui utilisent l'apprentissage par renforcement sont pour le moment utilisés plutôt pour des modèles simplifiés que pour de réels robots humanoïdes.

2.4 Contrôleurs bio-inspirés

Malgré les bons résultats obtenus pour la locomotion avec les méthodes explicitées précédemment, des recherches essayent de rendre le mouvement plus naturel, plus proche de celui d'un être humain. Par exemple, avec le contrôleur SIMBICON l'extension de la hanche est trop faible et l'orientation du pied est constante. Le genou ne se fléchit pas assez pendant la phase oscillante, et ne s'étend pas assez lors du contact du talon avec le sol (*heel-strike*). Certains contrôleurs obtiennent des résultats plus proches de ceux d'un être humain [14], mais les différences au niveau des hanches et des genoux persistent. De plus, tous ces modèles se basent sur des paramètres qui sont ajustés manuellement ou via des optimisations, et qui ne modélisent pas le changement de stratégie de contrôle pour de nouvelles contraintes (par exemple un handicap, l'âge de la personne ...). D'autres modèles bio-inspirés sont utilisés pour essayer d'obtenir un mouvement plus proche de ceux d'être humain, adapté à la morphologie du robot, et faire face à des perturbations extérieures.

2.4.1 Modèle musculaire

La démarche bio-inspirée utilisée permet de simuler des muscles virtuels pour le robot. Les articulations sont contrôlées par un couple de référence généré par le modèle musculaire. Celui-ci est le modèle de muscle-tendon de Hill (MTU, pour *Muscle-Tendon Unit*) [35]. Les différentes propriétés des muscles permettent d'obtenir des couples articulaires plus proches de ceux des être humains.

Le modèle musculaire est utilisé pour générer un couple aux articulations du robot. Des modèles musculo-squelettiques complets font également l'objet de recherches en simulations. De tels modèles ont déjà été utilisés pour l'animation faciale [58], les mains humaines [62], ou le haut du corps [42]. Ce type de modèle est très complexe et est beaucoup plus difficile à mettre en œuvre que les modèles basés sur l'actionnement d'articulations.

L'approche bio-inspirée se base également sur les dépenses énergétiques du métabolisme de ces muscles virtuels. Anderson et Pandy [3] ont montré qu'une démarche humaine peut être trouvée en minimisant l'énergie métabolique consommée par distance parcourue.

2.4.2 Modèles de réflexes

Un modèle musculaire agissant au niveau des articulations sagittales des hanches, genoux et chevilles a été implémenté par Geyer et Herr [29]. Il démontre qu'une marche très proche de celle de l'homme peut être générée au moyen de lois de contrôle simples basées sur des réflexes musculaires. Les lois de contrôle envoient des stimulations aux muscles qui génèrent des couples aux articulations. Ce modèle produit une démarche robuste et efficace énergétiquement, et similaire à celle d'un être humain d'un point de vue des positions et couples articulaires. Le modèle a ensuite été étendu pour la démarche 3D [60].

Le modèle initial a été adapté par Wang et al. [70] pour un déplacement en 3D et pour permettre la course. Le modèle de Geyer et Herr a également été utilisé sur le robot COMAN (utilisé dans le mémoire, voir section 3.1) pour obtenir des résultats de marche [5] [66].

Un désavantage de ce modèle est qu'il requiert des optimisations pour trouver les valeurs des nombreux paramètres libres générant la démarche souhaitée. Pour changer significativement la démarche (vitesse, longueur de pas ...), il faut recourir à de nouvelles optimisations pour avoir de nouvelles valeurs pour ces paramètres. Générer les paramètres pour toutes les démarches voulues n'est pas très naturel (l'homme ne stocke pas de table de correspondances pour chaque démarche), et changer de démarche pendant la locomotion n'est pas aisé.

Song et Geyer [61] ont développé une stratégie pour pouvoir moduler la vitesse d'avance lors de la démarche en modulant certains paramètres. Des résultats de course ont été trouvés dans cette contribution. Cependant cette approche ne permet pas de large variation de vitesse.

2.4.3 *Central Pattern Generator* (CPG)

Chez les êtres vertébrés, la locomotion est en partie contrôlée par le *Central Pattern Generator* (CPG) qui se situe dans la moelle épinière. Il s'agit d'un réseau de neurones capable de produire des oscillations en boucle ouverte (sans retours de capteurs par exemple). Taga et al. [63] ont introduit le CPG chez l'être humain en simulant un modèle musculaire avec des oscillateurs non linéaires distribués et il a été observé que le robot peut naturellement générer un mouvement de marche ou de course de manière robuste par rapport aux perturbations. Hase et al. [33] ont obtenu des résultats de marche et de course 3D stable en utilisant un système CPG hiérarchique.

Un avantage de cette technique est qu'une entrée simple permet de moduler les oscillations produites. Cela donne la possibilité de générer plusieurs démarches différentes, par exemple un changement de vitesse, en modulant uniquement certains paramètres.

Van der Noot utilise le modèle de CPG sur le COMAN en simulation [67]. Le contrôleur utilisé est un mélange de réflexes et de CPG pour contrôler les muscles des jambes. Le CPG est conçu comme un réseau d'oscillateurs de Matsuoka de six neurones. Ce contrôleur permet d'obtenir des résultats de marche pour une large gamme de vitesse (0.4m/s - 0.9m/s) à partir d'une seule optimisation. Cette approche permet également de prédire le moment du prochain contact du pied avec le sol, et de moduler la fréquence de pas.

2.5 Vue d'ensemble

Ce chapitre permet de se rendre compte de l'évolution intervenue dans le domaine de la robotique et plus particulièrement du robot humanoïde. D'une part, l'évolution des connaissances du corps humain ainsi que des techniques de fabrication permettent de construire des robots de plus en plus proches du physique et des capacités de l'être humain. D'autre part, le contrôle de ceux-ci est complexe de par le grand nombre de degrés de liberté présents sur de tels robots. La locomotion du robot est une tâche primordiale dans l'utilisation de ceux-ci, et de grands progrès ont également été effectués. Les recherches concernant la course sont encore peu nombreuses par rapport à la marche, mais ont l'avantage d'utiliser des contrôleurs dynamiques, ce qui produit une démarche plus proche de celle de l'être humain.

Comme cela est montré dans ce chapitre, il existe une multitude de principes de contrôles pour obtenir une démarche stable et robuste aux perturbations. La majorité des contrôleurs sont basés sur une ou plusieurs méthodes présentées dans ce chapitre, qui sont adaptées selon les caractéristiques du robot utilisé. Malgré la difficulté de la modélisation des principes physiques et neurologiques de la démarche humaine, les approches bio-inspirées se développent de plus en plus. Celles-ci permettront sans doute à terme d'obtenir de meilleurs résultats afin de reproduire l'efficacité de notre démarche. De nombreux projets de recherche sont effectués sur des environnements de simulations, il faut maintenant concrétiser les résultats obtenus sur des modèles physiques.

Chapitre 3

Robot COMAN & Outils

L'objectif de ce mémoire est de concevoir un contrôleur bio-inspiré afin de réaliser une course proche de celle d'un être humain sur un robot humanoïde, et ce au travers d'un simulateur. Le robot et les différents outils nécessaires pour tester le contrôleur sont décrits dans ce chapitre.

Tout d'abord, le robot utilisé est présenté. Étant donné que l'implémentation du contrôleur s'effectue au moyen d'un simulateur, celui-ci est ensuite introduit. Cet outil est indispensable pour tester le contrôleur, tant pour des raisons de sécurité (pour l'être humain et le robot) que dans un souci de gain de temps. En effet, de nombreuses optimisations doivent être lancées afin de déterminer les paramètres optimaux du contrôleur, et il n'est pas envisageable de les faire sur le robot réel. Enfin, l'algorithme et les outils utilisés pour ces phases d'optimisations sont présentés.

3.1 Robot COMAN

3.1.1 Présentation du robot

Le robot utilisé dans le cadre de ce mémoire est le COMAN (*COmpliant HumANoid Platform*). Il a été développé par l'Italian Institute of Technology (IIT) dans le cadre du projet Européen AMARSI (<https://www.amarsi-project.eu>). Il est représenté à droite de la figure 3.1.

Spécifications techniques

Ce paragraphe détaille les spécifications techniques du robot (issues de [5], [26] et [52]). Le COMAN a une morphologie similaire à celle d'un enfant de 5 ans sans tête et sans main. Il mesure 95 cm (des pieds jusqu'au cou) et pèse 31 kg. Il a 23 degrés de liberté (DDL), 14 d'entre eux sont *compliant* (*compliant joint*), et 9 sont rigides (voir figure 3.1). Ses composants mécaniques sont fabriqués en alliage de titane, en acier inoxydable et en alliage d'aluminium ce qui lui permet une bonne robustesse. Ses articulations sont actionnées par des moteurs DC, des engrenages *Harmonic Drive gears* et des éléments séries élastiques. Les articulations des bras, des jambes et du torse peuvent générer des pics de couple de 55 Nm. Néanmoins, la limite de couple qui peut être envoyé au contrôleur bas-niveau est de 30 Nm, et ce dû à des limitations logicielles.

Des capteurs de couple sont présents dans chaque articulation. De plus, des capteurs de force et de couple à 6 degrés de liberté sont inclus dans les chevilles afin de mesurer les forces de réaction du sol. Le COMAN est aussi équipé d'une centrale inertielle (IMU pour *Inertial Measurement Unit*) attachée à sa taille.

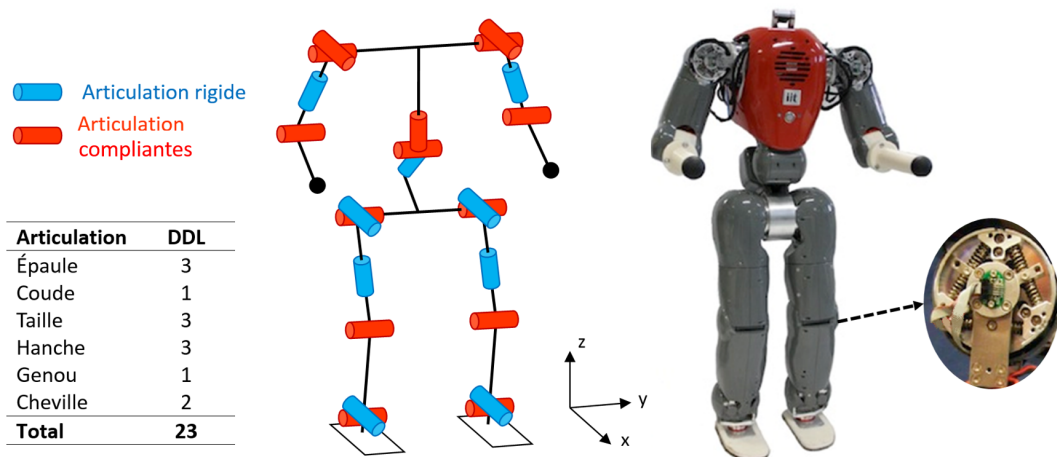


FIGURE 3.1 – Représentation des 23 degrés de liberté du COMAN et visualisation d’un actionneur séries élastiques du genou. Image adaptée de [26].

Enfin, le robot est totalement autonome. Le torse contient un Pentium Dual-Core PC104, une batterie embarquée et un système de gestion des batteries qui permet jusqu’à 2h30 d’autonomie en fonctionnement.

Actionneurs séries élastiques

Les articulations compliantes sont implémentées en utilisant des actionneurs séries élastiques. Il s’agit d’un actionneur où un élément flexible (typiquement un ressort) est intercalé entre les engrenages et la charge à entraîner. L’actionneur du genou du COMAN est montré à droite de la figure 3.1, où les éléments flexibles sont bien visibles. Ce type d’actionneur permet de mieux résister aux chocs extérieurs, d’être plus précis et stable en contrôle de force, et d’avoir des interactions avec l’environnement plus sûres. De plus, de tels actionneurs ont de nombreux avantages pour le déplacement comme expliqué dans [5]. Tout d’abord cela permet d’avoir une démarche plus humaine. En effet, pour se déplacer, un être humain (ou animal) utilise des muscles connectés au squelette via des tendons qui ont tous les deux des propriétés élastiques (toutefois, les tendons humains ne sont pas positionnés comme sur le COMAN étant donné qu’ils sont ici inclus aux moteurs). Un autre avantage est que les ressorts peuvent stocker et libérer de l’énergie, ce qui permet une meilleure efficacité énergétique. Par exemple, de l’énergie élastique peut être accumulée lorsque le pied rentre en contact avec le sol, et relâchée lorsqu’il quitte le sol. Cela permet notamment de faciliter la course.

Entrées et sorties disponibles sur le COMAN

Le simulateur n’est qu’un outil afin de simuler et implémenter notre contrôleur avant de le tester sur le robot réel (même si ce n’est pas réalisable dans le cadre de ce mémoire). Il est donc primordial que les parties contrôleur et simulateur soient découplées afin de pouvoir utiliser le contrôleur sur le robot réel. Pour ce faire, seules les entrées/sorties disponibles sur le COMAN peuvent être utilisées.

Les données suivantes issues des capteurs sont les entrées disponibles du contrôleur :

- les forces de réaction du sol et couples mesurés en dessous du pied ;
- les positions, vitesse et couple mesurés à chaque articulation ;
- la matrice d’orientation absolue fournie par l’IMU.

De ces entrées, le contrôleur envoie des positions ou couples de références au contrôleur bas-niveau des articulations.

3.1.2 Modèles de pieds

Les pieds jouent un rôle très important dans la locomotion humaine. Une grande différence entre les pieds du COMAN et les pieds humains est la souplesse de ces derniers. En effet, le COMAN est initialement équipé de pieds rigides. Le dimensionnement du modèle des pieds rigides présent dans le simulateur et utilisé dans [67] et [66] est représenté à la figure 3.3a.

Afin de se rapprocher le plus possible d'un modèle humain, un nouveau modèle de pieds avec orteils a été créé en simulation dans le cadre de ce mémoire. Il s'agit du modèle rigide auquel est ajoutée à son extrémité une deuxième partie rigide liée à la première avec un élément élastique passif comme représenté schématiquement à la figure 3.3b. Un tel modèle a déjà été investigué sur le COMAN dans [5]. Ce modèle a pour but de se rapprocher du pied humain, ce qui pourrait permettre une locomotion plus rapide, plus efficace d'un point de vue énergétique, et une démarche plus proche de celle d'un humain. De plus, l'ajout d'un élément passif peut s'avérer utile pour stocker de l'énergie lorsque le pied entre en contact avec le sol, pour ensuite la libérer lorsqu'il quitte le sol (par exemple). Cela peut évidemment aider le robot lors de la propulsion pour la course.

Dimensionnement du pied avec orteils

Comme expliqué, le degré de liberté supplémentaire du pied est ajouté avec une articulation passive du type ressort-amortisseur. Le nouveau pied peut être caractérisé par :

- La longueur totale du pied ;
- La proportion des orteils (nouvelle partie rigide fixée à l'extrémité du pied) par rapport à la longueur totale du pied ;
- Le coefficient de raideur du ressort de torsion ;
- Le coefficient d'amortissement du ressort de torsion.

Concernant le dimensionnement de la taille du pied, nous nous basons sur celui utilisé dans [5]. L'idée était d'avoir la partie rigide du pied de la même taille que celui utilisé initialement, et d'ajouter les orteils afin de pouvoir au mieux comparer les résultats. Pour trouver la taille des orteils, une distribution de la pression plantaire d'un pied humain est utilisée dans le but d'être le plus proche d'un pied humain (voir figure 3.2). Ce modèle de pied est appelé dans la suite de ce mémoire *pied long avec orteils* et est représenté schématiquement à la figure 3.3b.

Un troisième modèle de pied est utilisé dans le cadre de ce mémoire : *pied court avec orteils*. Il s'agit également d'un pied avec orteils, mais de taille égale au pied rigide initialement utilisé (tout en gardant les proportions utilisées précédemment). Ce modèle est représenté schématiquement à la figure 3.3c. Concernant les motivations de raccourcir le pied, nous avons remarqué lors de diverses simulations que le robot avait tendance à activer au maximum et en permanence le muscle de la cheville responsable de dorsiflexion du pied (muscle *tibialis anterior*, voir section 4.3.1). L'objectif de cette activation permanente est d'éviter que le bout du pied ne touche le sol lors de la période oscillante. Réduire la longueur totale du pied permet d'éviter plus facilement ce phénomène qui conduit à une chute du robot, tout en gardant les propriétés des pieds flexibles. Une comparaison de ces pieds est effectuée dans l'analyse des résultats à la section 5.5.

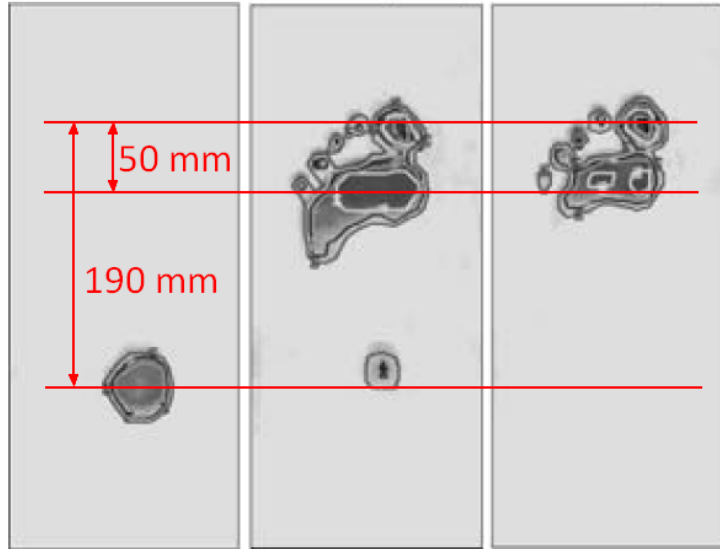


FIGURE 3.2 – Distribution de la pression plantaire d’un pied humain utilisée pour trouver la proportion de longueur orteils-pied. Image issue de [5].

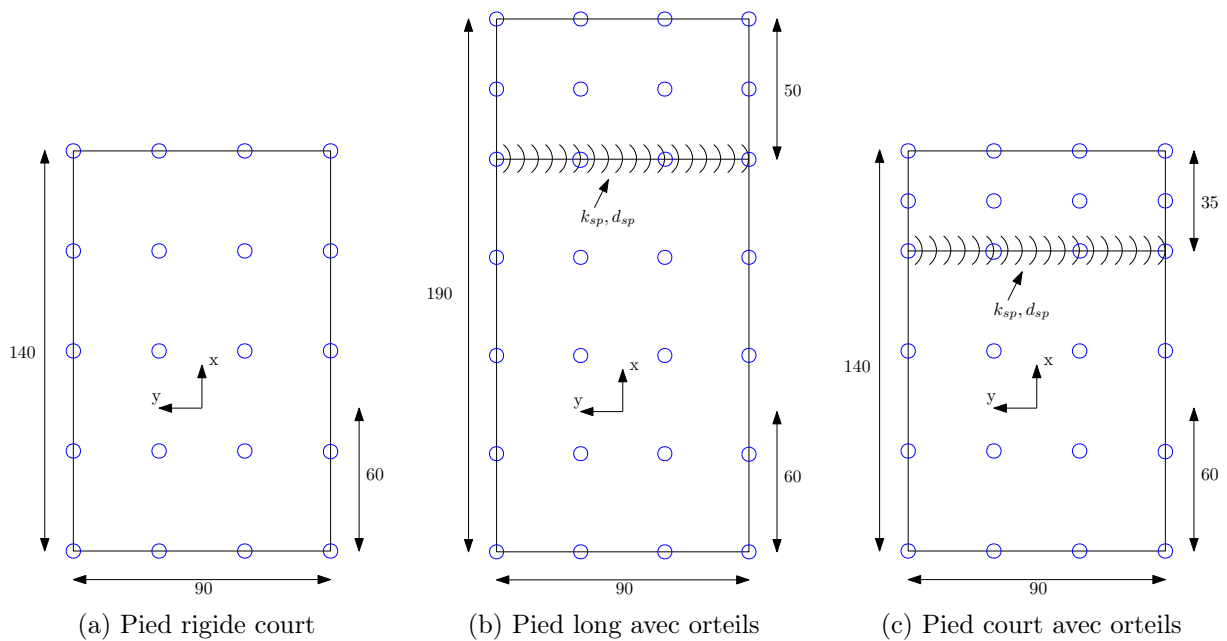


FIGURE 3.3 – Dimensionnement (en mm) des trois différents modèles de pieds utilisés. Les cercles bleus représentent le maillage du modèle de contact avec le sol. k_{sp} et d_{sp} sont respectivement les coefficients de raideur et d’amortissement du ressort de torsion. L’origine du repère est le point d’attache du pied avec la cheville.

Concernant les paramètres de ressort de torsion, nous avons tout d’abord fixé le coefficient d’amortissement à 1 Nm s/rad, essentiellement pour stabiliser l’intégration numérique des équations du mouvement (valeur définie après essais-erreurs). En ce qui concerne la valeur du coefficient de raideur, il est le résultat d’optimisations : nous laissons l’optimisateur libre de choisir la meilleure valeur afin d’obtenir la démarche souhaitée (via une fonction d’objectif, voir la section 3.3 pour plus de détails concernant le processus). Les résultats ont convergé vers une valeur de 30 Nm/rad, ce qui démontre que l’ajout de cet élément passif est a priori utilisé et bénéfique dans notre cas (une valeur convergeant vers 0 tend à ne pas utiliser les orteils, et une valeur

convergeant vers $+\infty$ tend à utiliser un pied totalement rigide). C'est cette valeur que nous utilisons par la suite dans toutes les simulations¹ (tant pour le modèle de pieds avec orteils longs que courts).

Implémentation

Le couple appliqué à l'articulation entre les deux plaques rigides est fourni par l'équation classique ressort-amortisseur :

$$\tau = -k_{sp}\theta - d_{sp}\dot{\theta} \quad (3.1)$$

où k_{sp} et d_{sp} sont respectivement les coefficients de raideur et d'amortissement du ressort de torsion, et θ et $\dot{\theta}$ sont les position et vitesse angulaires de l'articulation des orteils (avec une position et vitesse initiales nulles).

Notons que nous avons dû faire face à des erreurs numériques à la suite de l'ajout des modèles de pieds avec orteils : le pied était fortement "repoussé" par le sol lors du contact. Pour résoudre ce problème, la masse du pied a été augmentée (en allongeant son épaisseur) et le pas de temps de l'intégrateur a été divisé par quatre (passant de $500 \mu s$ à $125 \mu s$). La force de contact pied/sol étant proportionnelle à la pénétration du pied dans le sol (voir section 3.2.2), diminuer le pas de temps permet de diminuer la force de réaction, car le pied a moins le temps de s'enfoncer dans le sol (et il recalcule cette force 4 fois plus souvent), et augmenter la masse permet d'avoir une accélération plus faible (pour une force de réaction identique).

3.2 Outils de simulation

Le simulateur utilisé dans le cadre de ce mémoire a pour but de simuler le robot COMAN et son environnement afin de pouvoir tester le contrôleur de course tout en étant le plus proche possible des conditions existantes sur le vrai robot. Pour ce faire, un modèle du robot a été développé par N. Van der Noot & A. Barrea [5], et est notamment utilisé dans [67]. Cet outil inclut la cinématique et la dynamique du robot, la dynamique des moteurs et un modèle de contact avec le sol pour simuler les forces de réaction du sol qui agissent sur les pieds du robot. Le simulateur est un outil très précieux (voire indispensable) pour pouvoir tester et valider le contrôleur avant d'éventuellement le tester sur le robot réel (ce qui n'est pas le but de ce mémoire). En effet, le contrôleur utilise plus de 70 paramètres dont les valeurs doivent être déterminées (grâce à la phase d'optimisation) afin d'obtenir la démarche souhaitée. Automatiser les tests et l'optimisation (faire fonctionner le robot des centaines/milliers de fois avec des paramètres différents pour converger vers les meilleurs) n'est pas envisageable sur le robot réel, et cela peut même être dangereux pour le robot et pour l'être humain.

3.2.1 Modèle Robotran

Le modèle du COMAN est développé avec l'environnement de simulation multicorps Robotran [55] développé au Centre de Recherche en Mécatronique (CEREM) de l'Université catholique de Louvain (UCL). Il s'agit d'un programme symbolique pour modéliser et analyser des systèmes multicorps. Un tel système est composé de corps rigides connectés entre eux par des articulations qui peuvent être sujettes à des déplacements transversaux et/ou des rotations. Ces corps peuvent être soumis à des forces et couples internes ou externes. Pour plus d'informations concernant cet outil, le lecteur peut consulter le site officiel de Robotran (<http://www.robotran.be>) et [55].

1. Il s'agit d'un choix fait suite à des optimisations initiales; on pourrait imaginer d'optimiser ce paramètre pour chaque simulation

Une représentation visuelle du système multi corps du robot est fournie à la figure 3.4. Le robot est attaché au repère inertiel via 6 articulations (3 articulations prismatiques, et 3 articulations rotoïdes), et les corps sont connectés entre eux via une ou plusieurs articulations rotoïdes.

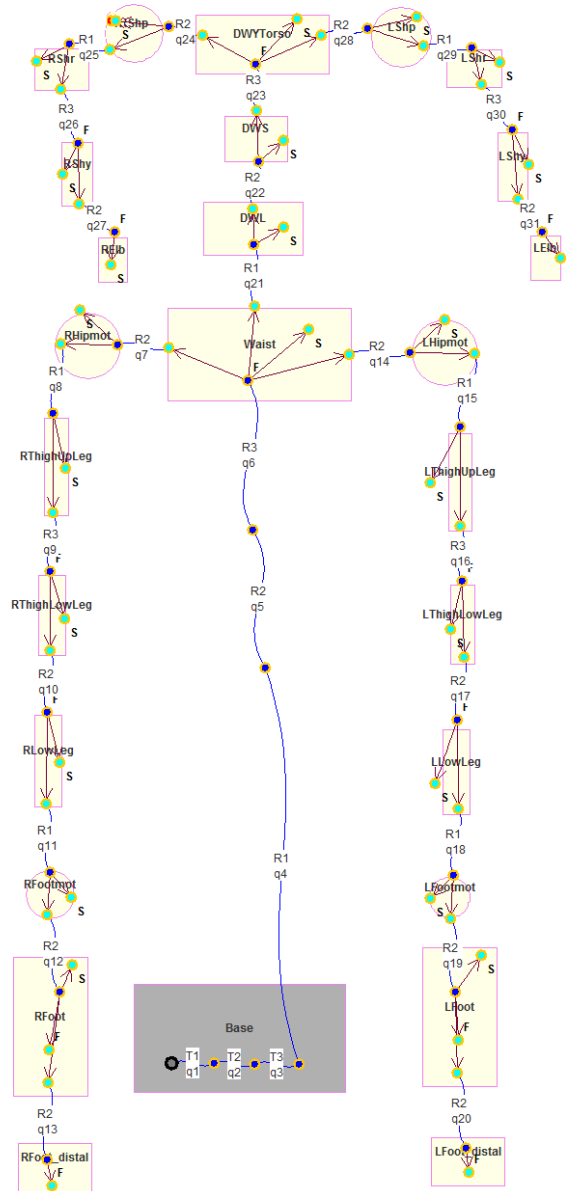


FIGURE 3.4 – Représentation du système multicorps du robot COMAN via l’interface graphique de Robotran, MBSysPad.

3.2.2 Partie simulateur

En plus de la modélisation cinématique et dynamique du robot, il faut modéliser son environnement et les interactions qu’il peut avoir avec celui-ci. Dans le cas de la locomotion, la modélisation du contact des pieds avec le sol est une partie très importante. C’est pourquoi le modèle utilisé est détaillé dans cette partie.

Modèle de contact avec le sol

Afin de modéliser les forces de réaction du sol agissant sur les pieds du robot, un modèle de contact avec le sol basé sur un maillage est utilisé. Celui-ci a été initialement implémenté sur le simulateur dans [5] (et a depuis évolué) et est basé sur [15]. L'avantage d'utiliser un maillage est de pouvoir modéliser des formes de pieds et/ou des sols complexes.

Le maillage utilisé pour les différents pieds est visible à la figure 3.3 (cercles bleus). À chaque point du maillage, les forces d'interaction entre le pied et le sol sont calculées (F_x , F_y et F_z).

La force de réaction normale du sol F_z (positive vers le haut) est modélisée en utilisant un modèle de ressort-amortisseur :

$$F_z = -k_z \Delta z (1 - d_z \dot{z}) \quad (3.2)$$

où k_z est le coefficient de raideur de contact normal (8150 kN/m), d_z est le coefficient d'amortissement de contact normal (3000 s/m), Δz est la pénétration du point du maillage dans le sol, et \dot{z} est sa vitesse de pénétration. Cette force de réaction est non nulle si $\Delta z > 0$ (pénétration du point dans le sol) et $d_z \dot{z} < 1$.

F_x et F_y sont les composantes tangentielles de la force de réaction du sol. Ces forces sont calculées à chaque point du maillage différemment selon qu'il s'agit de frottement sec statique ou dynamique. Un point est en frottement sec statique si la norme de sa vitesse tangentielle est inférieure à 1 cm/s : $\sqrt{\dot{x}^2 + \dot{y}^2} < 1 \text{ cm/s}$. Ce point passe en frottement sec dynamique si la norme de la force tangentielle de frottement est supérieure à la force de réaction normale multipliée par le coefficient statique de frottement μ_{st} : $\sqrt{F_{x,st}^2 + F_{y,st}^2} > \mu_{st} F_z$, avec $\mu_{st} = 0.9$.

En cas de frottement sec statique, les forces tangentielles se modélisent comme suit :

$$F_{x,st} = -k_{xy} \Delta x (1 + \text{sgn}(\Delta x) d_{xy} \dot{x}) \quad (3.3)$$

$$F_{y,st} = -k_{xy} \Delta y (1 + \text{sgn}(\Delta y) d_{xy} \dot{y}) \quad (3.4)$$

où k_{xy} est le coefficient de raideur de contact tangentiel (8200 N/m), d_{xy} est le coefficient d'amortissement de contact tangentiel (3000 s/m), Δx et Δy sont les différences entre la position initiale et actuelle du point de contact, et \dot{x} et \dot{y} sont leur vitesse tangentielle. Ces forces tangentielles sont non nulles si $\text{sgn}(\Delta x) d_{xy} \dot{x} > -1$ et $\text{sgn}(\Delta y) d_{xy} \dot{y} > -1$.

En cas de frottement sec dynamique, les forces tangentielles se modélisent comme suit :

$$F_{x,dyn} = -\frac{\mu_{dyn} F_z \dot{x}}{\sqrt{\dot{x}^2 + \dot{y}^2}} \quad (3.5)$$

$$F_{y,dyn} = -\frac{\mu_{dyn} F_z \dot{y}}{\sqrt{\dot{x}^2 + \dot{y}^2}} \quad (3.6)$$

où μ_{dyn} est le coefficient de glissement (0.8), et \dot{x} et \dot{y} sont les vitesses tangentielles du point du maillage.

Pas de temps et bruit

Le pas de temps du simulateur est fixé à 125 μs . La méthode utilisée est celle de Runge-Kutta d'ordre 4, ce qui signifie que le modèle est évalué 4 fois pour chaque pas de temps de 125 μs . Cette faible valeur est nécessaire pour avoir un contact pied-sol le plus correct possible (tout en

maintenant la valeur suffisamment élevée pour limiter le temps de simulation).

Afin de se rapprocher le plus de la réalité (utilisation du vrai robot), un bruit aléatoire est ajouté à l'erreur de couple (utilisée dans le contrôleur bas-niveau des moteurs) d'amplitude maximale de 0.4 Nm et de période de 10 ms. Cela signifie que le bruit est actualisé toutes les 10 ms, et sa valeur à chaque pas de temps est une interpolation linéaire entre la valeur précédente et la nouvelle.

3.2.3 Partie contrôleur

Le contrôleur en tant que tel est expliqué dans le chapitre 4. Nous expliquons dans ce chapitre très brièvement les outils utiles pour implémenter le contrôleur.

Calculs cinématiques

Comme expliqué à la section 3.1.1, il faut garder à l'esprit que les entrées du contrôleur sont limitées à celles réellement présentes sur le vrai robot, à savoir : les forces et couples de réaction du sol sous les pieds ; les positions, vitesses et couples de chaque articulation ; la matrice d'orientation absolue fournie par l'IMU.

Afin d'avoir toutes les données utiles pour le contrôleur, les différentes données cinématiques suivantes sont calculées dans la partie contrôleur (à partir d'équations symboliques) :

- Les positions et vitesses du centre de masse du robot ;
- Les positions, vitesses et orientations des pieds ;
- L'orientation et la vitesse angulaire du bassin et du torse.

États de la marche

Il est également nécessaire de connaître à tout moment dans quel état se trouve chaque pied : phase oscillante (*swing phase*) ou phase d'appui (*stance phase*). Cet état est déterminé en fonction des forces de réaction du sol. Un pied est en phase oscillante s'il n'est pas en contact avec le sol :

$$F_z < F_{lim} \ \&\& \ F_{z,moy} < F_{lim} \ \&\& \ F_{z,orteils} < F_{lim} \ \&\& \ F_{z,orteils,moy} < F_{lim} \quad (3.7)$$

où F_z et $F_{z,orteils}$ sont respectivement les forces de réaction normale sous le pied et les orteils (si présents), $F_{z,moy}$ et $F_{z,orteils,moy}$ sont les forces moyennées sur les 50 dernières valeurs, et F_{lim} est le seuil de détection fixée à 50 N. Les forces instantanées et moyennées sont utilisées afin de pallier à certaines erreurs dues aux bruits des capteurs (en réalité) ou du modèle de contact du sol (en simulation). En plus des forces de réaction du sol, certaines sécurités ont été ajoutées. Pour entrer en phase oscillante, le temps passé en phase d'appui doit être supérieur à 0.2 seconde.

Pour entrer dans la phase d'appui, il faut respecter les conditions suivantes :

- Le pied n'est pas en contact avec le sol (équation (3.7) pas respectée) ;
- Le temps passé en phase oscillante doit être supérieur à 0.15 seconde ;
- L'angle sagittal de la cheville < 0.6 radian.

Cette dernière condition permet d'éviter le passage indésirable dans cette phase au cas où le pied viendrait à frotter le sol au milieu de sa course.

Pas de temps et bruit

Le contrôleur est appelé toutes les 1 ms. Il s’agit du pas de temps du contrôleur sur le vrai robot (et suffisant pour tenir compte du changement de dynamique des muscles).

Du bruit est également ajouté dans le modèle musculaire. Il est ajouté aux stimulations envoyées aux muscles (S_m) et est d’amplitude maximale de 0.05 et de période 0.1 s (voir section 4.2.2 pour plus de détails).

Le tableau 3.1 récapitule les pas de temps ainsi que les bruits maximums ajoutés pour la partie contrôleur et simulateur.

| | Δt | Bruits |
|------------|-------------|-----------------------|
| Simulateur | 125 μs | $S_m(1 \pm 0.05)$ |
| Contrôleur | 1 ms | $err_\tau(1 \pm 0.4)$ |

TABLE 3.1 – Récapitulatif des pas de temps et du bruits maximal présent pour la partie contrôleur et simulateur. S_m est la stimulation envoyée au muscle m , et err_τ est l’erreur de couple.

3.3 Optimisations

Le contrôleur de locomotion bio-inspiré présenté au chapitre 4 possède plus de 70 paramètres qui doivent être déterminés afin d’obtenir la démarche souhaitée. Trouver la valeur optimale pour chacun d’eux n’est pas une chose facile, et une phase d’optimisation est donc requise.

Chaque paramètre est borné entre certaines valeurs déterminées manuellement. Ces paramètres bornés constituent l’espace de recherche dans lequel une solution optimale doit être trouvée, en considérant certains objectifs. Ceux-ci sont évalués au moyen d’une fonction d’objectif qui permet de quantifier la qualité des paramètres testés. Le but de la phase d’optimisation est donc de trouver les valeurs des paramètres qui maximisent cette fonction d’objectif (décrite à la section 4.7.1).

Une fois les optimisations terminées, les valeurs optimales des paramètres sont extraites pour être utilisées dans le contrôleur (ou pour déterminer la valeur de la raideur des orteils dans les pré-optimisations).

3.3.1 Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

Pour déterminer les valeurs optimales des paramètres, l’algorithme de stratégie d’évolution avec adaptation de matrice de covariance CMA-ES (*Covariance Matrix Adaptation Evolution Strategy*) est utilisé. Il s’agit d’une méthode d’optimisation stochastique destinée à résoudre des problèmes d’optimisation complexes non linéaires et non convexes dans un domaine continu [32], ce qui correspond à notre cas. Étant donné qu’il s’agit d’un algorithme non déterministe (car le problème est trop complexe pour utiliser une méthode déterministe), il se peut que les solutions trouvées soient des optima locaux et non globaux.

L’objectif de l’algorithme est de maximiser une fonction d’objectif. Exprimons cela mathématiquement : soit x l’ensemble des paramètres à optimiser, et soit χ le domaine de recherche, tel que $x \in \chi$. χ est généralement choisi dans l’espace euclidien \mathbb{R}^n , où n est la dimension du domaine de recherche (soit le nombre de paramètres à optimiser). Dans le projet, on borne les paramètres dans l’intervalle unitaire pour le processus d’optimisation (via une transformation linéaire). Nous avons donc $x \in \chi = [0, 1]^n \subset \mathbb{R}^n$. Considérons maintenant la fonction d’objectif

$f : \mathcal{X} \subset \mathbb{R}^n \rightarrow \mathbb{R} : x \rightarrow f(x)$.

Pour toute recherche stochastique, les paramètres de la distribution θ sont initialisés et la taille de la population $\lambda \in \mathbb{N}$ est fixée. Pour chaque itération, les trois étapes suivantes sont effectuées :

- Distribution de l'échantillon $P(x|\theta) \rightarrow x_1, \dots, x_\lambda \in \mathbb{R}$;
- Évaluation de x_1, \dots, x_λ avec f ;
- Mise à jour des paramètres $\theta \leftarrow F_\theta(\theta, x_1, \dots, x_\lambda, f(x_1), \dots, f(x_\lambda))$.

Chaque algorithme dépend de la définition de P et de F_θ . Dans le cas d'algorithme d'évolution, le nombre de particules représente la taille de la population λ , et le nombre de générations représente le nombre d'itérations. Ces deux paramètres sont à déterminer par l'utilisateur pour avoir un bon compromis rapidité/convergence des résultats. Dans le cas présent, les nouveaux points de recherche sont distribués selon une distribution normale pour une génération $g = 0, 1, 2, \dots$:

$$x_k^{(g+1)} \sim m^{(g)} + \sigma^{(g)} \mathcal{N}(0, C^{(g)}) \quad \text{pour } k = 1, \dots, \lambda \quad (3.8)$$

où la moyenne $m \in \mathbb{R}^n$ représente la solution favorite, l'écart type $\sigma \in \mathbb{R}_+$ contrôle la taille de pas (*step-size*), et la matrice de covariance $C \in \mathbb{R}^{n \times n}$ détermine la forme de la distribution. Une distribution normale est utilisée notamment car elle est largement observée dans la nature, elle est stable avec une variance finie (la somme de distribution normale et une distribution normale) et est isotrope (ne favorise aucune direction).

Pour terminer l'itération, il faut déterminer comment calculer $m^{(g+1)}$, $C^{(g+1)}$ et $\sigma^{(g+1)}$ pour la génération suivante $g + 1$. Les équations générales pour déterminer ces valeurs sont brièvement expliquées à l'annexe B.2. Pour plus de détails, voir [32].

3.3.2 Outils d'optimisation

Lancer les optimisations requiert une grande puissance de calcul. En effet, pour avoir des résultats fiables, les optimisations sont généralement lancées avec 70 particules et nous souhaitons avoir 1000 générations (soit 70 000 évaluations du modèle). Étant donné qu'un résultat de simulation n'est pas toujours identique dû au bruit ajouté à la lecture du couple des actionneurs ainsi qu'aux stimulations envoyées aux muscles, nous évaluons chaque modèle trois fois (soit 210 000 évaluations). Le temps de simulation est fixé à 15 secondes, et le pas de temps d'intégration à 125 microsecondes. Si on suppose que l'ordinateur utilisé parvient à faire tourner la simulation en temps réel, et que le robot ne tombe pas durant le temps de simulation (ce qui n'est pas le cas, du moins pour les premières itérations), il faudrait environ 36 jours pour avoir un résultat ! Pour obtenir de bons résultats d'optimisations avec un temps de calcul raisonnable, nous utilisons des superordinateurs. L'idée est de pouvoir lancer plusieurs simulations en parallèle (idéalement le même nombre que de particules), chacune sur un processeur différent.

Hardware : CECI clusters

Pour faire tourner les optimisations, les machines provenant du Consortium des Équipements de Calcul Intensif² (CECI) sont utilisées. Il s'agit d'un consortium de centres de calculs hautes performances de 5 universités belges (Université catholique de Louvain, Université libre de Bruxelles, Université de Mons, Université de Liège et Université de Namur).

Ces machines sont accessibles aux membres de ces universités au moyen du protocole SSH. Pour plus d'informations concernant la configuration et l'utilisation de ces machines, se référer

2. <http://www.ceci-hpc.be/>

à l'annexe B.1 et au site web du CECI : <http://www.cec-hpc.be/clusters.html>. Pour avoir une idée de la puissance de ces machines, voici les caractéristiques principales de l'une d'entre elles : NIC4 (hébergée à l'ULg). Elle possède 128 noeuds de calcul, chacun des noeuds ayant 2 processeurs Intel E5-2650 8 coeurs à 2 GHz (soit 2048 processeurs) et 64 GB de RAM.

Software : MPI & CMA-ES

Afin de minimiser la durée d'optimisation, les évaluations du programme sont effectuées en parallèle. Pour ce faire, nous utilisons la norme de communication MPI (*Message Passing Interface*) pour l'interaction entre les processeurs.

Afin de bien comprendre le processus, résumons les paramètres qui agissent lors de l'optimisation :

- N , le nombre de paramètres bornés à optimiser qui proviennent du contrôleur. Le but de l'optimisation est de trouver des valeurs pour ces paramètres qui maximisent la fonction d'objectif (qui reflète l'objectif voulu, par exemple la course du robot) ;
- N_{part} , le nombre de particules utilisée dans l'algorithme d'évolution CMA-ES. Chaque particule représente un ensemble des N paramètres bornés ;
- N_{gen} , le nombre de générations utilisée dans l'algorithme d'évolution CMA-ES. À chaque génération, les valeurs des paramètres de chaque particule sont mises à jour.
- N_{proc} , le nombre de processeurs demandés à la machine pour lancer les optimisations.

Un système de maître/esclaves est utilisé afin de répartir le travail entre les différents processeurs. Ce processus n'a de sens que si minimum 3 processeurs sont utilisés.³. Le nombre d'esclaves est déterminé par le nombre de processeurs moins un (le master). Pour chaque génération, le maître envoie à chaque esclave un ensemble de paramètres (une particule), l'esclave exécute le programme et renvoie la valeur de la fonction d'objectif au maître. Le programme est exécuté trois fois (car il n'est pas déterministe à cause du bruit ajouté) et le résultat de la fonction d'objectif est la moyenne des trois résultats obtenus. Tous les esclaves exécutent donc le programme en parallèle. L'idéal est de demander autant de processeurs que de particules (plus un pour le master). Si ce n'est pas le cas, le maître attend que l'esclave le plus rapide ait fini d'exécuter son programme pour lui envoyer la particule suivante, et ainsi de suite. Une fois tous les ensembles de paramètres testés, le maître met à jour la répartition de la population (nouveaux ensembles pour les particules), sauvegarde le meilleur résultat, et recommence le processus jusqu'à avoir atteint le nombre de générations voulu. Une représentation du processus est fournie à la figure 3.5.

L'implémentation de l'algorithme CMA-ES provient du projet GitHub de son créateur N. Hansen : <https://github.com/CMA-ES/c-maes>. Notons que par facilité, l'ensemble des paramètres bornés à optimiser subissent une transformation linéaire pour être bornés entre 0 et 1 lors de la phase d'optimisation. La transformation inverse est ensuite appliquée pour connaître les valeurs réelles à utiliser dans le contrôleur.

3. Un tel système n'a pas de sens pour 1 processeur, et utiliser 2 processeurs (1 maître et 1 esclave) est identique à faire tourner toutes les simulations sur un seul processeur.

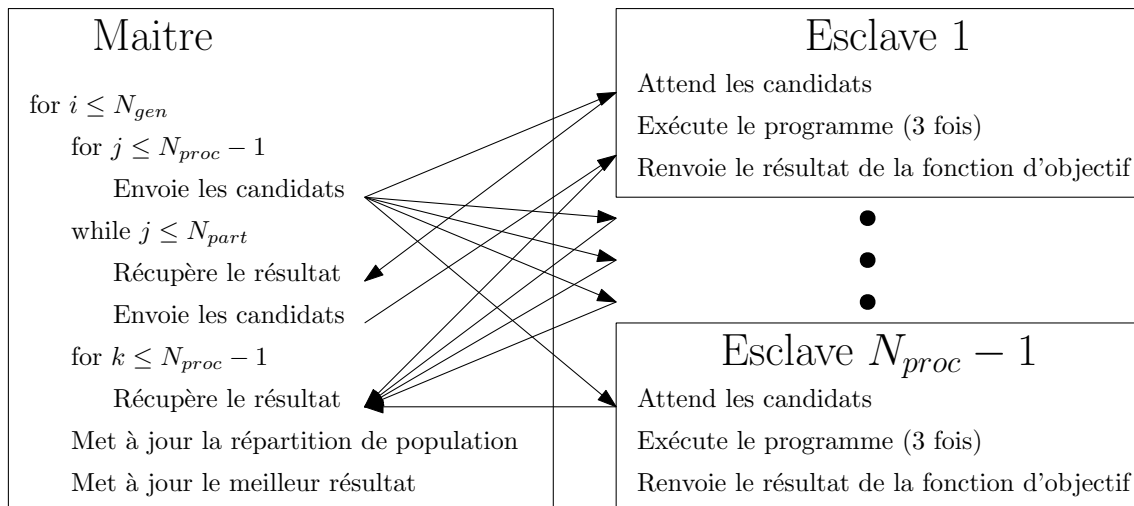


FIGURE 3.5 – Représentation du système maître/esclaves utilisé.

3.4 Conclusion

Le robot COMAN, de la taille d'un enfant de 5 ans, est le robot humanoïde (conçu par L'IIT) sur lequel le contrôleur de course implémenté dans le cadre de ce mémoire sera testé. Il a le mérite d'être conçu avec des articulations compliquées, ce qui donne la possibilité de retrouver l'élasticité présente dans les muscles chez l'être humain (même si l'emplacement et les propriétés physiques des éléments élastiques ne sont pas identiques).

Le simulateur conçu par N. Van der Noot et A. Barrea [5] (qui évolue avec les travaux de N. Van der Noot [67]) est un outil indispensable qui est utilisé dans le cadre de ce mémoire. Celui-ci est découpé en deux parties distinctes : la simulation du robot et de son environnement, et le contrôleur. Ce découplage est indispensable pour pouvoir tester le contrôleur sur le vrai robot. Le pas de temps de ces deux parties est différent : $125 \mu s$ pour la partie simulation, et $1 ms$ pour la partie contrôleur. De plus, du bruit est ajouté dans la partie simulation à la lecture de l'erreur de couple, ainsi que dans les stimulations envoyées aux muscles dans la partie contrôleur.

La plupart des outils de la partie simulation n'ont pas subi de changements majeurs, excepté pour les nouveaux modèles de pieds avec orteils et pour la configuration des optimisations avec le protocole MPI et l'algorithme CMA-ES qui sont des contributions de ce mémoire.

Chapitre 4

Locomotion bio-inspirée

Le but de ce mémoire est de concevoir un contrôleur bio-inspiré qui permet au robot humanoïde COMAN de courir (en simulation). Cette course a pour objectif de se rapprocher le plus possible de celle d'un être humain de morphologie similaire au robot utilisé (voir chapitre 5 pour l'analyse des résultats).

Tout d'abord, nous présentons une vue générale du contrôleur. Nous y introduisons notamment la définition des sens du mouvement du COMAN, les contraintes pour utiliser le contrôleur uniquement en deux dimensions (2D), ainsi que les interactions entre la partie contrôleur et la partie simulation du simulateur.

Les muscles virtuels créés pour le COMAN sont ensuite expliqués. Il en résulte que les couples de références des différentes articulations sont déterminés en fonction du niveau de stimulation envoyé à chacun des muscles virtuels du COMAN.

Le contrôleur bio-inspiré 2D utilisé est ensuite présenté. Celui-ci se décompose en deux parties distinctes : le contrôleur des degrés de libertés des jambes et celui des degrés de libertés du haut du corps (torse et bras). L'état initial du robot pour pouvoir réaliser une course est aussi décrit dans ce chapitre.

Enfin, la fonction d'objectif utilisée pour obtenir des résultats de course au travers d'optimisation est détaillée en fin de chapitre.

4.1 Vue générale du contrôleur

Pour rappel, on parle d'articulation sagittale, frontale ou transversale, en référence aux différents plans du corps humain qui sont représentés à la figure 1.1. Le repère inertiel ainsi que les références des positions des articulations sagittales sont représentés à la figure 4.1. Pour les articulations sagittales des jambes, une position positive engendre donc une extension de la hanche, une flexion du genou, et une flexion plantaire de la cheville.

Le contrôleur de locomotion implémenté agit uniquement sur les articulations sagittales. Pour tester ce contrôleur, la base flottante est contrainte de telle façon que le robot ne puisse pas tomber latéralement dans le simulateur (les articulations rotoïdes non sagittales de cette base ont une position imposée à zéro). De plus, toutes les articulations non sagittales des jambes ont une position de référence nulle. La généralisation du contrôleur pour une démarche non contrainte (3D) n'est pas couverte dans le cadre de ce mémoire.

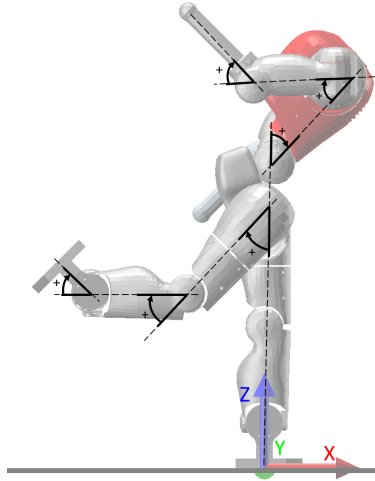


FIGURE 4.1 – Définition du repère inertiel et des références des positions des articulations sagittales. Les articulations sagittales de la jambe gauche et du bras gauche sont toutes à une position nulle, et celles de la partie droite du corps et du torse à une position $+\pi/4$.

Le contrôleur a pour but de fournir des couples/positions de références aux différents moteurs du robot à partir des entrées disponibles (voir section 3.1.1). Il est décomposé en deux parties : un contrôleur pour les jambes, et un contrôleur pour le haut du corps (torse et bras). Les articulations sagittales des jambes sont contrôlées avec des couples de références, et les articulations non sagittales des jambes sont contrôlées avec des positions de références. Les articulations du haut du corps (torse et bras) peuvent être contrôlées avec des positions ou des couples de références. Le schéma à la figure 4.2 représente les différentes parties du contrôleur et leurs interactions avec les outils de simulations.

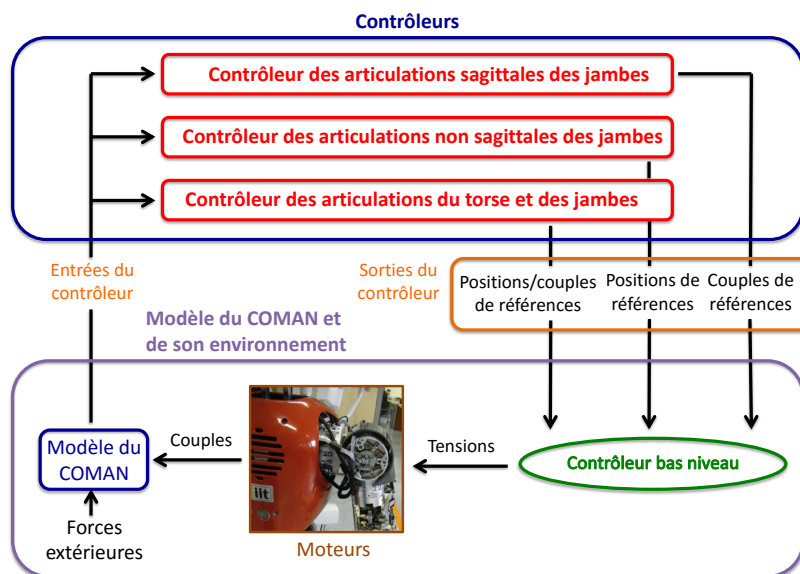


FIGURE 4.2 – Interactions entre la partie contrôleur et la partie simulation. Image adaptée de [5]

Le modèle musculaire qui permet de générer un couple de référence d'une articulation à partir des stimulations envoyées aux muscles connectés à celle-ci est expliqué à la section 4.2. Les différents muscles virtuels du COMAN sont présentés à la section 4.3. Le contrôle du haut du corps est ensuite expliqué à la section 4.4, et celui du bas du corps à la section 4.5. Le tableau 4.1 résume les contrôles (couple ou position) des articulations du robot.

| Corps | Articulations | Contrôle |
|--------|----------------|----------------------------------|
| Jambes | Sagittales | Couples via le modèle musculaire |
| | Non-sagittales | Positions nulles |
| Bras | Sagittales | Couples via le modèle musculaire |
| | Non-sagittales | Positions |
| Torse | Sagittale | Position nulle |
| | Non-sagittales | Positions nulles |

TABLE 4.1 – Récapitulatif du contrôle des articulations des différentes parties du robot.

4.2 Modèle musculaire

Dans le but de développer une démarche semblable à celle de l'être humain, des groupes de muscles virtuels sont ajoutés au robot. Chaque muscle agit sur une ou deux articulations (dans le cas de muscles dits "bi articulaires") du robot. Le but du contrôleur est d'envoyer les bonnes stimulations S_m aux muscles afin de faire courir le robot au moyen de lois de réflexes (ou de positions de références pour le haut du corps). Suite à l'envoi d'une stimulation au muscle, ce dernier se contracte et une force virtuelle est engendrée qui est convertie en couple appliqué aux articulations. Le modèle utilisé pour simuler ces muscles est le modèle de Hill [35]. La figure 4.3 résume ce processus de contrôle.

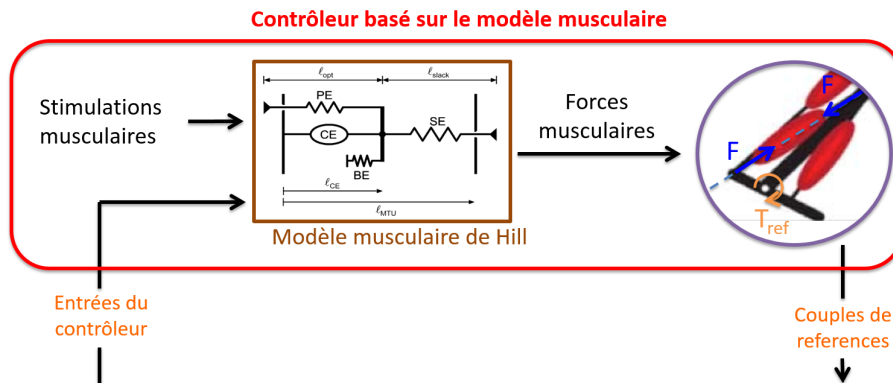


FIGURE 4.3 – Schéma de la conversion d'une stimulation musculaire en un couple de référence. Image adaptée de [5]

Notons que tous les outils pour simuler ces muscles étaient déjà présents dans le simulateur utilisé [67]. Le modèle existant a été amélioré en ajoutant un muscle aux jambes (le *rectus femoris*) ainsi que tous les muscles du haut du corps.

4.2.1 Modèle musculaire de Hill

Chaque groupe de muscles virtuels est représenté par une unité muscle-tendon (MTU pour *Muscle-Tendon Unit*). Dans son fonctionnement normal, chaque MTU est composé d'un élément contractile (CE pour *Contractile Element*) qui modélise la fibre musculaire, et d'un élément sériel élastique (SE pour *Serie elastic Element*) qui modélise le tendon connectant le muscle aux os. Le modèle est schématisé à la figure 4.4. Lorsque CE s'étend au delà de sa taille optimale ($l_{ce} > l_{opt}$), un élément parallèle élastique (PE pour *Parallel elastic Element*) est utilisé. De même, un élément tampon élastique (BE pour *Buffer elastic Element*) est activé lorsque la longueur de SE descend en dessous d'un certain seuil ($l_{mtu} - l_{ce} < l_{slack}$).

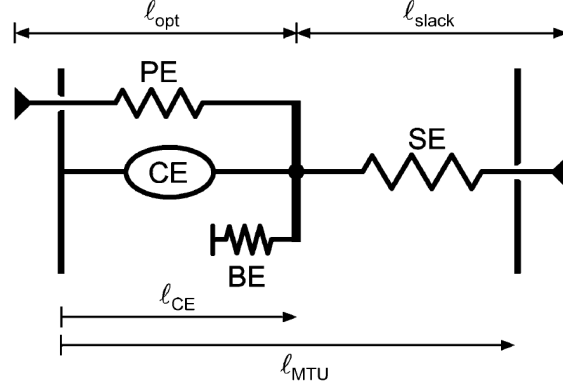


FIGURE 4.4 – Unité muscle-tendon (MTU) du modèle de Hill. Image issue de [29].

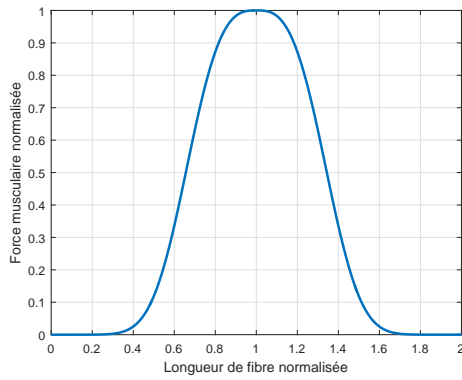


FIGURE 4.5 – Relation force-longueur de l'élément contractile

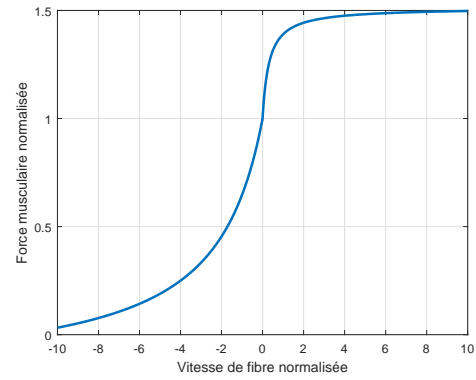


FIGURE 4.6 – Relation force-vitesse de l'élément contractile

Chaque MTU est attaché à des points physiques sur le robot. La longueur de la MTU l_{mtu} peut donc être connue via les angles des articulations du robot. La longueur de l'élément contractile (l_{ce}) dépend, elle, de nombreux paramètres parmi lesquels la position des articulations, l'état du muscle et le niveau d'activation du muscle A_m (voir section 4.2.2). La longueur de l'élément série élastique est alors également connue : $l_{se} = l_{mtu} - l_{ce}$. La force développée par le muscle $F_m = F_{se} = F_{ce} + F_{pe}$, où F_{se} , F_{ce} et F_{pe} sont respectivement les forces générées par le tendon, la fibre musculaire et l'élément parallèle élastique. Le CE produit une force

$$F_{ce} = A_m F_{max} f_l(l_{ce}) f_v(v_{ce}) \quad (4.1)$$

où F_{max} est la force isométrique maximale, et $f_l(l_{ce})$ et $f_v(v_{ce})$ sont les relations force-longueur et force-vitesse qui sont représentées aux figures 4.5 et 4.6 (pour la formulation de ces courbes, voir l'annexe C.1). Le muscle produit donc une force plus importante lorsque la longueur de l'élément contractile est proche de l_{opt} , et perd en efficacité lorsque le muscle est compressé ou étendu au-delà de cette longueur optimale. Le muscle perd également la capacité de générer une force lorsque celui-ci est contracté rapidement, et produit une plus grande force lorsqu'il est étendu ou immobile. Pour plus d'informations concernant ce modèle, le lecteur peut se référer à [29] et [30].

4.2.2 Activations des muscles

L'activation musculaire A_m est altérée par un processus électrochimique relativement lent basé sur le signal de stimulation musculaire S_m délivrée par le contrôleur bio-inspiré. Ces stimula-

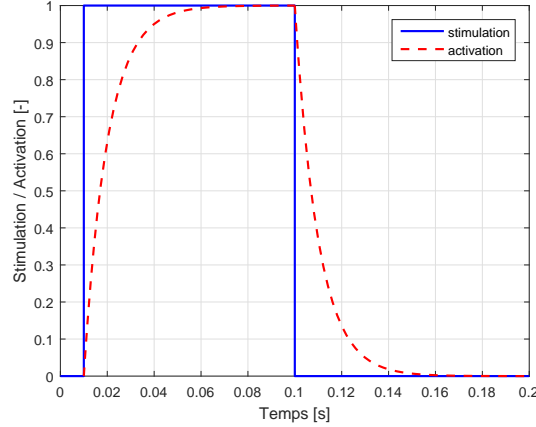


FIGURE 4.7 – Dynamique de l'activation des muscles.

tions sont bornées entre 0.01 et 1. Le délai de l'activation peut être modélisé par l'équation différentielle :

$$\tau \frac{dA_m(t)}{dt} = S_m(t) - A_m(t) \quad (4.2)$$

où τ est une constante de temps (fixée à 0.01 s). La réponse dynamique de l'activation des muscles est représentée à la figure 4.7.

De plus, du bruit est présent dans le réseau neuronal humain [25]. Afin de suivre la démarche bio-inspirée, du bruit est également ajouté aux stimulations envoyées aux muscles. Son amplitude est proportionnelle à la stimulation (avec un maximum de 5% de la stimulation) et de période de 0.1 seconde. Cela signifie que le bruit est actualisé toutes les 100 ms, et sa valeur à chaque pas de temps est une interpolation linéaire entre la valeur précédente et la nouvelle. Cette méthode permet de ne pas générer du bruit aléatoire à chaque pas de temps, ce qui reviendrait à ajouter en moyenne une contribution nulle (car le bruit aléatoire est centré en zéro).

4.2.3 Couples appliqués aux articulations

La force musculaire F_m génère un couple sur l'articulation (ou les deux articulations pour un muscle bi articulaire) à laquelle/auxquelles il est attaché : $\tau_m = \pm r_m(\phi)F_m$, où $r_m(\phi)$ est le bras de levier instantané. [29] utilise comme approximation pour $r_m(\phi)$ une valeur constante ou une relation cosinus :

$$r_m(\phi) = \begin{cases} r_0 & \text{pour la hanche} \\ r_0 \cos(\phi - \phi_{max}) & \text{pour le genou et la cheville} \end{cases} \quad (4.3)$$

$$(4.4)$$

où ϕ est la position de l'articulation, r_0 est le bras de levier constant (distance entre le point d'attache du muscle et l'articulation), et ϕ_{max} est l'angle produisant le bras de levier maximal. Il s'agit d'une bonne approximation (voir [23] pour la justification) en considérant le muscle comme droit. Le couple appliqué à chaque articulation est la somme des contributions des couples résultant des muscles liés à celle-ci. Il s'agit de la référence de couple qui est envoyée au contrôleur bas-niveau.

Le changement de longueur du MTU est quant à lui modélisé comme suit :

$$\Delta l_{mtu} = \begin{cases} \rho r_0 \sin(\phi - \phi_{ref}) & \text{pour la hanche} \\ \rho r_0 [\sin(\phi - \phi_{max}) - \sin(\phi_{ref} - \phi_{max})] & \text{pour le genou et la cheville} \end{cases} \quad (4.5)$$

$$(4.6)$$

où ϕ_{ref} est l'angle auquel $l_{mtu} = l_{opt} + l_{slack}$, et ρ est le facteur de pénétration (modélise le fait que la fibre musculaire n'est pas parfaitement alignée avec son point d'attache).

La caractérisation totale du muscle peut donc être connue à partir de 8 paramètres :

- 4 pour le MTU : F_{max} , v_{max} (vitesse maximale de contraction), l_{opt} et l_{slack} ;
- 4 pour l'attachement du muscle au squelette : r_0 , ρ , ϕ_{ref} et ϕ_{max} .

Ce sont ces paramètres qui doivent donc être trouvés pour chaque groupe de muscles (avec en plus la masse m et la proportion de fibres lentes λ pour le calcul de l'énergie consommée par le muscle, voir section 4.7.2).

Limites software

Le modèle musculaire de Hill ne prend pas en compte le fait que les mouvements des articulations sont limités. Pour inclure ces limites dans notre modèle, nous utilisons des *joint soft limits* introduits dans [29] et déjà présents dans le simulateur. L'idée est d'appliquer un couple supplémentaire à chaque articulation qui s'étend au-delà de sa limite. Dans la pratique, seul le genou est impacté par cette limite.

4.2.4 Mise à échelle dynamique

La mise à échelle dynamique des caractéristiques des muscles pour le COMAN est développée dans [65] (voir D.5). Le redimensionnement a pour but de s'adapter à la masse et à la taille du COMAN qui est plus léger et plus petit que le modèle utilisé par H. Geyer duquel provient la caractérisation des muscles.

Les rapports de masse et de longueur de jambes entre le COMAN (M_c et L_c) et le modèle utilisé dans [29] (M_H et L_h) sont les suivants :

$$\frac{L_C}{L_H} = \frac{42.7cm}{100cm} = 0.427 \quad \frac{M_C}{M_H} = \frac{28.3kg}{80kg} = 0.354 \quad (4.7)$$

À partir de ces rapports, les différents facteurs ci-dessous sont trouvés :

$$\frac{L_C}{L_H} = 0.427 \quad \text{pour } l_{opt}, l_{slack} \text{ et } r_0 \quad (4.8)$$

$$\frac{F_c}{F_H} = \frac{M_C}{M_H} = 0.354 \quad \text{pour } F_{max} \text{ et } m \quad (4.9)$$

$$\frac{v_c}{v_H} = \sqrt{\frac{L_H}{L_C}} \approx 1.53 \quad \text{pour } v_{max} \quad (4.10)$$

Les paramètres ϕ_{max} , ϕ_{ref} , ρ et λ restent inchangés selon le modèle utilisé.

4.2.5 Avantages et approximations du modèle

Le modèle musculaire implémenté se veut le plus proche possible de la réalité et comporte de nombreux avantages, notamment grâce à la dynamique engendrée par les muscles. Le couple fourni aux articulations sur base de stimulations sera beaucoup plus lisse et réaliste qu'un contrôleur classique, notamment grâce aux propriétés viscoélastiques des muscles.

Néanmoins, le modèle se base sur certaines approximations, la plus grosse étant qu'un muscle agit seulement dans un plan, ce qui n'est évidemment pas le cas dans la réalité. De plus, les paramètres des muscles résultent d'une pondération approximative de la contribution des muscles réels dans le mouvement souhaité (du moins pour le haut du corps, voir section 4.3.2).

4.3 Muscles virtuels du COMAN

4.3.1 Muscles des jambes

La caractérisation et l'implémentation des muscles du bas du corps étaient déjà présentes dans le simulateur [67], excepté pour le *recturus femoris* (RF) (voir figure 4.8). Celui-ci est présent dans les modèles [70] (sur lequel le contrôleur est basé) et [60]. Sa caractérisation provient de [60] après avoir subi une mise à échelle dynamique (voir section 4.2.4). Étant donné que nous ne travaillons qu'en deux dimensions dans le cadre de ce mémoire, seuls les muscles du plan sagittal seront utilisés et donc introduits ici.

Les huit muscles de Hill par jambes sont les suivants et sont représentés schématiquement à la figure 4.8 :

- 2 muscles mono articulaires agissant au niveau de la cheville : le *soleus* (SOL) et le *tibialis anterior* (TA) ;
- 1 muscle mono articulaire agissant au niveau du genou : le *vasti* VAS ;
- 1 muscle bi articulaire agissant à la cheville et au genou : le *gastrocnemius* (GAS) ;
- 2 muscles mono articulaires agissant au niveau de la hanche : le *gluteus* (GLU) et le *hip flexor* (HFL) ;
- 2 muscles bi articulaires agissant au niveau du genou et de la hanche : le *hamstring* (HAM) et *rectus femoris* (RF).

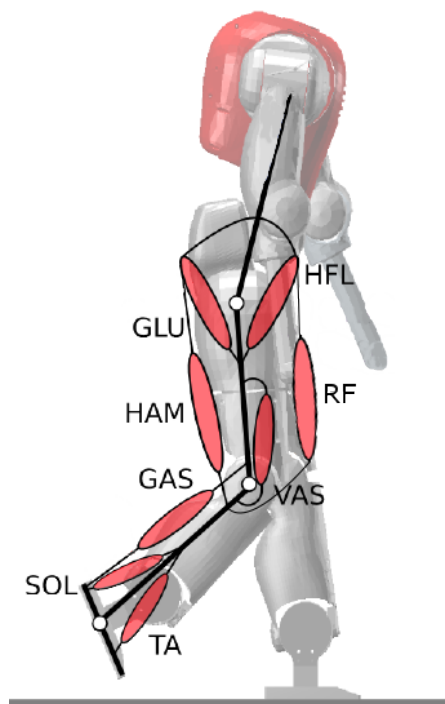


FIGURE 4.8 – Représentation schématique des huit muscles sagittaux de Hill de la jambe gauche. Image adaptée de [67].

La caractérisation de ces muscles se trouve à l'annexe C.2.1.

4.3.2 Muscles des bras et du torse

Le modèle musculaire du haut du corps n'était pas présent initialement dans le simulateur et a été développé dans le but d'avoir un modèle plus proche de celui d'un être humain. Initialement,

toutes les articulations étaient contrôlées en position (et non en couple). Donc, l'ajout d'un modèle musculaire a permis d'avoir un mouvement beaucoup plus fluide. Afin d'avoir un modèle complet, les muscles liés aux articulations agissant dans trois plans de référence (sagittal, frontal et transversal) sont modélisés.

La caractérisation est basée sur le document rédigé par N. Van der Noot [65] (non publié et en anglais) disponible à l'annexe D. On utilise le programme OpenSim [20] qui permet de modéliser, simuler, contrôler, et analyser un système musculo-squelettique. Le modèle utilisé pour les muscles du torse est *gait2354_simbody* [3] [2] [21] [75], et celui pour les muscles des bras est *UpperExtremityModel* [36].

Tous les muscles sont représentés schématiquement à la figure 4.8. Il ne s'agit que de muscles mono articulaires. Les 6 muscles du torse et les 8 muscles pour chaque bras sont les suivants :

- 2 muscles agissant au niveau du torse dans le plan sagittal : BET (*back extension*) et BFL (*back flexion*) ;
- 2 muscles agissant au niveau du torse dans le plan frontal : BTR (*back tilt right*) et BTL (*back tilt left*) ;
- 2 muscles agissant au niveau du torse dans le plan transversal : BRR (*back rotation right*) et BRL (*back rotation left*) ;
- 2 muscles agissant au niveau de l'épaule dans le plan sagittal : SET (*shoulder extension*) et SFL (*shoulder flexion*) ;
- 2 muscles agissant au niveau de l'épaule dans le plan frontal : SAB (*shoulder abduction*) et SAD (*shoulder adduction*) ;
- 2 muscles agissant au niveau de l'épaule dans le plan transversal : SIR (*shoulder internal rotation*) et SER (*shoulder external rotation*) ;
- 2 muscles agissant au niveau du coude dans le plan sagittal : EFL (*elbow flexion*) et EET (*elbow extension*).

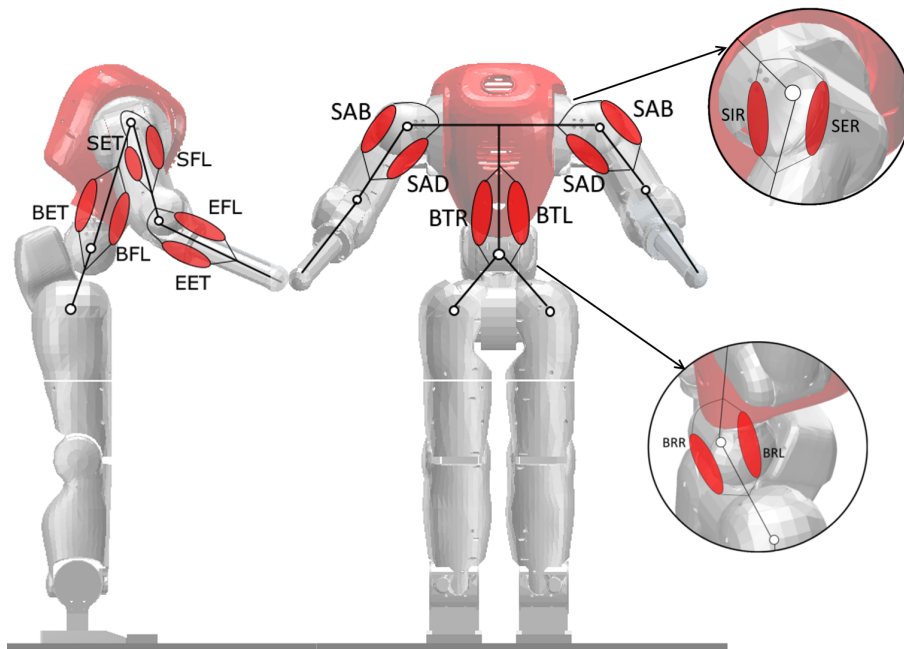


FIGURE 4.9 – Représentation schématique des muscles de Hill pour le haut du corps. À gauche les muscles du plan sagittal, au centre ceux du plan frontal, et à droite ceux du plan transversal.

Les paramètres de ces muscles sont donnés à l'annexe C.2.2.

4.4 Contrôle du torse et des bras

Le haut du corps est surtout utile pour maintenir un équilibre latéral lors de la locomotion. Étant donné qu'on étudie uniquement la démarche 2D, les articulations non sagittales du torse sont contrôlées en imposant une position de référence nulle. L'articulation sagittale du torse est également imposée à une référence nulle. Cette articulation peut être comparée à la colonne verticale chez l'être humain, qui reste droite lors de la locomotion. L'orientation du haut du corps est alors contrôlée au moyen de la position des hanches (via les muscles GLU, HAM, HFL et RF). Le tableau 4.2 résume le contrôle (couple ou position) des articulations du haut du corps.

| Corps | Articulations | Contrôles | Références |
|---------|---------------|-----------------------------|---|
| Épaules | Sagittales | Couples (modèle musculaire) | $\propto (\theta_{hanche,sag}^g - \theta_{hanche,sag}^d)$ |
| | Latérales | Positions | ± 0.09 rad |
| | Transversales | Positions | ± 0.14 rad |
| Coudes | Sagittales | Couples (modèle musculaire) | 0.25 rad |
| Torse | Sagittale | Position | 0.0 rad |
| | Latérale | Position | 0.0 rad |
| | Transversale | Position | 0.0 rad |

TABLE 4.2 – Récapitulatif du contrôle des articulations des bras et du torse.

Lors de la locomotion humaine, chaque bras se balance en suivant le mouvement de la jambe opposée. Ce mouvement crée un moment cinétique dans la direction opposée à la rotation du bas du corps, ce qui réduit le moment cinétique total du corps [49]. Bien que le balancement des bras ne soit pas indispensable pour obtenir un résultat de marche, celui-ci est modélisé car il est bénéfique pour la stabilité de la locomotion et réduit le coût énergétique de la démarche [13].

Les lois de contrôle utilisées, basées sur [70], sont celles utilisées pour la marche dans [67]. Les articulations non sagittales des bras sont contrôlées avec une position de référence constante, à savoir ± 0.09 radian pour l'articulation latérale de l'épaule, et ± 0.14 radian pour l'articulation transversale de l'épaule (+ pour le bras gauche, et – pour le bras droit).

Les articulations sagittales sont contrôlées avec une référence de couple, au moyen du modèle musculaire du haut du corps. Les stimulations à envoyer aux muscles sont calculées en utilisant un contrôleur proportionnel à l'erreur de position (voir équations (4.13) à (4.20)). La position de référence pour l'articulation du coude est constante et vaut $\theta_{epaule,ref} = -0.25$ radian, et la position de référence des articulations sagittales des épaules est proportionnelle à la différence d'angle au niveau des hanches :

$$\theta_{epaule,ref}^d = -0.3 \left(\theta_{hanche,sag}^g - \theta_{hanche,sag}^d \right) - 0.3 \quad (4.11)$$

$$\theta_{epaule,ref}^g = 0.3 \left(\theta_{hanche,sag}^g - \theta_{hanche,sag}^d \right) - 0.3 \quad (4.12)$$

Le modèle musculaire du haut du corps a été implémenté pour tous les muscles pour être complet, mais seuls les muscles agissant au niveau des articulations sagittales des bras sont donc utilisés dans ce contrôleur. Les stimulations envoyées à ces muscles sont définies (avec $K_p = 5$) :

$$S_{SET} = \begin{cases} 0.01 & \text{si } \theta_{epaule} \geq \theta_{epaule,ref} \\ K_p (\theta_{epaule,ref} - \theta_{epaule}) & \text{si } \theta_{epaule} < \theta_{epaule,ref} \end{cases} \quad (4.13)$$

$$(4.14)$$

$$S_{SFL} = \begin{cases} 0.01 & \text{si } \theta_{epaule} \leq \theta_{epaule,ref} \\ K_p (\theta_{epaule,ref} - \theta_{epaule}) & \text{si } \theta_{epaule} > \theta_{epaule,ref} \end{cases} \quad (4.15)$$

$$(4.16)$$

$$S_{EET} = \begin{cases} 0.01 & \text{si } \theta_{coude} \geq \theta_{coude,ref} \\ K_p (\theta_{coude,ref} - \theta_{coude}) & \text{si } \theta_{coude} < \theta_{coude,ref} \end{cases} \quad (4.17)$$

$$(4.18)$$

$$S_{EFL} = \begin{cases} 0.01 & \text{si } \theta_{coude} \leq \theta_{coude,ref} \\ K_p (\theta_{coude,ref} - \theta_{coude}) & \text{si } \theta_{coude} > \theta_{coude,ref} \end{cases} \quad (4.19)$$

$$(4.20)$$

Le contrôleur musculaire pour le torse et les articulations non-sagittales des bras est décrit à l'annexe C.3.

Dans [67], les articulations sagittales du haut du corps du COMAN sont contrôlées avec des positions de références. Le modèle musculaire et le contrôle en couple permettent d'avoir des mouvements des bras plus fluides et donc plus proches de ceux d'un être humain.

Une grande différence dans le mouvement du haut du corps entre la course et la marche est le fléchissement des coudes : ils sont plus fléchis à vitesse élevée [10]. Ce fléchissement place le centre de masse du bras plus haut (proche de l'épaule), ce qui permet de balancer les bras avec un coût énergétique plus faible (et d'augmenter leur fréquence propre). La fréquence du mouvement des bras augmentant avec la vitesse de locomotion (car liée au mouvement des jambes), le fléchissement est bénéfique. Néanmoins, ce n'est pas le cas pour le robot COMAN car la masse de son avant-bras est très faible comparée à celle d'un être humain (il n'a pas de main, et la masse est principalement répartie au niveau des articulations à cause des moteurs). Ainsi, la fréquence propre des bras évolue avec une différence maximale de 0.026 Hz pour l'angle du coude compris entre 0° et -120°. De plus, la contribution de la dynamique des bras dans les phases de propulsion et de support du centre de masse pour la course est négligeable (moins d'1% de l'accélération totale du centre de masse) [31]. C'est pour ces raisons que nous gardons les mêmes lois de contrôle pour la marche et la course.

4.5 Réflexes bio-inspirés pour la marche et la course

Pour rappel, seuls les articulations sagittales des jambes sont contrôlées au moyen de stimulations envoyées aux muscles à partir de lois qui modélisent les réflexes musculaires. Les articulations non sagittales des jambes sont contrôlées en position de références nulles (voir tableau 4.3).

| Corps | Articulations | Contrôle |
|--------|----------------|-----------------------------|
| Jambes | Sagittales | Couples (modèle musculaire) |
| | Non-sagittales | Positions nulles |

TABLE 4.3 – Récapitulatif du contrôle des articulations des jambes.

Les lois de contrôle qui permettent d'envoyer les stimulations aux muscles afin d'obtenir la démarche souhaitée sont celles proposées dans [70]. Elles sont elles-mêmes basées sur les lois de réflexes bio-inspirés proposées dans [29], qui ont déjà été implémentées et donné de bons résultats de marche sur le COMAN [5]. Les modifications effectuées par rapport à [29] permettent d'obtenir des résultats tant de marche que de course avec les mêmes lois de contrôle. C'est pourquoi nous avons décidé d'utiliser ce contrôleur.

Il y a deux ensembles de lois de contrôle différents en fonction de l'état dans lequel se trouve la jambe : phase d'appui ou phase oscillante. L'article [70] définit deux états supplémentaires : initiation à la phase oscillante, et préparation à la phase d'appui. Les lois de contrôle pour certains

muscles sont modifiées pour ces deux sous-états afin d'obtenir des résultats de course.

Un pied en phase oscillante passe en préparation à la phase d'appui si $d < d_{sp}$. d est la distance horizontale entre le centre de masse du robot et le pied, normalisé par la longueur du pied, et d_{sp} est un paramètre à optimiser.

Un pied en phase d'appui passe en initiation à la phase oscillante si $d > d_{si}$ ou s'il y a un double appui (les deux pieds sont en phase d'appui). d_{si} est également un paramètre à optimiser.

4.5.1 Lois de contrôle

En général, le contrôle de chaque muscle est basé sur trois différentes lois de contrôle : la rétroaction positive de force, la rétroaction positive de longueur de fibre musculaire, et le contrôle PD de la position articulaire. Chacune de ces lois ainsi que leurs utilités sont décrites dans cette section.

Les lois de contrôle reproduisent le délai temporel introduit par la propagation du signal électrique du cerveau vers les muscles. Ce délai Δt vaut 2.5 ms pour les MTU connectés à la hanche, 10 ms pour ceux connectés à la cheville, 5 ms pour le VAS et les contacts avec le sol [29] (avec une mise à échelle de la taille du robot).

Rétroaction positive de force

Pour un MTU m , la rétroaction positive de force est définie par :

$$S_m^F = G_m \frac{F_m(t - \Delta t_m)}{F_m^{max}} \quad (4.21)$$

où $F_m(t - \Delta t_m)$ est la force développée par le muscle m avec un délai Δt_m , F_m^{max} est la force isométrique maximale du muscle, et G_m est un paramètre de gain constant et positif. Ce gain est un paramètre à optimiser et est différent pour chaque MTU.

La stimulation produite est donc proportionnelle à la force générée par le MTU m au temps $(t - \Delta t_m)$. Plus la force développée par le muscle est forte, plus ce muscle est activé, ce qui génère encore plus de force. Néanmoins cette force ne peut pas augmenter indéfiniment, car elle suit une relation non linéaire en fonction de la longueur et de la vitesse de contraction de la fibre musculaire (voir l'équation (4.1)). Quand la force musculaire commence à diminuer à cause de l_{ce} et v_{ce} , la stimulation diminue également. Cette loi permet de produire des stimulations hautement non linéaires (grâce au modèle musculaire) observées lors de la démarche humaine avec seulement un paramètre à déterminer [70].

Cette loi est la principale source d'activation des muscles SOL, GAS et VAS lors de la phase d'appui.

Par souci de clarté, dans la suite de ce document on utilise $\tilde{F}_m = \frac{F_m(t - \Delta t_m)}{F_m^{max}}$

Rétroaction positive de longueur de fibre musculaire

La rétroaction positive de longueur de fibre musculaire l_{ce} est définie par :

$$S_m^L = \left\{ G_m \left(\frac{l_m^{ce}(t - \Delta t_m)}{l_m^{opt}} - l_{off,m} \right) \right\}_+ \quad (4.22)$$

où $l_m^{ce}(t - \Delta t_m)$ est la longueur de fibre musculaire du muscle m avec un délai Δt_m , l_{opt} est la longueur optimale de fibre, et G_m et $l_{off,m}$ sont des paramètres positifs à déterminer (gain et offset). $\{\}_+$ signifie que seule une valeur positive est prise en compte (sinon la stimulation vaut zéro).

Cette loi modélise un réflexe d'étirement du muscle : le muscle est activé lorsque sa longueur de fibre est supérieure à une certaine limite. Plus cette limite est dépassée, plus l'activation sera élevée.

Elle est principalement utilisée pendant la phase oscillante pour les muscles HFL et TA. L'activation du HFL permet une flexion de la hanche pour amener la jambe vers l'avant. L'activation du TA permet principalement d'avoir un mouvement de dorsiflexion de la cheville afin d'éviter que le robot ne trébuche sur le bout du pied (orteils).

Dans la suite de ce document on utilise $\tilde{l}_m^{CE} = \frac{l_m^{ce}(t - \Delta t_m)}{l_m^{opt}}$

Contrôle PD de la position articulaire

Enfin, le contrôle musculaire proportionnel-dérivé est défini par :

$$S_m^\theta = \left| \left\{ K_m (\theta(t - \Delta t_m) - \theta_m) + D_m \dot{\theta}(t - \Delta t_m) \right\}_\pm \right| \quad (4.23)$$

où $\theta(t - \Delta t_m)$ est la position angulaire de l'articulation sur laquelle agit le muscle m avec un délai Δt_m , $\dot{\theta}(t - \Delta t_m)$ est sa vitesse angulaire, et K_m , D_m et θ_m sont 3 paramètres à déterminer. K_m est le gain proportionnel, D_m est le gain dérivé, et θ_m est la position de référence. Le signe des accolades est positif si le couple généré par le muscle m est dans la direction opposée à θ , et négatif autrement.

Il s'agit d'un contrôleur PD classique, dont le but est de forcer la position articulaire θ vers la position de référence θ_m avec un amortissement sur la vitesse angulaire de l'articulation. L'activation des muscles s'effectue seulement après un délai Δ_m , et chaque muscle génère un couple qui entraîne l'articulation dans une seule direction. Cette loi est utilisée par des muscles antagonistes (dont les mouvements générés s'opposent) avec une même référence de position.

Ce type de contrôle est employé pendant la phase d'appui par les muscles liés à la hanche pour maintenir l'orientation du haut du corps à une certaine position (souvent pour avoir le dos droit). Il est également utilisé lors de la préparation à la phase d'appui pour préparer le contact du pied avec le sol.

4.5.2 Phase d'appui

Chaque muscle a une stimulation initiale $S_{0,m}$. Il s'agit de paramètres qui sont optimisés et dans la plupart des cas non dominants par rapport aux autres termes. Ces stimulations initiales sont différentes en fonction de la phase dans laquelle se trouve le pied. On les note $S_{0,m,st}$ pour la phase d'appui (*stance phase*).

Les muscles SOL et GAS sont stimulés par des rétroactions positives en force. Ce sont les muscles qui sont principalement responsables de la propulsion.

$$S_{SOL} = S_{0,SOL,st} + G_{SOL} \tilde{F}_{SOL} \quad (4.24)$$

$$S_{GAS} = S_{0,GAS,st} + G_{GAS} \tilde{F}_{GAS} \quad (4.25)$$

L'activation du muscle TA permet principalement de redresser la cheville lors de la phase oscillante au moyen d'une rétroaction positive en longueur de fibre musculaire. Cette activation est supprimée durant la phase d'appui, et ce proportionnellement à la force produite par le muscle SOL :

$$S_{TA} = S_{0,TA,st} + \left\{ G_{TA,st} \tilde{l}_{TA}^{CE} - l_{off,TA,st} \right\}_+ - G_{SOL,TA} \tilde{F}_{SOL} \quad (4.26)$$

La rétroaction positive en force agissant sur le muscle VAS permet de créer un grand couple d'extension au niveau du genou lors du contact avec le sol. Afin d'éviter une hyper extension du genou, la stimulation produite est supprimée lorsque le genou est en extension ($\dot{\theta}_{genou} < 0$) et que sa position angulaire dépasse la limite $\theta_{genou,off}$ (paramètre à optimiser).

$$S_{VAS} = S_{0,VAS,st} + G_{VAS} \tilde{F}_{VAS} + \{k_{\theta}(\theta_{genou} - \theta_{genou,off})\}_{-, \dot{\theta}_{genou} < 0} \quad (4.27)$$

L'activation du muscle RF est simplement induite par une stimulation constante :

$$S_{RF} = S_{0,RF,st} \quad (4.28)$$

Les muscles HAM, GLU et HFL sont responsables du maintien de l'orientation du haut du corps au moyen d'un contrôle PD. Si le tronc du robot est trop penché en avant (par rapport à la référence $\theta_{torse,ref}$), ce sont les muscles HAM et GLU qui sont activés, sinon c'est le muscle HFL. En cas de double appui (les deux pieds sont en phase d'appui, uniquement possible lors de la marche), ces lois de contrôle sont appliquées au seul pied qui est devant (celui qui entre en phase d'appui).

$$S_{HAM} = S_{0,HAM,st} + \left| \left\{ K_{HAM}(\theta_{torse} - \theta_{torse,ref}) + D_{HAM} \dot{\theta}_{torse} \right\}_{+,devant} \right| \quad (4.29)$$

$$S_{GLU} = S_{0,GLU,st} + \left| \left\{ K_{GLU}(\theta_{torse} - \theta_{torse,ref}) + D_{GLU} \dot{\theta}_{torse} \right\}_{+,devant} \right| \quad (4.30)$$

$$S_{HFL} = S_{0,HFL,st} + \left| \left\{ K_{HFL}(\theta_{torse} - \theta_{torse,ref}) + D_{HFL} \dot{\theta}_{torse} \right\}_{-,devant} \right| \quad (4.31)$$

Initiation à la phase oscillante

Vers la fin de la phase d'appui, le contrôleur entre en initiation à la phase oscillante lorsque la distance entre le centre de masse et le pied est supérieure à une certaine limite à optimiser (d_{si} , pour *swing initiation*), ou si le pied opposé entre en phase d'appui (double appui). Durant cette initiation, des excitations SI_m (bornées entre 0 et 1) sont ajoutées ou supprimées aux lois de contrôle décrites précédemment.

La stimulation du muscle VAS est diminuée afin de favoriser la flexion du genou nécessaire au début de la phase oscillante :

$$S_{VAS} = S_{VAS} - SI_{VAS} \quad (4.32)$$

$$(4.33)$$

La diminution de la stimulation envoyée au GLU, ainsi que l'augmentation des stimulations envoyées aux HFL et RF permettent de favoriser la flexion des hanches pour la phase oscillante :

$$S_{GLU} = S_{GLU} - SI_{GLU} \quad (4.34)$$

$$S_{HFL} = S_{HFL} + SI_{HFL} \quad (4.35)$$

$$S_{RF} = S_{RF} + SI_{RF} \quad (4.36)$$

4.5.3 Phase oscillante

Comme lors de la phase d'appui, chaque muscle a une stimulation initiale $S_{0,m}$. On les note $S_{0,m,sw}$ pour la phase oscillante (*swing phase*). Le mouvement de la jambe en phase oscillante nécessite beaucoup moins d'activations de ses différents muscles. Ainsi, les muscles SOL, GAS, RF et VAS sont simplement activés par des stimulations constantes (et souvent de faible intensité) :

$$S_{SOL} = S_{0,SOL,sw} \quad (4.37)$$

$$S_{GAS} = S_{0,GAS,sw} \quad (4.38)$$

$$S_{RF} = S_{0,RF,sw} \quad (4.39)$$

$$S_{VAS} = S_{0,VAS,sw} \quad (4.40)$$

Comme on l'a déjà mentionné, le muscle TA permet d'éviter que le robot trébuche sur le bout de ses pieds lors de la phase oscillante. Cela est réalisé au moyen d'une rétroaction positive en longueur de fibre musculaire :

$$S_{TA} = S_{0,TA,sw} + \left\{ G_{TA,sw} \tilde{l}_{TA}^{CE} - \tilde{l}_{off,TA,sw} \right\}_+ \quad (4.41)$$

Les muscles HAM et GLU sont activés au moyen d'une rétroaction positive en force. Ceux-ci permettent l'extension de la hanche pour préparer le contact avec le sol. Le muscle bi articulaire HAM permet également de contrer une hyper extension du genou avant d'entrer en phase d'appui.

$$S_{HAM} = S_{0,HAM,sw} + G_{HAM} \tilde{F}_{HAM} \quad (4.42)$$

$$S_{GLU} = S_{0,GLU,sw} + G_{GLU} \tilde{F}_{GLU} \quad (4.43)$$

L'activation du muscle HFL permet la flexion de la hanche au début de la phase oscillante au moyen d'une rétroaction positive en longueur de fibre. Cette contribution est annulée lorsque le muscle HAM est étendu au-delà d'une certaine limite $l_{off,HAM,HFL}$ en fin de phase. L'amplitude de la stimulation envoyée au HFL dépend également de la position angulaire du haut du corps lorsque la jambe entre en phase oscillante ($\theta_{0,torse}$). Si celui-ci est trop penché vers l'avant par rapport à $\theta_{torse,ref}$ (référence identique à la phase d'appui), l'activation sera plus importante.

$$S_{HFL} = S_{0,HFL,sw} + \left\{ G_{HFL} \tilde{l}_{HFL}^{CE} - l_{off,HFL} \right\}_+ - \left\{ G_{HAM,HFL} \tilde{l}_{HAM}^{CE} - l_{off,HAM,HFL} \right\}_+ + k_{\Theta} (\theta_{0,torse} - \theta_{torse,ref}) \quad (4.44)$$

Préparation à la phase d'appui

A fin de la phase oscillante, le contrôleur rentre en préparation à la phase d'appui lorsque la distance entre le centre de masse et le pied est inférieure à une certaine limite à optimiser (d_{sp} , pour *stance preparation*). Les stimulations pour les muscles VAS, GLU et HFL sont alors déterminées au moyen d'un contrôleur PD.

L'activation du muscle VAS a pour but de positionner la position angulaire du genou vers une position de référence $\theta_{genou,ref}$ afin de préparer le contact avec le sol :

$$S_{VAS} = S_{0,VAS,sw} + \left| \left\{ K_{sp,VAS} (\theta_{genou} - \theta_{genou,ref}) + D_{sp,VAS} \dot{\theta}_k \right\}_+ \right| \quad (4.45)$$

Les muscles GLU et HFL imposent, eux, la position de la hanche¹ pour préparer la phase d'appui :

$$S_{GLU} = S_{0,GLU,sw} + \left| \left\{ K_{sp,GLU}(\theta_{hanche} - \theta_{hanche,ref}) + D_{sp,GLU}\dot{\theta}_h \right\}_- \right| \quad (4.46)$$

$$S_{HFL} = S_{0,HFL,sw} + \left| \left\{ K_{sp,HFL}(\theta_{hanche} - \theta_{hanche,ref}) + D_{sp,HFL}\dot{\theta}_h \right\}_+ \right| \quad (4.47)$$

4.5.4 Paramètres à optimiser

Les bornes des 54 paramètres à déterminer issus du contrôleur présenté ci-dessus sont reprises dans le tableau 4.4. En plus de ces paramètres, des valeurs initiales pour les positions et vitesses des articulations sont aussi à optimiser. Celles-ci sont au nombre de 17 et sont reprises à la section 4.6.

| | min | max | | min | max | | min | max |
|----------------|-------|------|-------------------|------|------|-----------------------|-------|------|
| $S_{0,SOL,st}$ | 0.01 | 0.05 | G_{GAS} | 0.0 | 20.0 | $\theta_{genou,off}$ | 0.05 | 0.2 |
| $S_{0,GAS,st}$ | 0.01 | 0.05 | G_{VAS} | 0.82 | 5.0 | k_θ | 1.0 | 15.0 |
| $S_{0,TA,st}$ | 0.01 | 0.05 | G_{HAM} | 0.0 | 3.0 | $\theta_{torse,ref}$ | 0.017 | 0.11 |
| $S_{0,VAS,st}$ | 0.047 | 0.71 | G_{GLU} | 0.0 | 2.0 | K_{HAM} | 1.0 | 14.0 |
| $S_{0,RF,st}$ | 0.01 | 0.5 | $G_{TA,st}$ | 0.55 | 5.0 | K_{GLU} | 0.5 | 14.0 |
| $S_{0,HAM,st}$ | 0.01 | 0.32 | $G_{TA,sw}$ | 0.55 | 5.0 | K_{HFL} | 1.0 | 18.0 |
| $S_{0,GLU,st}$ | 0.01 | 0.32 | G_{HFL} | 0.17 | 3.0 | D_{HAM} | 0.0 | 0.5 |
| $S_{0,HFL,st}$ | 0.01 | 0.32 | $G_{HAM,HFL}$ | 0.0 | 10.0 | D_{GLU} | 0.0 | 0.5 |
| $S_{0,SOL,sw}$ | 0.01 | 0.05 | $l_{off,TA,st}$ | 0.5 | 0.8 | D_{HFL} | 0.0 | 0.5 |
| $S_{0,GAS,sw}$ | 0.01 | 0.2 | $l_{off,TA,sw}$ | 0.5 | 0.8 | k_Θ | 1.0 | 5.7 |
| $S_{0,TA,sw}$ | 0.01 | 0.05 | $l_{off,HFL}$ | 0.0 | 0.85 | $\theta_{genou,ref}$ | 0.0 | 3.0 |
| $S_{0,VAS,sw}$ | 0.01 | 0.1 | $l_{off,HAM,HFL}$ | 0.0 | 0.71 | $\theta_{hanche,ref}$ | 0.15 | 0.9 |
| $S_{0,RF,sw}$ | 0.01 | 0.05 | d_{si} | 0.2 | 0.7 | $K_{sp,VAS}$ | 0.0 | 5.0 |
| $S_{0,HAM,sw}$ | 0.01 | 0.1 | d_{sp} | -0.3 | 0.1 | $K_{sp,GLU}$ | 0.0 | 5.0 |
| $S_{0,GLU,sw}$ | 0.01 | 0.05 | SI_{VAS} | 0.3 | 1.0 | $K_{sp,HFL}$ | 0.0 | 5.0 |
| $S_{0,HFL,sw}$ | 0.01 | 0.05 | SI_{GLU} | 0.2 | 1.0 | $D_{sp,VAS}$ | 0.0 | 0.1 |
| G_{SOL} | 0.97 | 5.0 | SI_{HFL} | 0.0 | 1.0 | $D_{sp,GLU}$ | 0.0 | 0.1 |
| $G_{SOL,TA}$ | 0.4 | 7.0 | SI_{RF} | 0.0 | 1.0 | $D_{sp,HFL}$ | 0.0 | 0.1 |

TABLE 4.4 – Bornes des 54 paramètres des lois de contrôle à optimiser.

4.6 Initialisation du COMAN pour la course

Afin d'obtenir un résultat de course avec le contrôleur présenté à la section 4.5, l'état initial du robot COMAN a beaucoup d'importance. En effet, avec un contrôleur et une fonction d'objectif identiques, les résultats provenant des optimisations varient en fonction de l'état initial du robot² (l'optimisation a tendance à converger vers un optimum local de marche).

1. Dans l'article [70], la position de référence de la hanche est ajustée selon le *SIMBICON balance feedback* [77] : $\theta_{hanche,ref} = \theta_{0,hanche,ref} + c_d d + c_v v$. $\theta_{0,hanche}$ est la référence initiale, d est la distance horizontale entre le centre de masse du robot et le pied en phase d'appui, v est la vitesse d'avance du centre de masse, et c_d et c_v sont des gains. Ce contrôle n'est pas nécessaire dans notre cas (utile pour être plus robuste à des perturbations extérieures), et la vitesse d'avance du centre de masse n'est pas une entrée du contrôleur.

2. Pour plus d'informations concernant les différents états initiaux testés, voir l'annexe A

L'idée est de permettre au robot de commencer sa démarche avec des positions et vitesses articulaires similaires à celles obtenues lors de la course. La vitesse d'avance initiale du robot est fixée car l'optimisateur a tendance à maximiser sa valeur pour avancer le plus loin possible au début de la simulation (voir fonction d'objectif section 4.7.1). Les meilleures valeurs pour les autres positions et vitesses articulaires sont déterminées au moyen du processus d'optimisation. Les paramètres suivants initiaux sont à optimiser :

- La position verticale du robot T_z [m], ainsi que sa vitesse [m/s] ;
- La position angulaire sagittale du robot R_y [rad], ainsi que sa vitesse [rad/s] ;
- Les positions des articulations sagittales des jambes (θ_{hanche}^d , θ_{genou}^d et $\theta_{cheville}^d$ pour la jambe droite, et θ_{hanche}^d , θ_{genou}^d et $\theta_{cheville}^d$ pour la jambe gauche) [rad], ainsi que leurs vitesses ($\dot{\theta}_{hanche}^d$, $\dot{\theta}_{genou}^d$ et $\dot{\theta}_{cheville}^d$ pour la jambe droite, et $\dot{\theta}_{hanche}^d$, $\dot{\theta}_{genou}^d$ et $\dot{\theta}_{cheville}^d$ pour la jambe gauche) [rad/s] ;
- La vitesse angulaire de l'épaule droite [rad/s]. La vitesse de l'épaule gauche est l'opposé, et leurs positions sont déterminées par l'équation (4.12).

Pour déterminer les bornes de ces paramètres, les résultats de course de [70] sont utilisés. On privilégie un état initial où le robot est en phase de vol, où la jambe droite est/va entrer en phase d'appui, et où la jambe gauche est en phase oscillante. Les bornes des paramètres à optimiser sont listées dans le tableau 4.5. Une visualisation des positions articulaires initiales menant à un résultat de course (avec une vitesse initiale d'avance de 1.6m/s) est montrée à la figure 4.10.

| | min | max | | min | max |
|-----------------------|-------|-------|-----------------------------|------|-----|
| T_z | 0.46 | 0.58 | \dot{T}_z | -0.5 | 0.5 |
| R_y | 0.0 | 0.3 | \dot{R}_y | -0.5 | 0.5 |
| θ_{hanche}^d | -0.87 | -0.35 | $\dot{\theta}_{hanche}^d$ | -0.5 | 5.0 |
| θ_{genou}^d | 0.0 | 0.87 | $\dot{\theta}_{genou}^d$ | -0.5 | 5.0 |
| $\theta_{cheville}^d$ | -0.09 | 0.09 | $\dot{\theta}_{cheville}^d$ | -0.5 | 5.0 |
| θ_{hanche}^g | -0.09 | 0.35 | $\dot{\theta}_{hanche}^g$ | -0.5 | 5.0 |
| θ_{genou}^g | 0.35 | 1.75 | $\dot{\theta}_{genou}^g$ | -0.5 | 5.0 |
| $\theta_{cheville}^g$ | 0.09 | 0.45 | $\dot{\theta}_{cheville}^g$ | -0.5 | 5.0 |
| | | | $\dot{\theta}_{epaule}$ | -5.0 | 5.0 |

TABLE 4.5 – Bornes des 17 positions et vitesse articulaires initiales à optimiser.

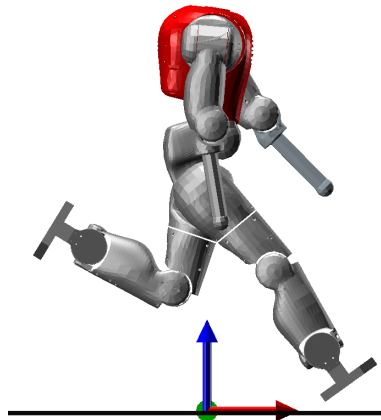


FIGURE 4.10 – Visualisation de l'état initial du robot pour un résultat de course avec une vitesse initiale d'avance de 1.6 m/s.

4.7 Fonctions d'objectif

Afin de quantifier la démarche produite et guider l'optimisateur vers le meilleur ensemble de paramètres, une fonction d'objectif est utilisée. Celle-ci est constituée de plusieurs tâches qui s'évaluent en parallèle. Le but de la phase d'optimisation est de trouver les paramètres qui maximisent cette fonction. La fonction d'objectif est détaillée dans la présente section. Étant donné que le modèle énergétique est une des tâches de cette fonction, celui-ci est également introduit.

4.7.1 Tâches de la fonction d'objectif

La fonction d'objectif est la somme de différentes tâches T_i . Elles sont inspirées des fonctions d'objectif de [5] et de [70]. La fonction telle que présentée dans cette section a permis d'obtenir des résultats de course. Voici les six tâches et leur pondération :

1. Locomotion durant une distance minimale (valeur maximale : 500) ;
2. Démarche sans tomber (valeur maximale : 500) ;
3. Vitesse d'avance (valeur maximale : 300) ;
4. Position verticale du haut du corps (valeur maximale : 50) ;
5. Minimisation de la phase d'appui (valeur maximale : 25) ;
6. Minimisation de l'énergie consommée (valeur maximale : 50).

Les deux premières tâches imposent au robot d'avancer et de ne pas tomber durant le temps de la simulation. Leur pondération est importante (valeur de 1000 au total) pour s'assurer que le robot avance sans tomber. La première tâche pénalise le robot s'il se déplace sur une distance inférieure 3.75 mètres (pour 15 secondes de simulations, soit une vitesse minimale de 0.25m/s si le robot ne tombe pas) :

$$T_{dist} = \min \left(500, 500 \frac{x}{0.25t_{final}} \right) \quad (4.48)$$

où x est la distance parcourue par le robot, et t_{final} est le temps total de la simulation. Si cette tâche n'est pas incluse dans la fonction d'objectif, l'optimisateur a tendance à trouver une solution où le robot est en équilibre statique après quelques pas. La deuxième tâche a pour but d'obtenir une démarche sans tomber durant tout le temps de simulation :

$$T_{chute} = t \frac{500}{t_{final}} \quad (4.49)$$

où t est le temps actuel de simulation (si le robot chute, la simulation s'arrête et la fonction renvoyée n'est donc pas maximale).

La troisième tâche impose une vitesse d'avance de référence à atteindre v_{ref} . Une fonction gaussienne est utilisée pour imposer cette vitesse de référence :

$$T_{vit} = \begin{cases} 300 & \text{si } |v_{ref} - v| \leq 0.05 \\ 300 \exp(-10(v_{ref} - v)^2) & \text{si } |v_{ref} - v| > 0.05 \end{cases} \quad (4.50)$$

$$(4.51)$$

où v est la vitesse moyenne d'avance du robot. Cette tâche n'est pas indispensable pour trouver des résultats de course, mais elle permet de favoriser la course en imposant une vitesse de référence assez élevée. La fonction est représentée à la figure 4.11.

Toutes les tâches suivantes utilisent également une fonction gaussienne, qui permet de pénaliser fortement toute solution qui s'écarte d'une valeur souhaitée. Les paramètres des fonctions ont été déterminés afin d'avoir le maximum pour la valeur recherchée (qui n'est pas toujours atteignable ; l'énergie consommée ne peut pas être nulle par exemple), et d'avoir une valeur non nulle pour des mauvaises solutions afin qu'elles convergent vers la solution voulue.

Afin d'avoir l'orientation du haut du corps plus ou moins verticale, la quatrième tâche est définie (au moyen d'une fonction gaussienne, voir figure 4.12) :

$$T_{torse} = \begin{cases} 50 & \text{si } |\theta_{torse}| \leq 0.1 \\ 50 \exp(-20\theta_{torse}^2) & \text{si } |\theta_{torse}| > 0.1 \end{cases} \quad (4.52)$$

où θ_{torse} est la position angulaire du torse.

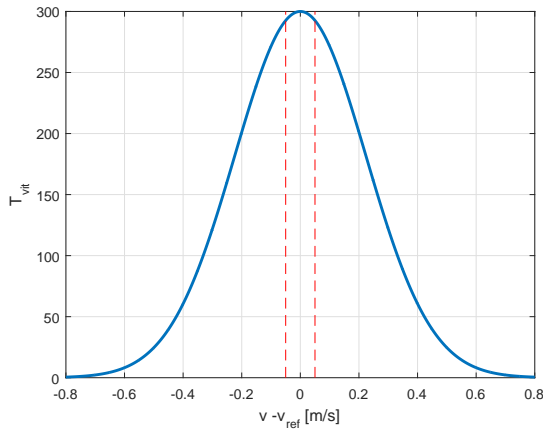


FIGURE 4.11 – Fonction gaussienne associée à la tâche qui favorise la vitesse d'avance du robot à v_{ref} .

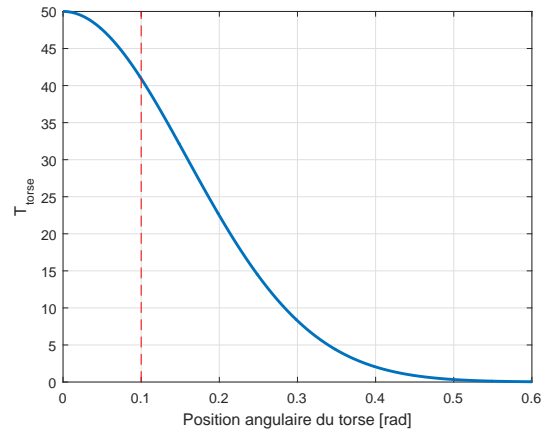


FIGURE 4.12 – Fonction gaussienne associée à la tâche qui favorise l'orientation verticale du haut du corps.

La cinquième tâche minimise le temps de la phase d'appui lors d'un cycle complet. Cette tâche est très importante pour obtenir des résultats de course. En effet, lors de la course, la phase d'appui représente moins de 50% d'un cycle. La tâche est définie au moyen d'une fonction gaussienne :

$$T_{appui} = 25 \exp(-7(appui_d + appui_g - 0.6)^2) \quad (4.54)$$

où $appui_d$ (et $appui_g$) représente la proportion du temps du cycle pendant lequel le pied droit (et gauche) est en phase d'appui. La fonction est maximale pour une phase d'appui qui représente en moyenne 30% du cycle. Le graphe de la fonction gaussienne est représenté à la figure 4.13.

Enfin, lors de la sixième tâche, l'énergie métabolique consommée pendant la locomotion est minimisée. Cela s'effectue au moyen d'une fonction gaussienne (figure 4.14) :

$$T_{energie} = 50 \exp(-0.01E^2) \quad (4.55)$$

où E est l'énergie métabolique consommée par unité de distance parcourue et normée par sa masse. Le modèle utilisé pour quantifier cette énergie est décrit dans la section suivante.

4.7.2 Modèle de l'énergie métabolique consommée

Afin de suivre une démarche bio-inspirée, l'énergie métabolique consommée par les muscles virtuels est minimisée. Cette énergie est constituée d'un travail mécanique produit par les muscles

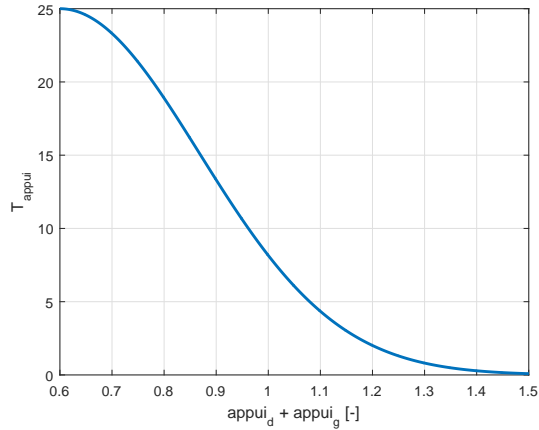


FIGURE 4.13 – Fonction gaussienne associée à la tâche qui minimise la phase d'appui.

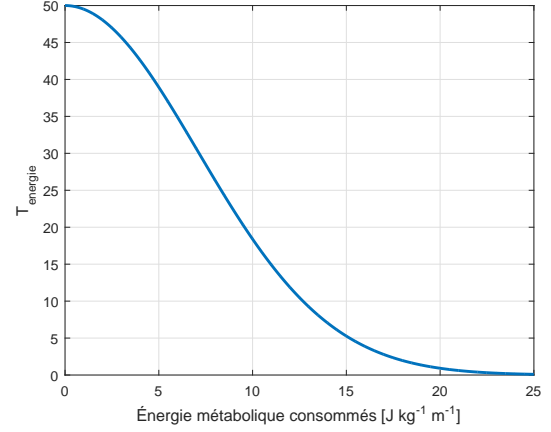


FIGURE 4.14 – Fonction gaussienne associée à la tâche qui minimise l'énergie métabolique consommée.

et de la chaleur relâchée par ceux-ci lors de la contraction. Le modèle utilisé pour quantifier l'énergie métabolique est celui proposé dans [7], et utilisé et implémenté dans [5]. L'explication qui suit est basée sur [5].

Selon la première loi de la thermodynamique, la variation de dépense d'énergie métabolique \dot{E} pour une unité muscle-tendon (MTU) donné est composé du taux de chaleur libérée (\dot{H}) et de la vitesse à laquelle le travail est effectué (\dot{W}) :

$$\dot{E} = \dot{H} + \dot{W} \quad (4.56)$$

$$= (\dot{A} + \dot{M} + \dot{S} + \dot{B}) + \dot{W} \quad (4.57)$$

où

\dot{A} est le taux d'énergie convertie en chaleur par le MTU pour un niveau de stimulation S_m donné. Ce taux est une fonction de la masse du muscle m_{MTU} et de sa stimulation.

\dot{M} modélise le taux de chaleur pour maintenir une contraction musculaire à un certain niveau. Il dépend de la masse du muscle m_{MTU} , de son niveau d'activation m et de sa longueur de fibre l_{ce} .

\dot{S} modélise la chaleur générée par le raccourcissement du muscle. Il est proportionnel à la force générée par le muscle (F_m) et à sa vitesse de contraction v_{ce} .

\dot{B} est le métabolisme de base. Il est estimé à $0.0225 \cdot m_{mtu}$.

\dot{W} est la puissance mécanique produite par le MTU pendant la contraction. Il dépend de la vitesse et de la force de l'élément contractile (v_{CE} et F_{CE}).

4.8 Conclusion

Le contrôleur de locomotion bio-inspiré utilisé dans le cadre de ce mémoire emploie des muscles virtuels qu'il faut stimuler afin que ceux-ci fournissent un couple de référence à l'articulation à laquelle/auxquels ils sont connectés. Le modèle musculaire des jambes a déjà été utilisé sur le

COMAN. Nous y avons ajouté un muscle, le *rectus femoris*. Une contribution de ce mémoire est également l'ajout du modèle musculaire complet du haut du corps. Pour modéliser les muscles à partir de leurs caractéristiques physiques, le modèle de Hill est utilisé.

Les différentes méthodes de contrôle des articulations du COMAN sont résumées au tableau 4.6. Étant donné qu'on s'intéresse uniquement à la démarche en deux dimensions (le robot ne peut pas tomber latéralement), seules les articulations sagittales sont contrôlées à partir du modèle musculaire (excepté pour le torse). Les articulations non sagittales sont contrôlées au moyen de références de positions nulles ou constantes (pour les bras).

L'utilité du mouvement du haut du corps est limitée pour la locomotion en deux dimensions (cela permet surtout de garder l'équilibre latéral). Néanmoins, ce mouvement est tout de même bénéfique d'un point de vue énergétique chez l'être humain, et il est donc modélisé sur le robot. Pour tirer profit des avantages du modèle musculaire (dynamique semblable à celle des êtres humains, couples de références plus lisses et réalistes grâce aux propriétés visco-élastiques des muscles), les articulations sagittales des bras sont contrôlées en imposant des couples de références produits par les stimulations envoyées aux muscles. Ces stimulations sont simplement proportionnelles à l'erreur de position par rapport à la référence imposée. Celle-ci est constante pour le coude, et proportionnelle à la différence d'angle au niveau des hanches pour l'épaule.

| Corps | Articulations | Contrôles | Références/Stimulations |
|---------|---------------|-----------------------------|--|
| Jambes | Sagittales | Couples (modèle musculaire) | Réflexes musculaires |
| | Latérales | Positions | 0.0 rad |
| | Transversales | Positions | 0.0 rad |
| Épaules | Sagittales | Couples (modèle musculaire) | $\propto K_p(\theta_{hanche,sag}^g - \theta_{hanche,sag}^d)$ |
| | Latérales | Positions | ± 0.09 rad |
| | Transversales | Positions | ± 0.14 rad |
| Coudes | Sagittales | Couples (modèle musculaire) | $K_p(0.25 - \theta_{coude})$ |
| Torse | Sagittale | Position | 0.0 rad |
| | Latérale | Position | 0.0 rad |
| | Transversale | Position | 0.0 rad |

TABLE 4.6 – Récapitulatif du contrôle des articulations des différentes parties du robot.

Les jambes sont contrôlées au moyen de lois de réflexes bio-inspirées basées sur les propriétés physiques des muscles. L'activation de chacun des muscles est basée sur une combinaison de trois lois de contrôle. La rétroaction positive de force génère une stimulation proportionnelle à la force produite par le muscle. La rétroaction positive de longueur de fibre musculaire génère une stimulation proportionnelle à l'étirement de sa longueur de fibre. Enfin, le contrôle PD de la position articulaire stimule le muscle pour imposer une position articulaire. Les lois de contrôle sont différentes en fonction de l'état de la jambe concernée : phase d'appui, phase oscillante, initiation à la phase oscillante et préparation à la phase d'appui. Ces deux derniers états ont été ajoutés par rapport aux modèles proposés dans [29] pour pouvoir obtenir des résultats de course.

Enfin, la position initiale du robot ainsi que la fonction d'objectif doivent être judicieusement déterminées pour obtenir le résultat souhaité (la course dans notre cas). Les positions et vitesses initiales du robot sont optimisées en favorisant un état initial en phase de vol. Nous n'avons pas trouvé de résultat de course à partir d'une position statique du robot. La fonction d'objectif doit être choisie pour que l'optimisateur favorise une démarche de course. Pour cela, on impose au robot de ne pas tomber, de suivre une vitesse de référence, d'avoir le haut du corps relativement droit, de minimiser le temps d'appui ainsi que l'énergie consommée. C'est avec cette configuration que des résultats de course ont été obtenus.

Chapitre 5

Résultats et analyses des simulations

Au chapitre 3, nous avons décrit le robot COMAN, sa modélisation ainsi que l’environnement de simulation. Ensuite, le chapitre 4 a été consacré à la description du modèle musculaire et du contrôleur bio-inspiré utilisés pour obtenir des résultats de course.

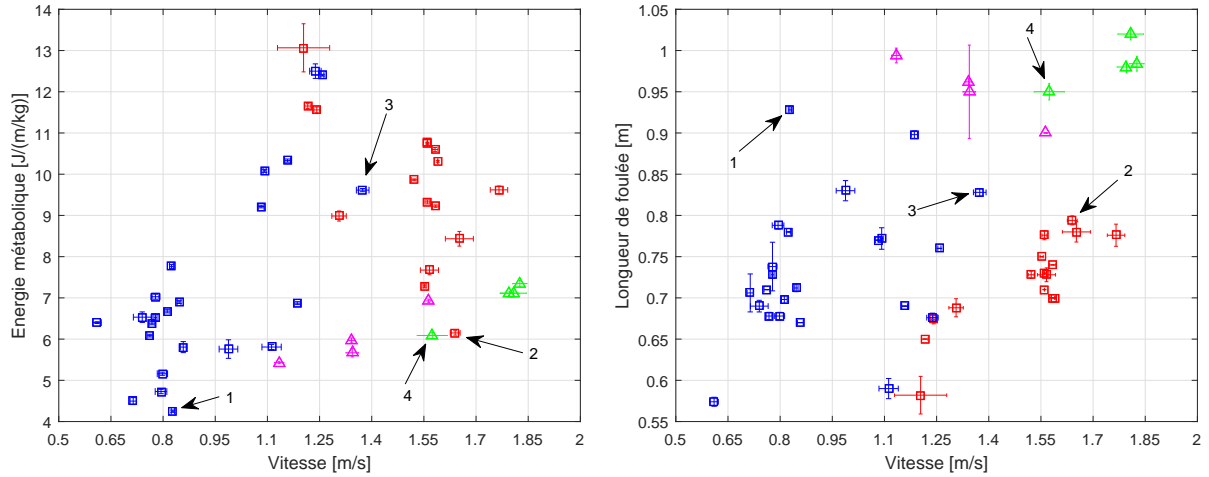
Dans ce cinquième chapitre, nous présentons et analysons les résultats des simulations. Nous exposons tout d’abord les caractéristiques principales des différentes démarches obtenues avec le contrôleur. Nous en sélectionnons ensuite quatre qui sont analysées en détail : une marche à vitesse optimale, deux courses avec des modèles de pieds différents, ainsi qu’une marche dite rapide. Lors de ces analyses, une attention particulière est apportée aux différences entre la course et la marche, ainsi qu’entre les résultats obtenus et ceux issus de données humaines.

5.1 Aperçu des résultats

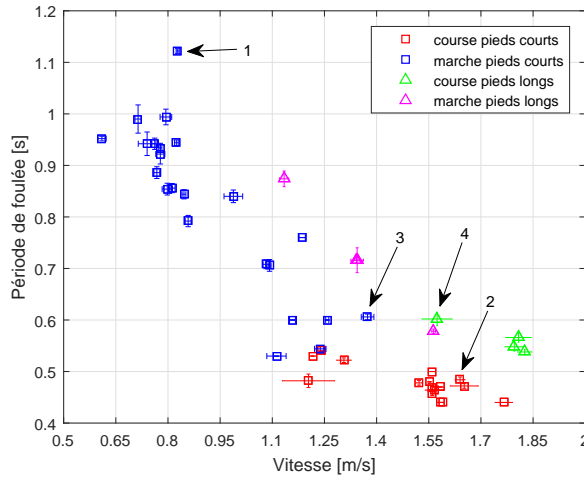
De nombreux résultats ont été obtenus suite aux phases d’optimisation afin de déterminer les valeurs des paramètres du contrôleur pour avoir la démarche voulue. Les caractéristiques principales de l’ensemble des résultats sont représentées à la figure 5.1. L’énergie métabolique consommée par unité de distance, la longueur d’une foulée (distance parcourue pendant un cycle) ainsi que sa période y sont représentées en fonction de la vitesse d’avance (figures 5.1a, 5.1b et 5.1c).

Ces graphes montrent la moyenne de cinq exécutions du programme pour chaque contrôleur, et les barres verticales et horizontales représentent l’écart type (chaque exécution du programme ne fournit pas le même résultat à cause du bruit aléatoire ajouté à l’erreur de couple ainsi qu’aux stimulations). Les carrés bleu et rouge représentent les résultats de marche et de course (respectivement) avec le modèle de pieds courts avec orteils (figure 3.3c), et les triangles cyan et vert représentent les résultats de marche et de course (respectivement) avec le modèle de pieds longs avec orteils (figure 3.3b).

Pour choisir les résultats que nous allons analyser en profondeur dans la suite de ce chapitre, nous utilisons comme critère de sélection l’énergie métabolique consommée. La minimisation de celle-ci fournit une démarche plus proche de celle d’un être humain [3]. Ces résultats sont mis en valeur au moyen de flèches sur les figures 5.1. Les résultats de marche (1) et de course (2) avec le modèle de pieds courts avec orteils seront analysés aux sections 5.2 et 5.3. La course avec le modèle de pieds longs avec orteils (4) sera analysée à la section 5.5. Enfin un résultat de marche rapide (3) sera également détaillé à la section 5.4. Le tableau 5.1 résume les caractéristiques moyennes de chaque démarche sélectionnée.



(a) Évolution de l'énergie métabolique consommée par unité de distance parcourue en fonction de la vitesse d'avance du robot. (b) Évolution de la longueur d'une foulée en fonction de la vitesse d'avance du robot.



(c) Évolution de la période d'une foulée en fonction de la vitesse d'avance du robot.

FIGURE 5.1 – Aperçu des principales caractéristiques des différents résultats obtenus. Les barres horizontales et verticales représentent l'écart type obtenu en exécutant cinq fois le même contrôleur. Les carrés sont des résultats en utilisant le modèle de pieds courts avec orteils, et les triangles utilisent le modèle de pieds longs avec orteils. Tant les résultats de marche (en bleu et cyan) que de course (en rouge et vert) sont représentés sur les figures. Les flèches pointent les résultats qui sont développés dans la suite de ce chapitre : marche, marche rapide et course avec le modèle de pieds courts avec orteils (1, 2 et 3 respectivement), et course avec le modèle de pieds longs avec orteils (4).

5.1.1 Évolution des caractéristiques avec la vitesse d'avance

Afin de comparer les résultats obtenus avec ceux d'un homme moyen, il faut recourir à une mise à échelle des paramètres. Si on suppose que l'homme moyen possède des jambes de longueur maximale de 100 cm pour une masse de 80 kg, les différents facteurs de mise à échelle suivants sont trouvés (voir section 4.2.4) :

- La vitesse du COMAN est multipliée par la racine carrée du rapport de longueur des jambes, soit 1.53 ;
- La période d'une foulée est multipliée par la racine carrée du rapport de longueur des

jambes, soit 1.53;

- La longueur d'une foulée est multipliée par le rapport de longueur des jambes, soit 2.34.

| | énergie | | | | section |
|-----------------------|------------------|---------------------------|---------------------------|--------------------------|---------|
| | vitesse [m/s] | métabolique [J/(kg*m)] | longueur de foulée [m] | période de foulée [s] | |
| 1. marche normale | 0.83 | 4.25 | 0.93 | 1.12 | 5.2 |
| 2. course | 1.64 | 6.15 | 0.79 | 0.48 | 5.3 |
| 3. marche rapide | 1.37 | 9.61 | 0.83 | 0.61 | 5.4 |
| 4. course pieds longs | 1.57 | 6.09 | 0.95 | 0.60 | 5.5 |

TABLE 5.1 – Caractéristiques principales de chacune des démarches analysées (pointées à la figure 5.1) dans la suite du chapitre aux sections mentionnées. Les valeurs sont la moyennée sur cinq exécutions de chaque contrôleur.

Tout d'abord, on remarque à la figure 5.1 que la transition de la marche vers la course s'effectue à une vitesse d'environ 1.25 m/s pour le COMAN, soit 1.92 m/s en mettant à échelle pour un homme moyen. Cela correspond bien à la vitesse de transition pour l'être humain qui se situe à environs 1.96 m/s [31].

On remarque sur la figure 5.1a que l'énergie consommée par unité de distance parcourue est minimale pour la marche à une vitesse de 0.8 m/s. Il s'agit de la vitesse naturelle de marche du COMAN, c'est-à-dire la vitesse à laquelle le robot consomme le moins d'énergie. Cette vitesse naturelle pour le COMAN est également obtenue dans [5] et [18]. Des simulations ont été lancées sans contraindre la vitesse d'avance afin de déterminer la vitesse de course naturelle, mais les résultats ne convergent pas vers une vitesse spécifique. Pour la marche, le coût énergétique de la démarche augmente largement pour des vitesses supérieures à la vitesse naturelle du robot. La course à environ 1.6 m/s est plus efficace d'un point de vue de l'énergie consommée par unité de distance que la marche supérieure à 1,2 m/s. Cela démontre l'utilité de la course. Cette observation est également constatée chez l'homme, comme représenté à la figure 5.2. Cette courbe permet également d'expliquer pourquoi les optimisations ne convergent pas vers une vitesse naturelle de course, étant donné que l'énergie consommée par distance parcourue ne présente pas de minimum significatif (contrairement à la marche). Notons également que le modèle de pieds longs avec orteils consomme globalement moins d'énergie par unité de distance que le modèle court, et ce à vitesse égale. La comparaison de ces deux modèles est effectuée à la section 5.5.

La distance parcourue pendant un cycle augmente globalement avec la vitesse d'avance de la marche (figure 5.1b). Une vitesse de marche plus élevée est donc atteinte en faisant des pas plus grands. Cela est également montré sur le COMAN dans [67]. L'évolution de cette distance en fonction de la vitesse de course n'est pas claire, d'autant plus que peu de résultats à vitesses différentes ont été trouvés.¹ On remarque cependant que la longueur de foulée lors de la course n'est pas significativement plus élevée que lors de la marche. Il s'agit d'une différence avec l'être humain où cette distance augmente jusqu'à saturer à partir d'une certaine vitesse (environ 4m à partir de 7m/s) [10]. Cette différence peut s'expliquer par la démarche de course du COMAN qui garde les genoux trop fléchis (voir section 5.3). On remarque également que le modèle de pieds longs avec orteils a tendance à fournir des résultats avec des plus grandes longueurs de foulée qu'avec le modèle de pieds courts.

1. Le nombre de résultats est limité par les phases d'optimisation qui sont longues et demandent beaucoup de ressources

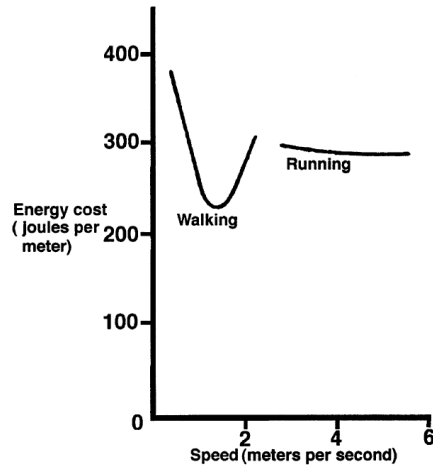


FIGURE 5.2 – Évolution de l'énergie métabolique consommée en fonction de la vitesse d'avance chez l'être humain. Image issue de [47].

Enfin, l'évolution de la période d'une foulée (durée d'un cycle) diminue lorsque la vitesse d'avance augmente. Cela est nettement visible à la figure 5.1c. L'élévation de la vitesse lors de la marche est donc générée non seulement par l'augmentation de distance parcourue lors d'un cycle, mais également par la diminution de la durée de celui-ci. Cela a encore une fois déjà été montré sur le COMAN dans [67]. La période de foulée continue à diminuer pour la course, ce qui est également le cas chez l'être humain [10].

5.1.2 Symétrie de la locomotion

Par souci de clarté, seule la jambe droite est utilisée pour analyser certaines caractéristiques de la démarche dans la suite de ce chapitre. La figure 5.3 montre que les positions articulaires sagittales de la jambe droite sont symétriques à ceux de la jambe gauche. Il s'agit des moyennes des positions sur 10 cycles pour la course sélectionnée précédemment (numéro 2 sur la figure 5.1). En toute généralité, cette symétrie intervient dans tous les résultats obtenus, étant donné que les lois de contrôle utilisées sont identiques pour les deux jambes (en fonction de la phase dans laquelle elles se situent).

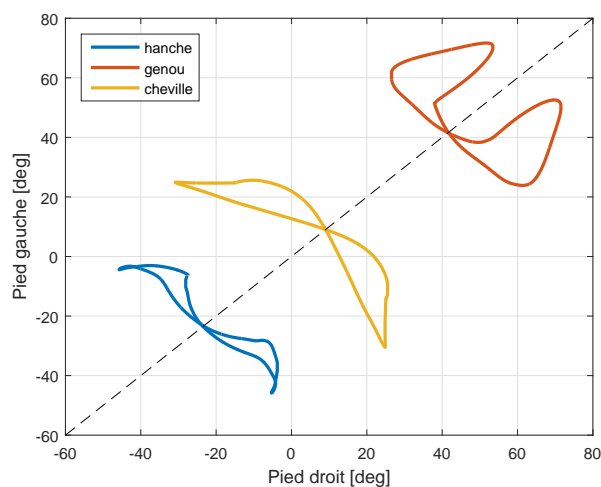


FIGURE 5.3 – Positions des articulations sagittales de la jambe gauche en fonction de ceux de la jambe droite. Les données sont moyennées sur 10 cycles.

5.2 Analyse de la marche

Afin de comparer la course avec la marche, cette dernière est décrite dans la section suivante. La marche analysée est celle représentée par le numéro 1 sur la figure 5.1, et dont les caractéristiques principales sont reprises ci-dessous.

| | énergie | | | |
|-------------------|------------------|---------------------------|---------------------------|--------------------------|
| | vitesse [m/s] | métabolique [J/(kg*m)] | longueur de foulée [m] | période de foulée [s] |
| 1. marche normale | 0.83 | 4.25 | 0.93 | 1.12 |

TABLE 5.2 – Caractéristiques principales de la marche dite normale. Les valeurs sont la moyennée sur cinq exécutions de chaque contrôleur.

La figure 5.4 montre les différentes phases du cycle de la marche sélectionnée à la section 5.1. Visuellement, la démarche ressemble à celle d'un être humain présentée à la section 1.3.2. La transition de la phase d'appui vers la phase oscillante s'effectue autour de 60% du cycle (figure 5.4e). Les deux phases de double appui sont également présentes (figures 5.4a à 5.4b, et 5.4d à 5.4e). Une comparaison des positions articulaires obtenues avec des données humaines est effectuée dans le point suivant pour déceler les éventuelles différences.

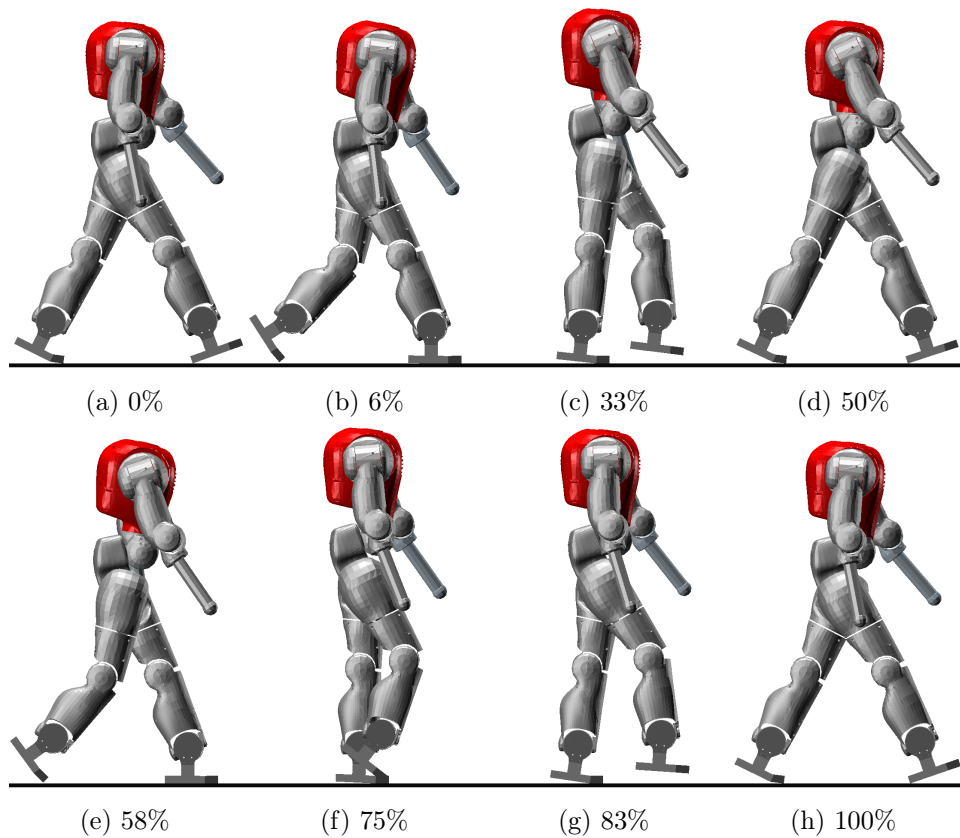


FIGURE 5.4 – Instantanés d'un cycle de marche du COMAN à 0.83 m/s dans l'environnement de simulation. Les légendes indiquent le pourcentage du cycle.

5.2.1 Positions et couples des articulations sagittales des jambes

La figure 5.5 représente les positions, couples et puissances mécaniques des articulations sagittales de la jambe droite du COMAN. Les positions et couples articulaires sont comparés avec des données humaines de marche à 1.25 m/s issues de [70] (aires grisées). Une mise à échelle a été effectuée pour comparer les données de couples articulaires (via le rapport de la taille des jambes et de la masse des robots, voir D.5).

Les positions de la hanche, du genou et de la cheville du COMAN sont très similaires aux données humaines. Cependant, le genou reste un peu trop droit durant la période de mise en charge, juste après l'impact du talon du pied droit (début du cycle). De plus, le genou n'est pas assez fléchi lors de la phase oscillante.

Les couples articulaires du COMAN sont également semblables aux données humaines, excepté pour le genou au début du cycle. Le robot génère un couple afin de fléchir le genou pour éviter une hyper extension de celui-ci, tandis que pour l'être humain il a pour but de compenser la flexion du genou après l'impact du talon. De plus, comme pour l'être humain, deux pics de couples sont présents à 15% et 50% du cycle. Le premier est produit par la hanche de la jambe en appui pour bouger le corps en avant. Le second est produit par la cheville pour propulser le robot en avant.

La puissance mécanique \dot{W} développée à chaque articulation est également représentée à la figure 5.5. Il s'agit de la somme de la puissance mécanique produite par chaque MTU lié à l'articulation pendant la contraction de celui-ci ($v_{CE} * F_{CE}$). Les courbes de puissance sont similaires aux couples appliqués aux articulations.

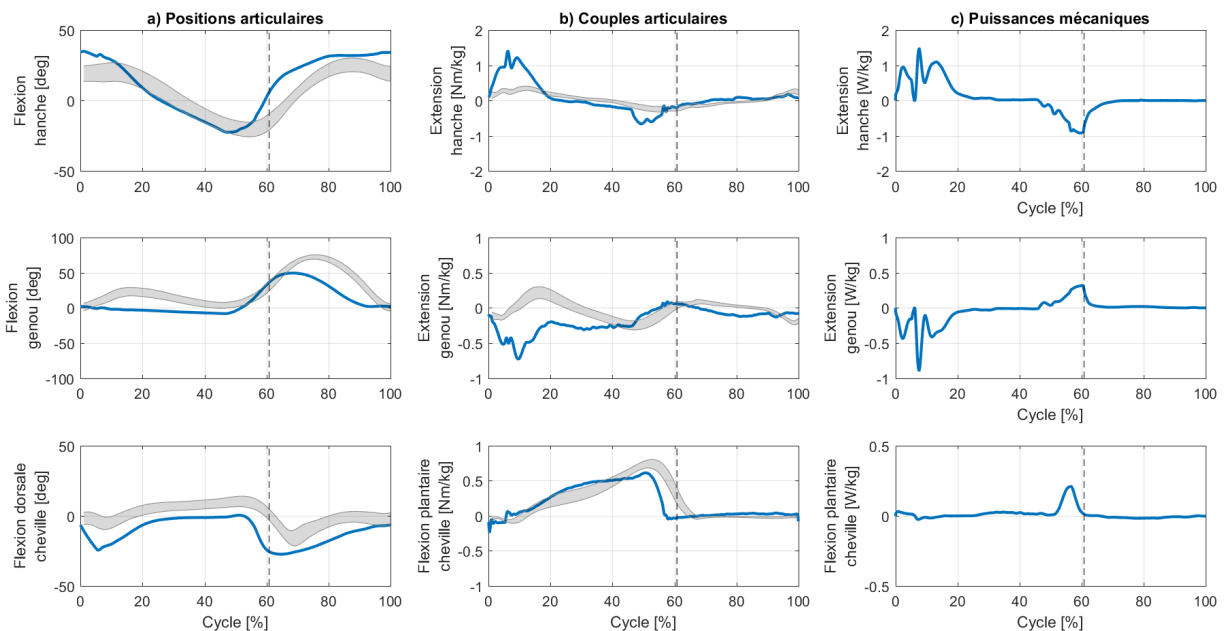


FIGURE 5.5 – Positions, couples et puissances mécaniques des articulations sagittales de la jambe droite du COMAN pendant un cycle de marche. Les aires grisées représentent l'écart type issu de données humaines à 1.25 m/s [70]. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

5.2.2 Forces de réaction du sol

La figure 5.6 montre l'évolution des forces de réaction du sol agissant sous les pieds du robot pendant un cycle de marche. Le pied en phase d'appui (pied droit, en bleu sur la figure) supporte tout le poids du robot jusqu'à l'impact du talon du pied gauche (en rouge sur la figure) à 50% du cycle. Son poids est ensuite transféré sur son pied gauche. On peut y observer deux pics lors de la phase d'appui : le premier lors du contact du talon avec le sol, et le second lors de la propulsion. Entre ces deux pics, la force chute, car le pied est à plat sur le sol.

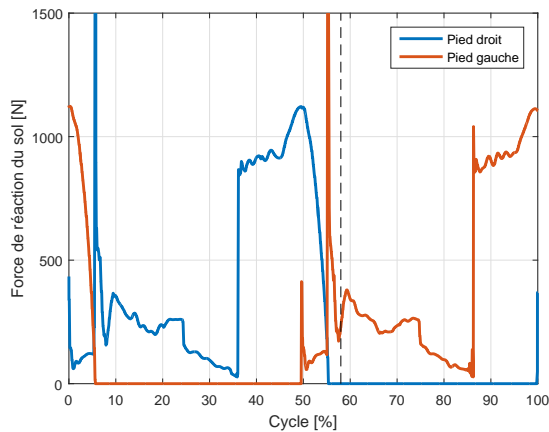


FIGURE 5.6 – Forces verticales de réaction du sol agissant sous le pied droit (trait bleu) et gauche (trait rouge) du robot pendant un cycle de marche.

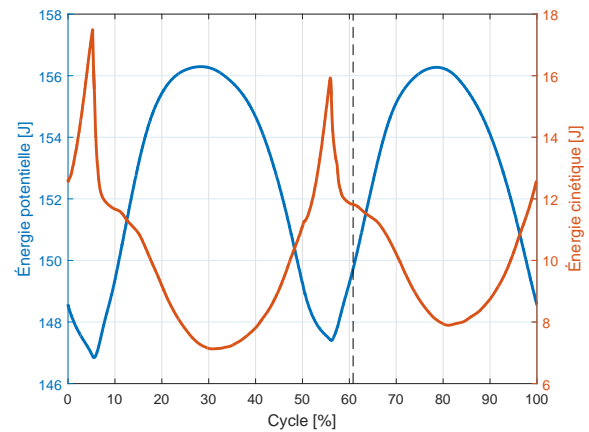


FIGURE 5.7 – Évolution de l'énergie cinétique et potentielle du centre de masse du robot pendant un cycle de marche. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

5.2.3 Énergie mécanique du centre de masse

L'évolution de l'énergie cinétique et potentielle du centre de masse du robot pendant le cycle est représentée à la figure 5.7. Pendant la marche, le robot bascule autour de la jambe d'appui qui est relativement droite (figure 5.4c et 5.4g). Le centre de masse atteint donc la position la plus haute (par rapport au sol) au milieu de la phase d'appui. L'énergie potentielle y est maximale.

L'énergie cinétique du centre de masse atteint sa valeur minimale au milieu de la phase d'appui, car les forces horizontales de réaction du sol tendent à décélérer le corps durant la première moitié de la phase d'appui. Durant la deuxième moitié, le corps accélère. Un pic est présent pour l'énergie cinétique en début de phase d'appui, car la vitesse horizontale du centre de masse augmente rapidement lorsque le robot bascule autour du talon en contact avec le sol.

Il en résulte que l'énergie cinétique et l'énergie potentielle du centre de masse sont déphasées d'environ 180 degrés lors de la marche (figure 5.7). Pendant la première moitié de la phase d'appui, l'énergie cinétique peut donc être convertie en énergie potentielle, et inversement lors de la seconde moitié.

5.2.4 Énergie métabolique

La puissance métabolique consommée par chacun des muscles lors d'un cycle de marche est représentée à la figure 5.8a. Pour rappel, le modèle de l'énergie métabolique prend en compte la chaleur H produite par les muscles ainsi que leur travail mécanique W , à savoir $\dot{E} = \dot{H} + \dot{W} =$

$(\dot{A} + \dot{M} + \dot{S} + \dot{B}) + \dot{W}$ (voir section 4.7.2). La puissance mécanique développée par chacun des muscles est également représentée à la figure 5.8b.

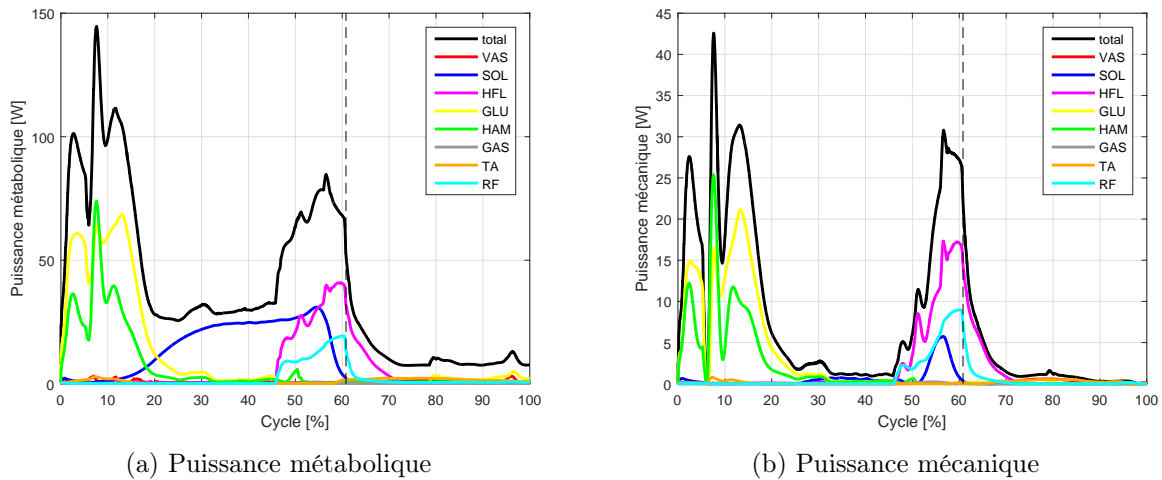


FIGURE 5.8 – Puissance métabolique et puissance mécanique consommées par les muscles virtuels pendant un cycle de marche. La ligne verticale représente la transition phase d’appui/phase oscillante. Les données sont moyennées sur 10 cycles.

Sur chacune des figures, nous observons deux pics à 10% et 55% du cycle. Le premier pic est présent après le contact du talon avec le sol. Les principales contributions de celui-ci proviennent des muscles de la hanche GLU et HAM qui maintiennent le torse dans une position stable après le contact avec le sol. Le deuxième pic est produit par les muscles HFL et RF et dans une moindre mesure SOL pour permettre la propulsion de la jambe de la phase d’appui vers la phase oscillante.

La principale contribution de la puissance métabolique entre ces deux pics provient de l’activation du muscle SOL. En effet, ce muscle est fortement activé durant la phase d’appui n’engendrant un changement de position de la cheville qu’en fin d’appui lorsque le pied décolle du sol.

5.3 Analyse de la course

La figure 5.9 montre les différentes étapes de la course sélectionnée à la section 5.1, et dont les caractéristiques principales sont reprises ci-dessous.

| | énergie | | | |
|-----------|------------------|---------------------------|---------------------------|--------------------------|
| | vitesse [m/s] | métabolique [J/(kg*m)] | longueur de foulée [m] | période de foulée [s] |
| 2. course | 1.64 | 6.15 | 0.79 | 0.48 |

TABLE 5.3 – Caractéristiques principales de la course. Les valeurs sont la moyennée sur cinq exécutions de chaque contrôleur.

Visuellement, il s’agit bien d’une démarche de course, les deux phases de vol étant clairement visibles (figures 5.9d et 5.9h). Une première comparaison visuelle avec le cycle de course humaine présenté à la section 1.3.3 fournit un bon aperçu des principales différences entre la course du COMAN et celle d’un être humain : les pas effectués sont plus petits, le genou est trop fléchi avant le contact avec le sol, et le pied est trop à plat lors de ce contact (le contact initial est généralement effectué avec l’arrière du pied, mais ce n’est pas le cas de tous les coureurs [47]).

Une analyse plus rigoureuse des différences avec les données humaines ainsi que la caractérisation de la course par rapport à la marche est effectuée dans les sous-sections suivantes.

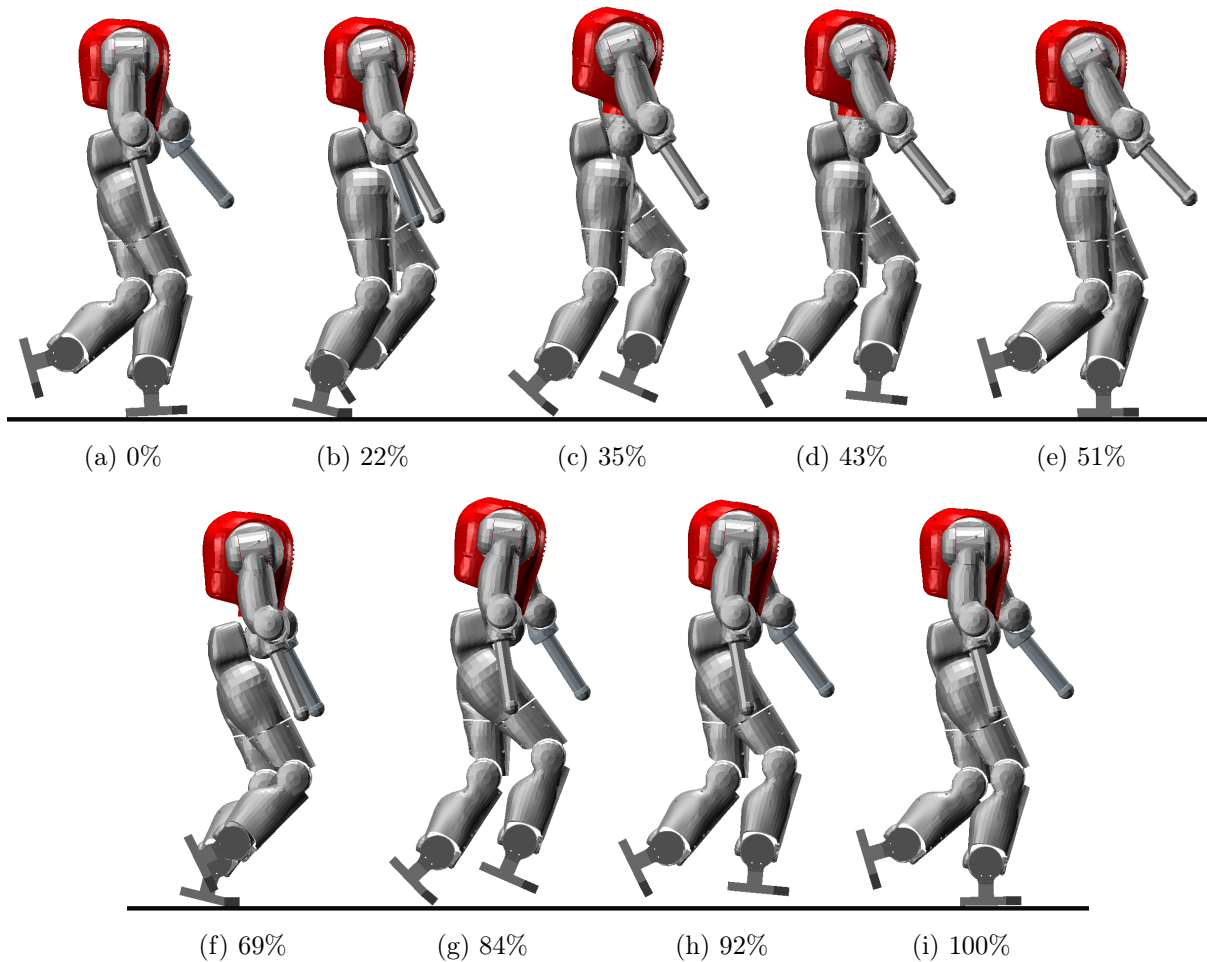


FIGURE 5.9 – Instantanés d’un cycle de course du COMAN à 1.64 m/s dans l’environnement de simulation. Les légendes indiquent le pourcentage du cycle.

5.3.1 Positions et couples des articulations sagittales des jambes

La figure 5.10 représente les positions, couples et puissances mécaniques des articulations sagittales de la jambe droite du COMAN. Les positions et couples articulaires sont comparés avec des données de courses humaines à 2 m/s issues de [70] (aires grisées). Afin de rendre la comparaison avec la marche plus facile, les données humaines de la marche à 1.25 m/s sont représentées en pointillés rouges (n’ayant pas ces données pour la puissance mécanique, les résultats de la marche de la section précédente sont utilisés à leur place).

Le mouvement des articulations ne suit pas aussi bien les données humaines que lors de la marche, mais les caractéristiques principales des courbes sont présentes. Une première différence est que les extensions de la hanche et de la jambe atteignent leur maximum plus tôt que les données humaines. De plus, comme pour la marche, la flexion maximale du genou lors de la phase oscillante est trop faible. La plus grosse différence concerne l’extension du genou en fin de cycle qui n’est pas suffisant : la jambe n’est pas assez tendue avant le contact avec le sol (figure 5.9e et 5.9i). Cela a également pour conséquence que le pied est trop à plat lors du contact avec le sol, malgré une flexion dorsale plus importante que les données humaines.

Les couples articulaires peuvent expliquer ces différences cinématiques. Le couple d'extension du genou atteint son maximum plus tôt avec une amplitude supérieure aux données humaines. Cela est dû au fait que le genou est trop fléchi au début du cycle. Le couple du genou est également différent en fin de phase oscillante, ce qui provoque cette non-extension de celui-ci.

Les courbes de la puissance mécanique développée à chaque articulation sont similaires aux couples articulaires. La puissance au niveau du genou et de la cheville ont une amplitude beaucoup plus élevée que lors de la marche.

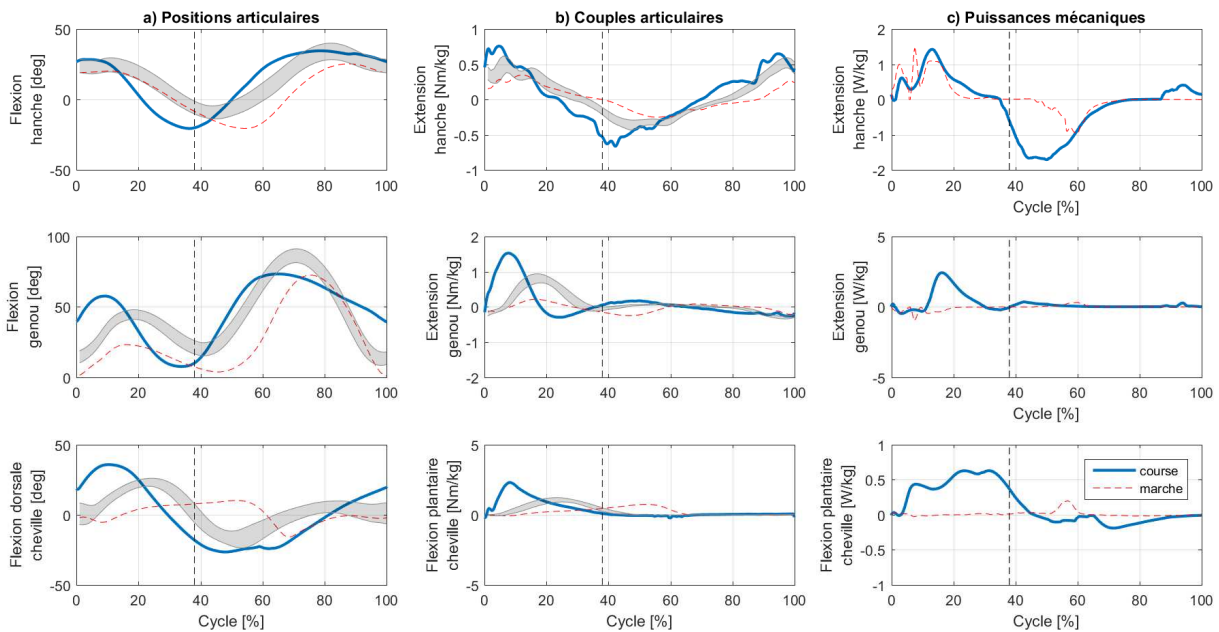


FIGURE 5.10 – Positions, couples et puissances mécaniques des articulations sagittales de la jambe droite du COMAN pendant un cycle de course. Les aires grisées représentent l'écart type issu de données humaines à 2.0 m/s [70]. Les traits en pointillés rouges représentent les données de marche humaine à 1.25 m/s. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

Différences cinématiques avec la marche

Le mouvement de la hanche est assez similaire pour la marche et la course. La flexion maximale de la hanche en phase oscillante est légèrement plus importante pour la course que pour la marche, et celle-ci augmente avec la vitesse avant d'effectuer de plus grands pas [47]. Une différence notable est l'extension de la hanche en fin de cycle de course, pas présente lors de la marche, afin de préparer le contact initial. Cette différence a pour but d'empêcher une décélération trop importante du corps lors du contact initial qui est accentué si le pied est trop loin du centre de masse du robot.

Les caractéristiques du mouvement du genou sont similaires pour la marche et la course, mais leurs amplitudes maximales sont différentes. Pendant la période d'amortissement de la phase d'appui, le genou se fléchit plus lors de la course que de la marche. La flexion du genou lors de la phase oscillante est également plus importante pour la course que la marche, et cette différence s'accroît avec l'augmentation de la vitesse [47].

Afin d’avoir un contact initial avec le talon, une plus grande flexion dorsale est requise pour la course que pour la marche, car le genou est plus fléchi. Au début du cycle de la marche, la cheville est en flexion plantaire, tandis qu’elle est en flexion dorsale pour la course afin de transférer le poids du corps sur le pied en appui lors de la période d’amortissement. Le reste du cycle est assez similaire pour la marche et la course (le timing est différent, car les phases n’ont pas les mêmes proportions).

5.3.2 Contributions des différents muscles

Les stimulations envoyées aux différents muscles virtuels du COMAN ainsi que la force produite par ceux-ci sont visibles aux figures 5.11a et 5.11b. Dans cette partie, on explique les contributions principales de chacun des muscles (lois qui sont fortement utilisées suite aux simulations, et celles qui ne le sont pas). Pour rappel, les différentes lois de contrôle sont précisées à la section 4.5. Afin de comparer les résultats de la course avec ceux de la marche, les stimulations et forces obtenues pour le résultat de marche sont représentées en traits pointillés à la figure 5.11.

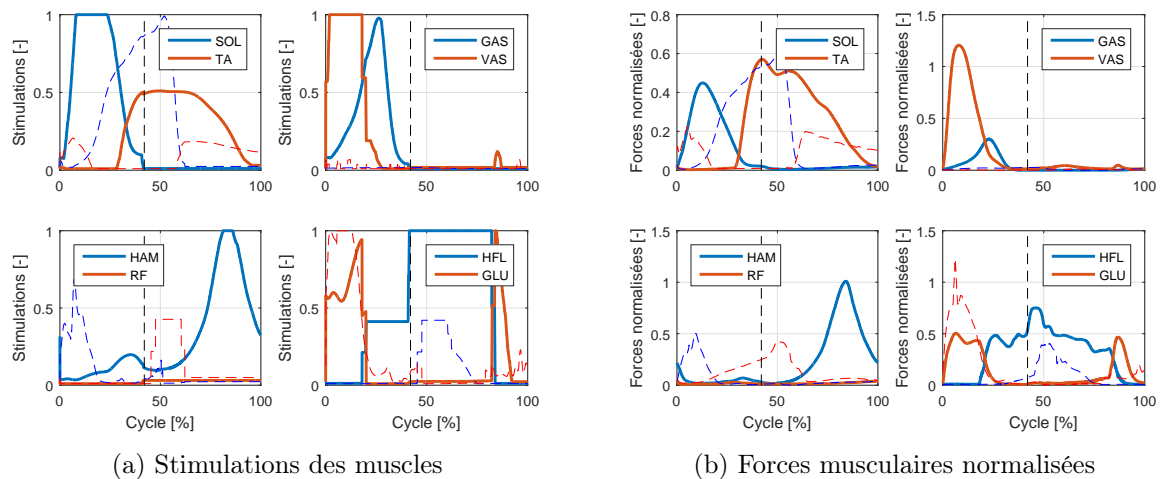


FIGURE 5.11 – Évolutions des stimulations envoyées aux muscles ainsi que des forces musculaires générées par celles-ci pendant un cycle de course. Les traits en pointillés représentent les résultats obtenus pour la marche présentée à la section 5.2. Les données sont moyennées sur 10 cycles.

Au niveau de la cheville, les muscles SOL et GAS sont fortement activés (au moyen de la rétroaction en force) lors de la phase d’appui pour permettre de propulser le robot. L’activation du muscle GAS est nécessaire pour propulser le robot lors de la course, alors qu’il n’est pas utilisé pour la marche. Le muscle TA (au moyen d’une rétroaction en longueur de fibre musculaire) est ensuite activé en phase oscillante pour redresse la cheville.

L’activation du muscle VAS (au moyen d’une rétroaction en force) en début de phase d’appui est également très utilisée, contrairement à la marche. Elle a pour but de créer une extension du genou lors de l’amortissement de la phase d’appui afin de garder une flexion du genou acceptable. La différence d’activation par rapport à la marche peut s’expliquer, car d’une part la vitesse d’impact avec le sol est plus élevée, et d’autre part car le genou est déjà fléchi lors de cet impact. L’activation est supprimée lors de l’initiation à la phase oscillante afin de préparer la flexion du genou. On remarque également que le muscle VAS est brièvement activé au début de la préparation à la phase d’appui (au moyen d’un contrôle PD), mais néanmoins pas assez pour permettre une bonne extension du genou. L’optimisateur ne fournit pas un gain assez élevé pour permettre d’avoir la jambe tendue en fin de cycle (alors que les bornes sont suffisantes pour y arriver).

Le muscle bi articulaire RF (excité uniquement au moyen de stimulations constantes) n'est pas utilisé lors de la course.

Au niveau de la hanche, les muscles GLU et HAM sont activés au début de la phase d'appui (au moyen d'un contrôle PD sur la position du torse) pour maintenir le torse droit. Le muscle HFL n'est pas activé, car le torse est trop penché en avant par rapport à sa référence. Lors de la préparation à la phase oscillante, l'activation du muscle GLU est annulée, et le muscle HFL est activé pour préparer la flexion de la hanche.

Pendant de la phase oscillante, le muscle HFL est activé au maximum pour permettre une flexion de la jambe. L'activation est générée principalement par la rétroaction positive en longueur de fibre de HFL. Le muscle HAM est activé (au moyen d'une rétroaction en force) pour permettre l'extension de la hanche en fin de cycle. La rétroaction en force n'est par contre pas du tout utilisée pour le muscle GLU. Lors de l'initiation à la phase d'appui, le muscle GLU est activé au moyen d'un contrôle PD pour permettre l'extension de la hanche afin de préparer l'impact avec le sol (le muscle HFL n'est pas activé, car la hanche est trop fléchie par rapport à la référence).

Comme cela est expliqué dans [56], l'apport de chaque muscle chez l'être humain pour d'une part accélérer le corps verticalement et d'autre part le propulser vers l'avant n'est pas très clair. Les principales contributions à l'accélération verticale du corps proviennent de l'extension de la hanche et du genou (GLU et VAS) ainsi que de la flexion plantaire de la cheville (muscle SOL). La propulsion en avant est issue principalement du muscle GLU du début du cycle jusqu'au milieu de la phase d'appui, et ensuite de la flexion plantaire (SOL et GAS) et du HAM du milieu de la phase d'appui jusqu'à la fin de celle-ci.

5.3.3 Comparaison avec la marche

Les différences de dynamique entre la marche et la course sont explicitées dans cette section. On y détaille les forces de contact du sol, l'énergie mécanique du centre de masse, l'énergie métabolique consommée et les contributions de chacun des muscles dans celle-ci.

Forces de réaction du sol

Les forces de réaction du sol sont nettement différentes en cas de marche ou en cas de course, dans la mesure où il n'y a pas la présence, dans la course, de phases de double appui, mais des phases de vol. Les forces de réaction verticales du sol pour la course sont représentées à la figure 5.12. Les phases de vol (pas de forces de réaction) sont bien visibles sur ce graphe. De plus, l'amplitude de ces forces est près de deux fois plus élevée pour la course que la marche.

Énergie mécanique du centre de masse

L'énergie mécanique du centre de masse du robot permet également de caractériser la course, dans le sens où elle est différente de celle de la marche. En effet, la jambe est fléchie durant la première moitié de la phase d'appui et s'étend durant la deuxième partie. Il en résulte que l'énergie potentielle du robot est minimale au milieu de la phase d'appui. L'évolution de l'énergie cinétique du centre de masse pendant le cycle est identique pour la course et la marche, car les forces de réaction du sol tendent à décélérer le corps durant la première partie de la phase d'appui, et à l'accélérer durant la seconde partie. Les énergies potentielle et cinétique sont donc en phase lors de la course, comme représenté à la figure 5.13.

Les pics de l'énergie cinétique sont causés par le contact avec le sol qui réduit fortement et rapidement la vitesse verticale du centre de masse. Le creux en fin de phase d'appui peut s'expliquer par la diminution de la vitesse verticale du centre de masse, tandis que la vitesse horizontale de

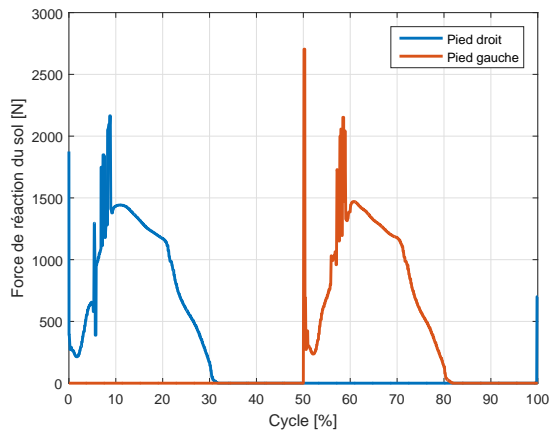


FIGURE 5.12 – Forces verticales de réaction du sol agissant sous le pied droit (trait bleu) et gauche (trait rouge) du robot pendant un cycle de course.

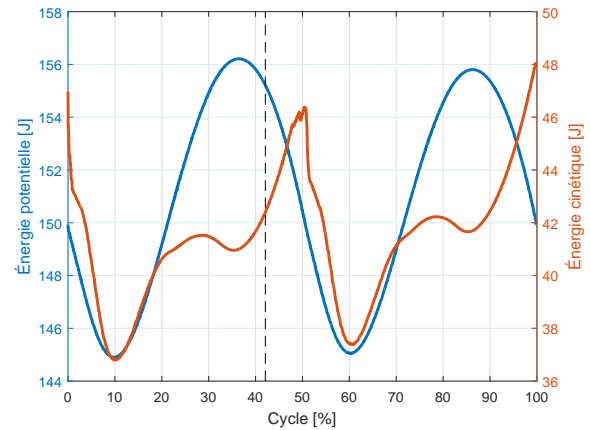


FIGURE 5.13 – Évolution des énergies cinétique et potentielle du centre de masse du robot pendant un cycle de course. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

celui-ci continue d'augmenter.

Comme les deux énergies sont en phase, la conversion d'énergie cinétique en énergie potentielle lors de la phase d'appui n'est donc pas fortement utilisée pour élever et accélérer le centre de masse du robot lors de la course, contrairement à la marche. Cependant cette énergie peut être stockée dans les tendons musculaires (ou physiquement dans la partie élastique des moteurs sur le robot) afin d'être utilisée plus tard dans le cycle.

Énergie métabolique

La puissance métabolique ainsi que la puissance mécanique consommée par les muscles virtuels pendant un cycle de course sont représentées à la figure 5.14. Elles possèdent un pic important au milieu de la phase d'appui. Les principales contributions sont les muscles VAS, GLU et SOL. Les muscles VAS et GLU sont principalement responsables du soutien de la jambe lors de l'amortissement et de l'accélération verticale du corps. Les muscles SOL et GAS sont responsables de la propulsion vers l'avant. Notons que le muscle GAS est peu représenté, car la force générée par celui-ci n'est pas très élevée par rapport aux autres muscles. De manière générale, les muscles de la chevilles sont également plus légers que ceux de la hanche, et contribuent donc moins dans la puissance métabolique (voir section 4.7.2).

Le second pic pendant la phase oscillante est généré principalement par le muscle HFL afin d'effectuer une flexion de la hanche. Ce pic est plus étalé que lors de la marche car la flexion du genou est plus importante lors de la course. Les muscles HAM et GLU sont à l'origine d'un troisième pic en fin de cycle afin de préparer le contact avec le sol.

Les différences entre l'allure de la puissance mécanique et celle de la puissance métabolique (marquées en phase oscillante) proviennent principalement de l'activation de muscles antagonistes. C'est le cas en début de cycle où les muscles GAS et VAS sont tous les deux activés, en fin de phase d'appui avec les muscles HAM et HFL, et en phase oscillante avec les muscles HAM et HFL.

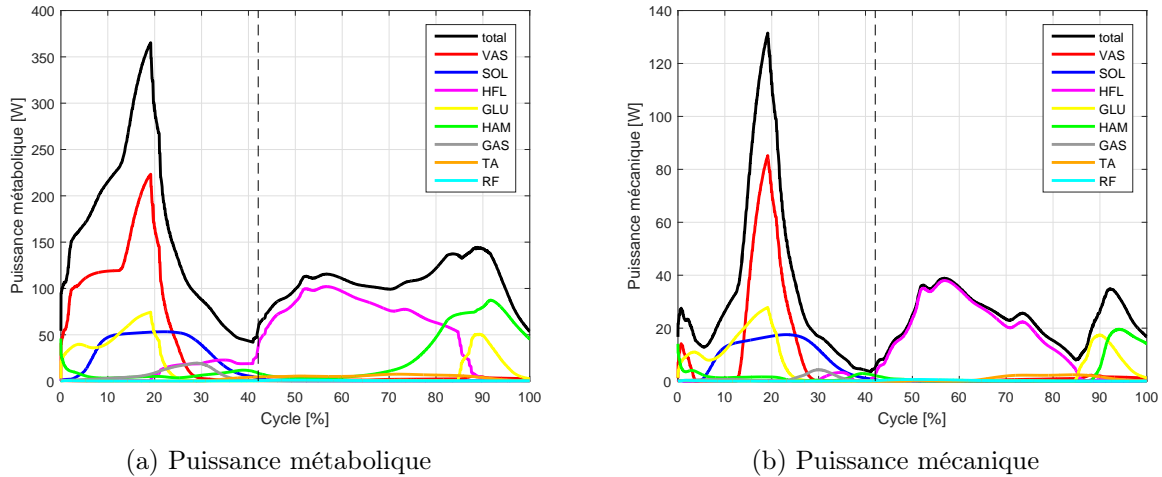


FIGURE 5.14 – Puissance métabolique et puissance mécanique consommée par les muscles virtuels pendant un cycle de course. La ligne verticale représente la transition phase d’appui/phase oscillante. Les traits en pointillés représentent les résultats obtenus pour la course avec le pied court. Les données sont moyennées sur 10 cycles.

5.4 Marche rapide

La marche rapide est celle représentée à la figure 5.1 avec le numéro 3. Ses caractéristiques principales sont rapellées dans le tableau ci-dessous.

| | énergie | | | |
|-------------------|------------------|---------------------------|---------------------------|--------------------------|
| | vitesse [m/s] | métabolique [J/(kg*m)] | longueur de foulée [m] | période de foulée [s] |
| 1. marche normale | 0.83 | 4.25 | 0.93 | 1.12 |
| 2. course | 1.64 | 6.15 | 0.79 | 0.48 |
| 3. marche rapide | 1.37 | 9.61 | 0.83 | 0.61 |

TABLE 5.4 – Caractéristiques principales de la marche normale, la marche rapide et la course. Les valeurs sont la moyennée sur cinq exécutions de chaque contrôleur.

On remarque dans ce tableau que sa vitesse (1.37 m/s), sa longueur de foulée (0.83 m) ainsi que sa période de foulée (0.48 s) se situent entre les valeurs de celles obtenues pour la course et la marche analysées dans les sections précédentes. Néanmoins, ces critères ne suffisent pas pour différencier la marche de la course, comme on peut l’observer sur la figure 5.1. L’énergie métabolique consommée par unité de distance lors de la marche rapide est nettement plus élevée que lors de la marche et de la course analysées précédemment, mais cela n’est également pas un critère suffisant, car des résultats de course et de marche consommant une énergie similaire ont été obtenus (voir figure 5.1).

Afin de déterminer les différences entre la marche rapide et la course, les caractéristiques suivantes sont analysées : les positions et couples articulaires, les forces de réaction du sol, l’énergie mécanique du centre de masse et les puissances métabolique et mécanique produites par les muscles.

5.4.1 Positions et couples des articulations sagittales des jambes

On note tout d’abord que la transition de la phase d’appui vers la phase oscillante s’effectue à 60%. Il y a donc deux périodes de double appui (et pas de phase de vol), caractéristique de

la marche. Les positions, couples et puissances articulaires lors d'un cycle sont représentés à la figure 5.15. Pour faciliter la comparaison avec les résultats de marche et de course analysés dans les sections précédentes, ceux-ci sont inclus sur la figure (en vert pour la marche, en rouge pour la course).

La position de la hanche est similaire à celle observée lors de la marche à vitesse optimale. Elle est légèrement plus fléchie en fin de cycle pour favoriser de grands pas. La position de l'articulation du genou évolue entre les résultats de marche et de course. Le genou est fléchi lors du début de la phase d'appui (comme pour la course, mais avec une amplitude moindre). La flexion du genou lors de la phase oscillante est de même amplitude que celle présente lors de la marche à vitesse optimale, et le genou reste légèrement fléchi en fin de cycle comme c'est le cas pour la course. La cheville est fléchie en début de cycle afin d'effectuer le contact avec le talon (la flexion est plus importante que celle lors de la marche à vitesse optimale, car le genou est fléchi en fin de cycle).

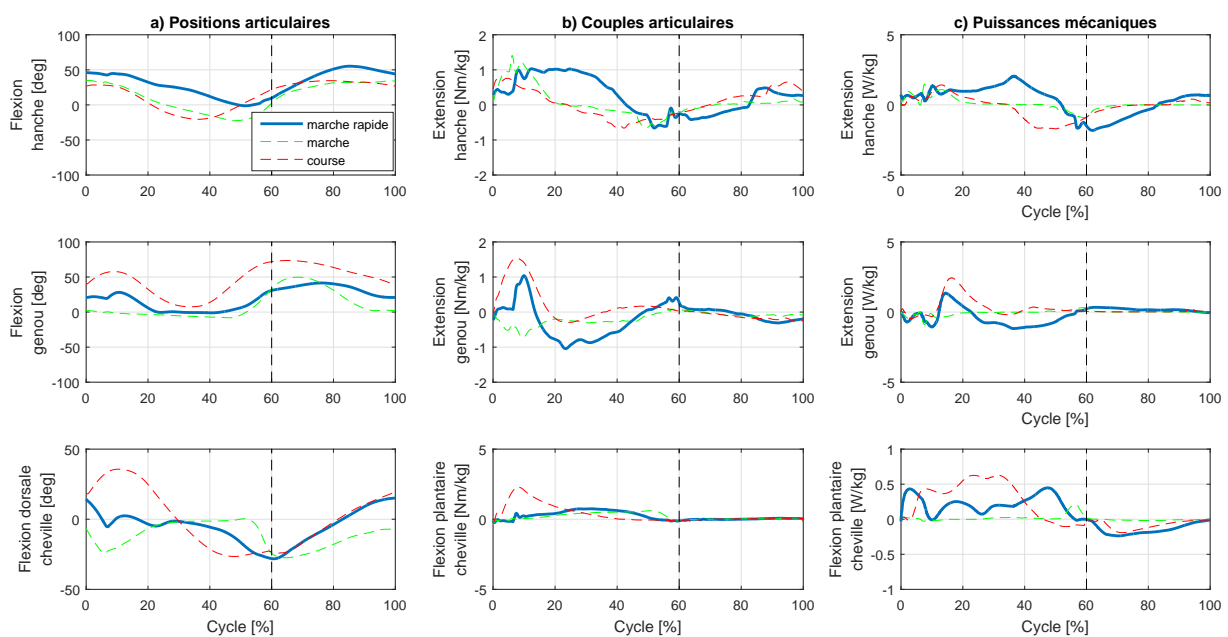


FIGURE 5.15 – Positions, couples et puissances mécaniques des articulations sagittales de la jambe droite du COMAN pendant un cycle de marche rapide. Les traits pointillés verts et rouges représentent les résultats de marche et de courses analysés aux sections 5.2 et 5.3, respectivement. La ligne verticale indique la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

5.4.2 Activations des muscles

On remarque sur la figure 5.16 que les stimulations envoyées aux différents muscles lors de la marche rapide sont plus importantes (et saturent durant le cycle pour la plupart d'entre elles) que lors de la marche à vitesse optimale ou lors de la course.

Une différence au niveau de la cheville est l'activation permanente du muscle GAS lors de la phase oscillante (alors qu'il n'est pas du tout activé pour la marche à vitesse optimale). Le muscle TA est également plus fortement activé lors de la phase oscillante. Comme pour la course, le muscle VAS est fortement utilisé en début de cycle pour étendre le genou. Les muscles HAM et GLU sont activés en début de cycle pour maintenir l'orientation du haut du corps. Le muscle RF est

fortement utilisé en phase oscillante, ce qui empêche l'extension de la jambe en fin de cycle.

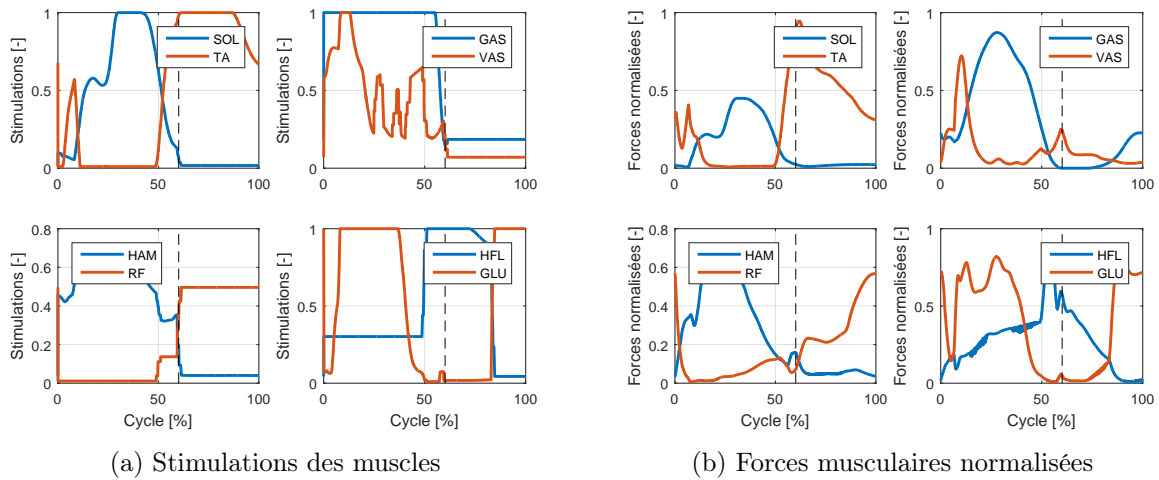


FIGURE 5.16 – Évolution des stimulations envoyées aux muscles ainsi que des forces musculaires générées par celles-ci lors d'un cycle de marche rapide. Les données sont moyennées sur 10 cycles.

5.4.3 Forces de réaction du sol

Les forces de réaction du sol sont similaires à celles observées lors de la marche à vitesse optimale, à savoir l'apparition de deux pics en phase d'appui. Le premier correspond au contact avec le sol, et le second à la propulsion.

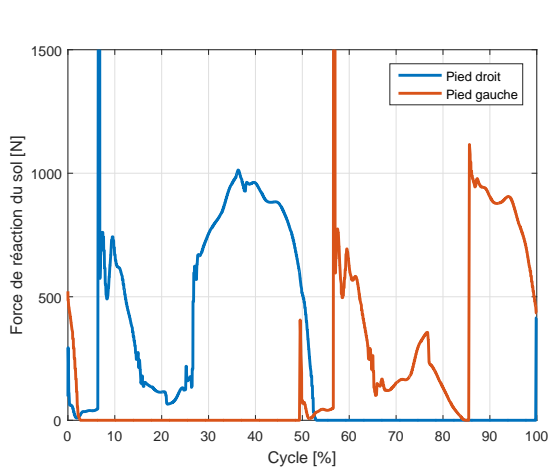


FIGURE 5.17 – Forces verticales de réaction du sol agissant sous le pied droit (trait bleu) et gauche (trait rouge) du robot pendant un cycle de marche rapide.

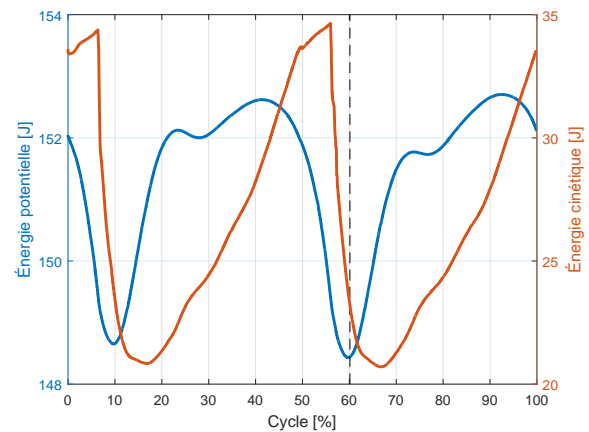


FIGURE 5.18 – Évolution des énergies cinétique et potentielle du centre de masse du robot pendant un cycle de marche rapide. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

5.4.4 Énergie mécanique du centre de masse

L'énergie potentielle et l'énergie mécanique du centre de masse lors d'un cycle de marche rapide sont représentées à la figure 5.18. L'énergie potentielle suit la même allure que celle présente lors de la marche à vitesse optimale : le robot bascule autour de la jambe d'appui (presque) tendue.

Elle est donc maximale en milieu de phase d'appui. Par contre, l'énergie cinétique diminue plus rapidement pour attendre son minimum avant la moitié de la phase d'appui. Il en résulte que ces deux énergies sont en léger déphasage. Il s'agit donc d'une situation différente de celles obtenues précédemment, où ces deux énergies étaient en phase ou en opposé de phase.

5.4.5 Énergie métabolique

Les puissances métabolique et mécanique sont représentées à la figure 5.19. On y retrouve à la fois les caractéristiques observées lors de la marche et lors de la course. Concernant la puissance mécanique, on retrouve une forte contribution des muscles GLU et HAM en phase d'appui afin de maintenir le torse droit (contribution principale lors de la marche), ainsi que du muscle VAS qui permet de contrer la flexion du genou (contribution principale lors de la course). En phase oscillante, la majeure contribution de la puissance mécanique provient du muscle HFL pour permettre la flexion du genou. Le muscle GLU fournit une puissance mécanique en fin de cycle pour préparer le contact initial (comme lors de la course).

La puissance mécanique est assez importante en phase d'appui (disparitions des pics) à cause de l'activation simultanée de nombreux muscles antagonistes.

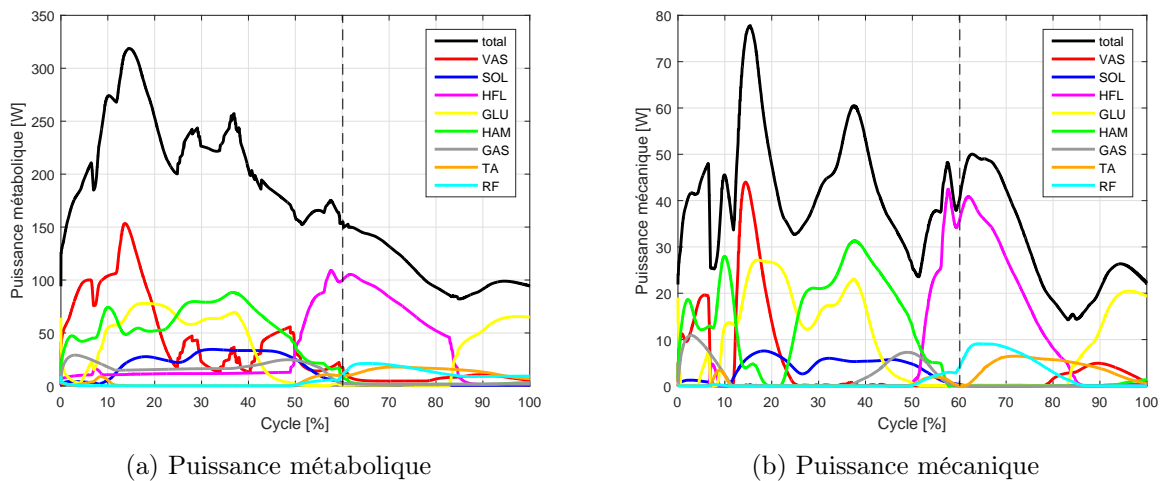


FIGURE 5.19 – Puissance métabolique et puissance mécanique consommées par les muscles virtuels pendant un cycle de marche rapide. La ligne verticale représente la transition phase d'appui/phase oscillante. Les données sont moyennées sur 10 cycles.

5.5 Comparaison des modèles de pieds

Dans cette section on compare les résultats de course obtenus avec les modèles de pieds avec orteils de tailles différentes. Les caractéristiques de ces courses sont rappelées au tableau 5.5.

| | énergie | | | |
|-----------------------|------------------|---------------------------|---------------------------|--------------------------|
| | vitesse [m/s] | métabolique [J/(kg*m)] | longueur de foulée [m] | période de foulée [s] |
| 2. course | 1.64 | 6.15 | 0.79 | 0.48 |
| 4. course pieds longs | 1.57 | 6.09 | 0.95 | 0.60 |

TABLE 5.5 – Caractéristiques principales de la course avec modèle de pieds à orteils courts (2) et longs (4). Les valeurs sont la moyennée sur cinq exécutions de chaque contrôleur.

On remarque que l'énergie consommée par unité de distance avec le modèle de pieds longs est moins importante qu'avec les pieds courts, et que le robot a tendance à faire des plus grandes foulées avec ces pieds long. La démarche est représentée à la figure 5.20. Elle est tout à fait similaire à celle obtenue avec le pied court (les différences avec les données humaines persistent). Les phases de vol sont néanmoins plus courtes (environ 10% contre 15% auparavant). Cela est principalement dû à la taille du pied qui engendre un contact initial plus rapide, et un décolllement du pied plus lent.

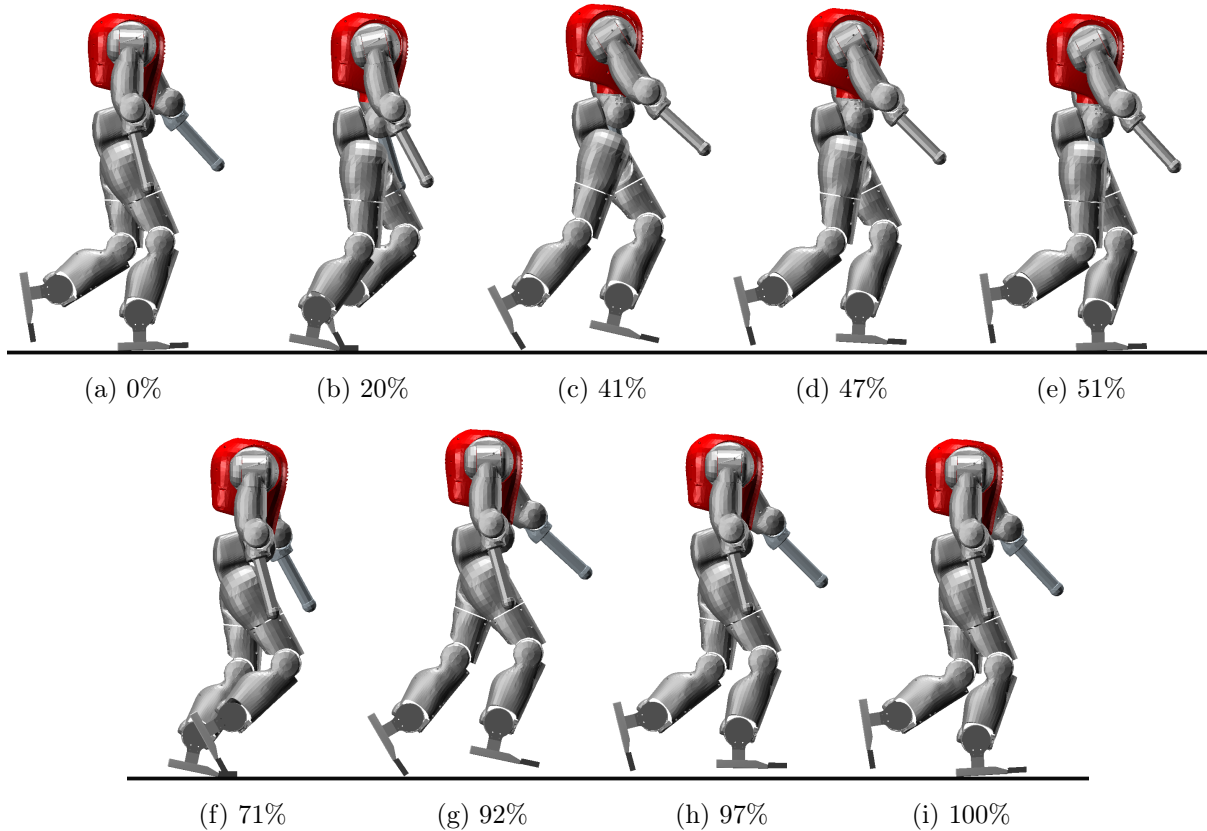


FIGURE 5.20 – Instantanés d'un cycle de course avec le modèle de pieds longs du COMAN à 1.57 m/s dans l'environnement de simulation. Les légendes indiquent le pourcentage du cycle.

Les positions et couples des articulations sagittales sont similaires à ceux obtenus pour les résultats de course avec le modèle de pieds courts. Il en est de même pour les forces de réaction du sol, l'énergie mécanique du centre de masse, ainsi que les contributions de chacun des muscles dans la puissance mécanique.

Les différences sont donc uniquement liées à la taille du pied. D'une part, le modèle avec pieds longs est bénéfique, car le bras de levier étant plus important au niveau des orteils, une plus grande quantité d'énergie élastique peut donc être libérée au niveau de l'articulation passive. La hauteur maximale du centre de masse est également plus élevée (figure 5.21).

D'autre part, cette taille engendre des phases de vol moins significatives, et le bout du pied en phase oscillante est très proche du sol (figure 5.20f). Pour éviter de trébucher avec le bout des pieds, le muscle TA est plus activé qu'avec le modèle de pieds courts, mais la taille du pied reste tout de même un facteur de chute important. Les stimulations envoyées aux muscles sont représentées à la figure 5.16a (pour faciliter la comparaison avec la course obtenue avec le modèle de pieds courts avec orteils, les stimulations de celle-ci sont représentées en traits pointillés).

Une autre différence importante au niveau des stimulations est le muscle HAM qui est moins activé en phase oscillante. Cela réduit l'énergie métabolique consommée dans la mesure où HAM

et HFL ne sont plus activés en même temps.

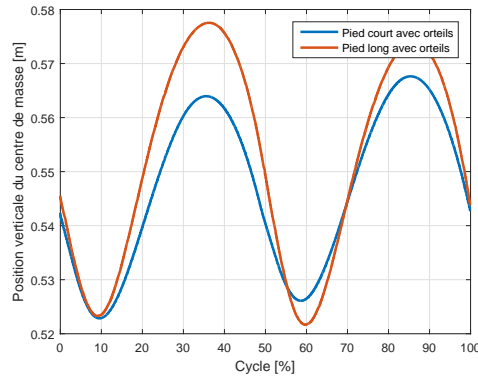


FIGURE 5.21 – Évolution de la position verticale du centre de masse du robot pendant un cycle de course. La courbe bleue est la course avec le modèle de pieds courts avec orteils, et la courbe rouge utilise le modèle de pieds longs avec orteils. Les données sont moyennées sur 10 cycles.

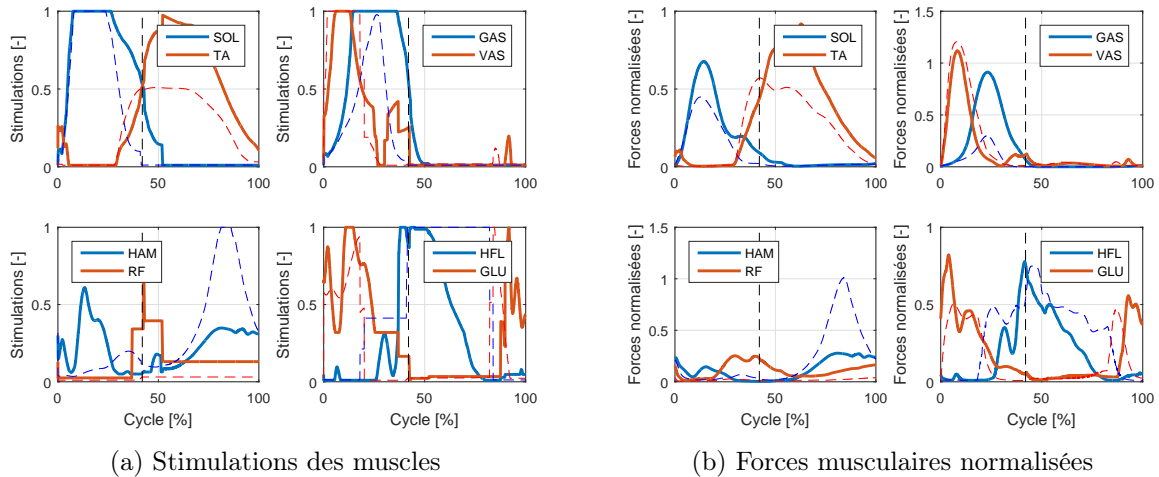


FIGURE 5.22 – Évolution des stimulations envoyées aux muscles ainsi que des forces musculaires générées par celles-ci lors d'un cycle de course avec le modèle de pieds longs avec orteils. Les traits pointillés représentent les résultats de course avec le pied court analysés à la section 5.3. Les données sont moyennées sur 10 cycles.

5.6 Contrôle musculaire du haut du corps

Le but de cette section est de valider le contrôleur musculaire du haut du corps décrit à la section 4.4. Dans le cadre de ce mémoire, seules les articulations sagittales des bras sont contrôlées au moyen du modèle musculaire avec un contrôleur proportionnel.

La figure 5.23 montre l'évolution des positions et stimulations musculaires des articulations sagittales du bras droit pendant trois cycles de course. La position de référence des épaules est proportionnelle à la différence d'angle au niveau des hanches (figure du bas en bleu), et celle des coudes est constante (figure du bas en orange). Les stimulations des différents muscles concernés sont représentées sur la figure du haut.

On remarque que les positions de références sont relativement bien suivies (comme expliqué à la section 4.4, l'influence du haut du corps est très limitée sur la locomotion, le tracking ne doit

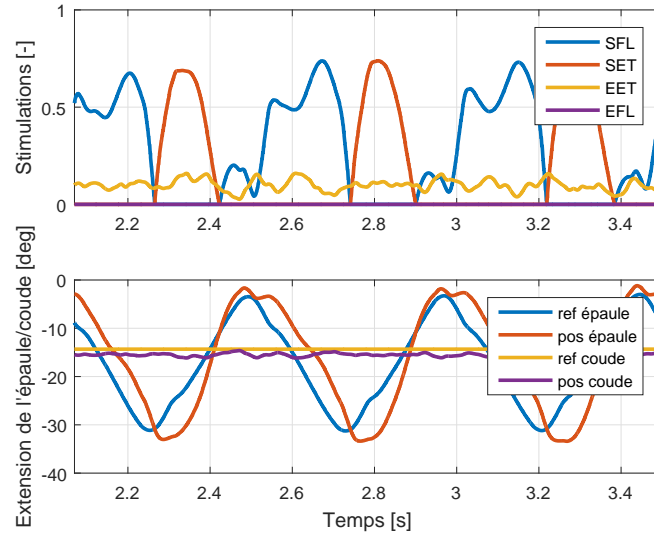


FIGURE 5.23 – Évolution des références et positions sagittales de l'épaule et du coude droits du robot lors de trois cycles en course. Les stimulations envoyées aux muscles sont représentées à la figure du dessus.

donc pas être parfait). Le contrôleur musculaire pour le torse et les articulations non sagittales du haut du corps (voir annexe C.3) a également été testé même s'il n'est pas utilisé dans ce projet. Les résultats de ce dernier sont montrés à l'annexe E.1.

5.7 Conclusion

Ce chapitre a montré que le contrôleur présenté au chapitre 4 permet d'obtenir des résultats de marche et de course en simulation (avec le modèle développé au chapitre 3). Les lois simples de contrôle basées sur le modèle musculaire permettent d'obtenir des activations complexes nécessaires à la réalisation de la démarche.

La marche du COMAN est optimale (dans le sens énergétique du terme) à une vitesse de 0.83 m/s. Les résultats cinématiques obtenus sont très proches de la démarche d'un être humain. La principale différence est la non-flexion du genou après l'impact avec le sol.

Les forces de réaction verticale du sol présentent deux pics lors de la phase d'appui (au début et à la fin de celle-ci), et la masse du robot est transférée progressivement d'une jambe à l'autre lors de la période de double appui.

La jambe restant tendue lors de la phase d'appui, l'énergie cinétique et l'énergie potentielle du centre de masse sont déphasées de 180 degrés, ce qui permet un transfert d'énergie.

L'énergie métabolique consommée durant un cycle possède deux pics. Le premier survient après le contact initial, et a pour but de maintenir le torse droit au moyen des muscles de la hanche GLU et HAM. Le second survient en fin de phase d'appui, et permet de propulser la jambe vers la phase oscillante au moyen des muscles de la hanche HFL et RF, ainsi que du muscle de la cheville SOL.

Le résultat de course minimisant l'énergie consommée est obtenu à une vitesse de 1.64 m/s. Les positions des articulations sagittales de la jambes suivent les mêmes allures que les données humaines, mais des différences existent : le genou est trop fléchi lors du contact initial, et ce dernier ne se produit pas avec le talon. Les muscles responsables de l'extension du genou ne sont pas assez activés en fin de cycle (le contrôleur permet cependant leurs activations). Un changement

de la fonction d'objectif doit sans doute être effectué pour tenter de résoudre ce problème. Les forces de réaction verticale du sol présentent un grand pic en phase d'appui. Les phases de vol sont facilement identifiables au moyen de celles-ci.

Lors de la course, la jambe se fléchit après le contact initial, pour ensuite s'étendre et propulser le robot. Il en résulte que les énergies potentielle et cinétique sont en phase.

L'énergie métabolique consommée durant un cycle est plus importante que lors de la marche, et possède 3 pics. Le premier permet à la fois le soutien des jambes après le contact initial (au moyen des muscles de la hanche GLU et du genou VAS) et la propulsion du robot vers l'avant (au moyen des muscles de la cheville SOL et GAS). Les muscles VAS et GAS sont fortement activés lors de la période d'appui, alors qu'ils ne le sont pas pendant la marche. Le deuxième pic intervient au début de la phase oscillante et permet d'élever la jambe au moyen du muscle de la hanche HFL. Enfin, le troisième pic permet de préparer le contact avec le sol au moyen des muscles de la hanche HAM et GLU. Le muscle HAM est également fortement activé en fin de phase oscillante, alors que ce n'est pas le cas lors de la marche.

La marche rapide (1.37 m/s) a également été analysée. Les activations musculaires lors de celle-ci sont intenses, sans toutefois réussir à faire décoller le robot. Cela se traduit par une forte consommation d'énergie en comparaison à la marche à vitesse optimale. On observe dans les différentes caractéristiques un mélange de celles observée lors de la marche et lors de la course.

Une comparaison entre le modèle de pieds longs avec orteils et le modèle de pieds courts avec orteils a également été effectuée. Il en résulte que la propulsion est favorisée par le bras de levier plus important au niveau de l'articulation passive des orteils. Néanmoins, la taille du pied est un désavantage lors de la phase oscillante car le bout de celui-ci est très proche du sol. Cela entraîne trop souvent la chute du robot, c'est pourquoi le modèle de pieds courts est préféré.

Enfin, le contrôleur du haut du corps a été validé. Celui-ci utilise le nouveau modèle musculaire du haut du corps. Même si seules les articulations sagittales des bras sont contrecollées via celui-ci, l'ensemble du contrôleur a été testé.

Chapitre 6

Discussion & conclusion

La locomotion joue un rôle essentiel dans le développement des robots humanoïdes. Ceux-ci possèdent de nombreux avantages par rapport aux autres types de robots mobiles, de par leur morphologie proche de celle d'un être humain. Beaucoup de contrôleurs différents ont été développés, mais aucun d'entre eux ne permet (pour le moment) de reproduire une démarche avec des performances comparables à celles des êtres humains (efficace énergiquement, robuste à des perturbations extérieures, vitesse, etc.).

La démarche bio-inspirée a pour but de reproduire une locomotion plus proche de celle-ci. Des résultats convaincants de marche ont été obtenus par Nicolas Van der Noot [67] sur le robot utilisé dans le cadre de ce mémoire, le COMAN, en simulation et sur le vrai robot en utilisant un modèle de réflexes introduit par H. Geyer [29]. L'évolution naturelle de cette démarche mène à essayer d'augmenter la vitesse du robot, afin d'obtenir des résultats de course. Il s'agit de l'objectif de ce mémoire. On se base pour cela sur le contrôleur proposé par J. M. Wang [70] qui est une extension de celui proposé par H. Geyer.

Pour implémenter le contrôleur, un simulateur du robot COMAN et de son environnement est utilisé. Ce dernier permet de tester facilement le contrôleur, tout en étant le plus proche possible des conditions réelles trouvées sur le vrai robot (utilisation uniquement des entrées disponibles sur le vrai robot, ajouts de bruits à la lecture des capteurs de couple des articulations, etc.). Cet outil est également indispensable pour effectuer les phases d'optimisation requises afin de déterminer les valeurs des paramètres du contrôleur permettant d'obtenir la démarche souhaitée. Il permet aussi d'implémenter facilement le contrôleur uniquement pour les articulations sagittales (contrôleur 2D).

6.1 Évaluation des résultats

Des résultats convaincants de marche et de course ont été obtenus avec le contrôleur implémenté, et ce jusqu'à une vitesse maximale de 1.8 m/s. Il s'agit d'une réelle amélioration par rapport aux marches déjà obtenues sur le COMAN qui plafonnaient vers 0.9 m/s.

Les principales modifications apportées au modèle de réflexes introduit par H. Geyer est l'ajout de deux phases, la préparation à la phase d'appui et l'initiation à la phase oscillante, durant lesquelles les lois de contrôle sont modifiées. Le modèle de pieds du robot a également été modifié en ajoutant une articulation passive au niveau des orteils.

Les résultats de marche obtenus sont très proches de la démarche humaine, et similaire à ceux déjà obtenus sur le robot [5]. La vitesse qui minimise la consommation d'énergie par distance parcourue (vitesse naturelle) est 0.8 m/s.

La course, caractérisée par deux phases de vol, a été obtenue pour des vitesses allant de 1.2 m/s à 1.8 m/s. La démarche suit les caractéristiques générales observées chez l'être humain, mais des différences persistent néanmoins. La principale est la flexion trop importante du genou lors du contact initial. Outre les positions et couples articulaires, la démarche de course peut être différenciée de la marche par l'analyse de l'évolution des données suivantes : les stimulations envoyées aux muscles, les forces de réaction verticale du sol, l'énergie mécanique du centre de masse ou encore l'énergie métabolique consommée.

6.2 Limitations et perspectives

Pour conclure, nous identifions les limites de ce contrôleur, et nous proposons différentes idées d'amélioration. Celles-ci permettraient vraisemblablement d'obtenir de meilleurs résultats de course, et/ou de faciliter l'utilisation du contrôleur sur le vrai robot.

De façon générale, l'environnement de simulation ainsi que le contrôleur devront subir des améliorations afin d'obtenir des résultats identiques sur le vrai robot. La limite de couple des moteurs du vrai robot n'a pas été tenue en compte lors des simulations, ce qui risque de pas assurer la reproduction les résultats obtenus.

Une amélioration notable serait de pouvoir effectuer une transition de la marche vers la course, et inversement. En effet, tester le contrôleur actuel sur le vrai robot n'est pas envisageable de par la configuration initiale du robot. Des tentatives de changements abrupts des valeurs des paramètres pour passer d'une démarche à une autre ont été effectuées, mais sans résultat probant. Une solution possible pour réaliser cette transition est l'utilisation d'un modèle CPG combiné avec le modèle de réflexe actuel (comme cela est fait dans [67]) afin moduler la vitesse du robot. Il s'agit d'une amélioration importante, qui permet également de réduire les optimisations dans la mesure où actuellement chaque démarche différente doit être l'objet de nouvelles optimisations (où les 71 paramètres du contrôleur doivent être renouvelés). L'utilisation de CPG permettrait de moduler un nombre de paramètres restreints pour changer la démarche.

La fonction d'objectif peut également être modifiée afin de converger plus facilement vers des résultats de course. En effet, les optimisations avec la fonction actuelle n'aboutissent pas systématiquement à de une course du robot. Les différences cinématiques avec les données humaines peuvent également être comblées au moyen de cette fonction (le contrôleur ne limitant pas l'extension du genou en fin de cycle).

De plus, des modifications peuvent également être apportées afin de rendre le robot plus robuste aux perturbations extérieures. Cela peut se faire d'une part en modifiant le contrôleur, et d'autre part en ajoutant des perturbations aléatoires lors des phases d'optimisation.

En outre, le contrôleur peut être étendu à une démarche 3D, c'est à dire non contrainte latéralement. Le modèle musculaire, tant pour le bas que pour le haut du corps, est déjà implémenté dans le contrôleur pour les degrés de liberté supplémentaires.

Enfin, le contrôle du haut du corps peut-être amélioré en déterminant des lois de réflexes modélisant son mouvement. En effet, pour le moment il est contrôlé à partir de positions de référence.

Annexe A

Améliorations du projet pour aboutir à des résultats de course

Cette annexe a pour but de brièvement expliquer les différentes modifications effectuées au projet pour obtenir des résultats de course. En effet, la course du robot n'a été possible que tardivement (début juillet), et le projet a subi de nombreux changements pour arriver à ce résultat. Le principal problème était que les optimisations convergeaient généralement vers des résultats de marche rapide c'est à dire supérieure à 1 m/s (la vitesse de marche la plus élevée atteinte sur le COMAN dans [67] est de 0.9 m/s). Outre le contrôleur utilisé, l'état initial du robot ainsi que la fonction d'objectif sont des paramètres importants pour obtenir la démarche souhaitée.

Modèle de pieds rigides et fonction d'objectif étagée

Initialement, le robot est composé de pieds rigides (figure 3.3a). Le contrôleur tel qu'expliqué à la section 4.5 est implémenté. La fonction d'objectif utilisée initialement est une fonction étagée. Le principe est qu'une tâche est accessible seulement si la précédente est complétée. Cela permet de s'assurer du respect de certaines tâches en priorité. Dans le cas présent, les tâches définies sont les suivantes :

1. Le robot est capable de se déplacer sur une distance minimale pendant le temps de simulation ;
2. Le robot est capable de ne pas tomber durant le temps de simulation. Pour entrer dans cette tâche, le robot doit avoir effectué le déplacement minimal défini dans la première tâche (afin de ne pas trouver une solution où le robot est en équilibre) ;
3. Le robot est capable de se déplacer à une vitesse donnée. Pour entrer dans cette tâche, le robot doit avoir respecté la tâche précédente ;
4. Le robot minimise la consommation de son énergie métabolique par unité de distance parcourue. Pour entrer dans cette tâche, le robot doit se déplacer à une vitesse proche de celle imposée à la tâche précédente (± 0.05 m/s).

L'initialisation du robot est réalisée en penchant le COMAN vers l'avant de manière à ce que la position selon l'axe x de son centre de masse suive une position référence à optimiser. Pour réaliser cela, le contrôleur en impédance avec inversion musculaire de [34] est utilisé pendant les deux premières secondes de la simulation.

Les premiers résultats ont été obtenus en imposant simplement une vitesse de référence élevée. Seuls des résultats de marche ont été trouvés, et ce avec une vitesse qui sature autour de 1m/s.

Modèle de pieds avec orteils

Pour favoriser la course, un modèle de pieds avec orteils a été créé (figure 3.3b, voir section 3.1.2). Cette démarche est motivée par le fait que l'articulation passive peut stocker de l'énergie pour potentiellement la libérer lors de la propulsion. De plus, le robot utilisé dans [70] (document sur lequel le contrôleur utilisé est basé) est également muni de pieds flexibles. La valeur du coefficient de raideur a été fixée suite à des optimisations.

Ce modèle a permis d'augmenter la vitesse de marche du robot, sans toutefois réussir à le faire courir.

Minimisation du double appui et maximisation de la phase de vol

Afin de favoriser la course, deux tâches ont été ajoutées à la fonction d'objectif. Celles-ci sont insérées entre la tâche qui impose une vitesse d'avance et celle qui minimise l'énergie métabolique (tâches 3 et 4 ci-dessus). Voici les deux tâches supplémentaires qui sont calculées en parallèle (les autres tâches sont toujours accessibles seulement si la précédente est complétée) :

- Minimiser le temps de double appui lors d'un cycle ;
- Maximiser le temps de vol lors d'un cycle.

Cette fonction d'objectif est utilisée avec ou sans contrainte de vitesse (tâche 3). Ces optimisations ont fourni des résultats de marche (très) rapide (jusqu'à 1.8 m/s) ou de course théorique. On parle de course théorique, car la phase de vol est très courte, et la démarche ressemble visuellement plus à de la marche qu'à de la course (d'ailleurs ces résultats ont une vitesse proche de 1 m/s).

Forces musculaires maximales doublées

Pour la plupart des résultats de marche rapide ou de course (théorique), certains muscles saturent (la stimulation envoyée est maximale) pendant le cycle. C'est le cas des muscles SOL (en partie responsable de la propulsion) et TA. Pour s'assurer que ces muscles ne soient pas sous-dimensionnés (non capables de produire un couple suffisant pour la course), leur force maximales isométriques ont été doublées. Les résultats obtenus sont similaires aux précédents (avec des saturations moindres).

Modèle de pieds raccourci

Dans les résultats obtenus précédemment, le muscle TA est activé au maximum quasiment pendant tout le cycle, et en particulier lors de la phase oscillante. Ce muscle est responsable de la dorsiflexion du pied. Nous en avons déduit que l'optimisateur active ce muscle afin d'éviter que le bout du pied ne touche le sol lors de la phase oscillante, ce qui engendre une chute du robot. Pour résoudre ce problème, la longueur des pieds a été réduite tout en gardant les propriétés des pieds flexible (voir figure 3.3c).

Les résultats obtenus avec ce nouveau modèle de pieds ne changent pas significativement (même si l'activation du muscle pour la phase oscillante est moins présente).

Modification des réflexes pour le premier pas

Afin de favoriser la propulsion en vue d'un résultat de course, les bornes inférieures des paramètres de la rétroaction positive en force des muscles SOL et GAS ont été adaptées (G_{SOL} et G_{GAS} , voir équation (4.24)). En effet, ces deux muscles sont majoritairement responsables de la propulsion. Or il a été constaté que la stimulation envoyée au muscle GAS est quasi nulle dans les résultats obtenus. Suite à ces modifications, le robot tombe souvent lors des premiers pas. Pour résoudre

ce problème, les stimulations des muscles liés à la cheville ont été modifiées pour le premier pas de la jambe en phase d'appui :

$$S_{SOL} = S_{min} \ ; \ S_{GAS} = S_{min} \ ; \ S_{TA} = S_{max} \quad (\text{A.1})$$

où S_{min} est la stimulation minimale (0.01) et S_{max} est la stimulation maximale (1.0).

Suite à ces modifications, le muscle GAS est effectivement utilisé (cela est forcé via les bornes d'optimisation) mais l'objectif de course n'est cependant pas atteint. Suite à ce changement, nous avons même eu des résultats surprenants : le robot marche en arrière !

Optimisation des objectifs en parallèle

La fonction d'objectif a été modifiée pour être celle présentée dans la section 4.7.1. Elle est inspirée de celle utilisée dans l'article sur lequel est basé le contrôleur [70]. Une différence par rapport à celle utilisée précédemment est que toutes les tâches sont effectuées en parallèle. En effet, imposer que le robot ne tombe pas pendant tout le temps de la simulation avant de tenir compte des autres tâches tend (peut-être) à converger vers des solutions locales de marche. De plus, la pondération de chaque tâche est adaptée.

La démarche des résultats utilisant cette nouvelle fonction d'objectif ressemble plus visuellement à la course (surtout grâce à la flexion du genou dans la phase oscillante), mais il s'agit encore de marche.

Commutation de contrôleurs

Afin d'éviter que les solutions convergent dans un optimum local de marche à cause de l'état initial du robot (à l'arrêt), un commutateur de contrôleur est implémenté. L'idée est d'utiliser un résultat précédent (marche rapide ou course) pour les premiers pas, et d'ensuite changer abruptement les valeurs des paramètres. Le moment du cycle où ce changement est effectué est un nouveau paramètre à optimiser.

Encore une fois, cette méthode n'a pas permis d'obtenir de résultats de course (mais de la marche un peu plus rapide).

Paramètres permettant de suivre les couples articulaires

Afin de vérifier si les bornes choisies pour les différents paramètres à optimiser sont correctes, une pseudo dynamique inverse a été implémentée. Les positions articulaires du robot sont imposées selon un résultat de course issu de l'article sur lequel le contrôleur est basé [70]. Le but est de trouver les valeurs des paramètres du contrôleur pour suivre les couples articulaires obtenus pour ce résultat de course. Pour ce faire, une optimisation a été effectuée en tenant compte uniquement de la partie contrôleur (la dynamique du robot et l'environnement extérieur étant inclus dans les couples articulaires à générer). La fonction d'objectif est d'obtenir le couple articulaire correspondant aux positions qui sont imposées.

Les résultats ont démontré que les couples articulaires pouvaient être suivis avec les bornes utilisées. Cependant, les valeurs trouvées ne donnent pas la démarche voulue. Cela est dû au fait que le robot et sa dynamique sont différents que dans [70], et l'environnement n'est probablement pas modélisé de la même façon. De plus, les résultats fournis le sont en supposant $\theta_{torse} = 0$ car sa position n'est pas fournie dans [70] (il s'agit d'une grosse approximation).

Positions et vitesse initiales à optimiser

Un résultat de course fut finalement trouvé en optimisant l'état initial du robot. L'initialisation est décrite à la section 4.6.

Annexe B

Outils de simulation

B.1 CECI clusters

Quatre machines disponibles dans le consortium ont été configurées pour le projet afin de maximiser le nombre de simulations à pouvoir être générées : NIC4 (ULg), Vega (ULB), Lemaitre2 (UCL) et Hmem (UCL). Leurs principales caractéristiques sont disponibles sur le site web du CECI : <http://www.cec-hpc.be/clusters.html>. Configurer plusieurs machines est très utile, car elles peuvent être utilisées par tous les membres du consortium, et leur disponibilité (processeurs et mémoire disponible) peut dès lors fortement varier en fonction du temps. De plus, les ressources de chaque machine sont allouées par un planificateur de tâches pour assurer un accès équitable entre tous les utilisateurs : pour chaque tâche soumise, une priorité lui est accordée en fonction non seulement des ressources et temps demandés pour cette tâche, mais également en fonction de l'historique de l'utilisateur (plus un utilisateur utilise une machine en demandant beaucoup de ressources, moins il devient prioritaire à l'avenir). Le planificateur de tâches employé est SLURM¹ (pour *Simple Linux Utility for Ressource Management*), qui est l'outil utilisé sur la majorité des plus puissants supercalculateurs de la planète.

Pour plus d'informations concernant la configuration et l'utilisation du projet, le lecteur peut se référer au projet Git https://git.immc.ucl.ac.be/bsomers/coman_bruno. Notons qu'il est possible de lancer les optimisations sur son ordinateur personnel en y installant MPI.

B.2 Algorithme CMA-ES

Afin de mettre à jour les paramètres à optimiser vers la meilleure solution (évaluée au moyen de la fonction d'objectif), il faut déterminer comment calculer $m^{(g+1)}$, $C^{(g+1)}$ et $\sigma^{(g+1)}$ pour la génération suivante $g + 1$ (voir equation (3.8)). Cette procédure est expliquée dans cette section.

La nouvelle moyenne $m^{(g+1)}$ est une moyenne pondérée des μ meilleures valeurs de $x_1^{(g+1)}, \dots, x_{i:\lambda}^{(g+1)}$:

$$m^{(g+1)} = \sum_{i=1}^{\mu} w_i x_{i:\lambda}^{(g+1)} \quad (\text{B.1})$$

où μ est le nombre de parents de la population, $w_{i=1 \dots \mu}$ sont des poids de recombinaison avec $\sum_{i=1}^{\mu} w_i = 1$ et $w_1 \geq w_2 \geq \dots \geq w_{\mu} > 0$, et $x_{i:\lambda}^{(g+1)}$ est le i ème meilleur individu de $x_1^{(g+1)}, \dots, x_{\lambda}^{(g+1)}$ (classé selon le résultat de la fonction d'objectif). Typiquement, on effectue la moyenne sur moins de la moitié de la population ($\mu \leq 0.5$), et les poids sont déterminés de telle sorte que la variance

1. <http://slurm.schedmd.com>

effective $\mu_{eff} = 1 / \sum_{i=1}^{\mu} w_i^2$ soit plus ou moins égale à $\lambda/4$.

La mise à jour de la matrice de covariance C est une combinaison d'un système efficace pour une petite population, et d'un système efficace si seulement un point est utilisé pour la mise à jour pour plusieurs générations. Tout d'abord la mise à jour pour une petite population :

$$C^{(g+1)} = (1 - c_{\mu} \sum_{i=1}^{\lambda} w_i) C^{(g)} + c_{\mu} \sum_{i=1}^{\lambda} w_i y_{i:\lambda}^{(g+1)} \left(y_{i:\lambda}^{(g+1)} \right)^T \quad (\text{B.2})$$

où $c_{\mu} \leq 1$ est un taux d'apprentissage pour la mise à jour de la matrice de covariance, et $y_{i:\lambda}^{(g+1)} = (x_{i:\lambda}^{(g+1)} - m^{(g)}) / \sigma^{(g)}$. La mise à jour de la matrice de covariance en n'utilisant qu'un point à chaque génération est définie à partir de la trajectoire d'évolution :

$$p_c^{(g+1)} = (1 - c_c) p_c^{(g)} + \sqrt{c_c(2 - c_c) \mu_{eff}} \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \quad (\text{B.3})$$

où $p_c^{(g)} \in \mathbb{R}^n$ est la trajectoire d'évolution à la génération g , $c_c < 1$, et $\sqrt{c_c(2 - c_c) \mu_{eff}}$ est une constante de normalisation. La mise à jour de la matrice de covariance s'exprime alors comme suit :

$$C^{(g+1)} = (1 - c_1) C^{(g)} + c_1 p_c^{(g+1)} p_c^{(g+1)T} \quad (\text{B.4})$$

Finalement, la mise à jour de la matrice de covariance est une combinaison de (B.2) et (B.4) :

$$C^{(g+1)} = (1 - c_1 - c_{\mu} \sum_{i=1}^{\lambda} w_i) C^{(g)} + c_1 p_c^{(g+1)} p_c^{(g+1)T} + c_{\mu} \sum_{i=1}^{\lambda} w_i y_{i:\lambda}^{(g+1)} \left(y_{i:\lambda}^{(g+1)} \right)^T \quad (\text{B.5})$$

La mise à jour de la taille de pas σ utilise le *Cumulative Step-size Adaptation* (CSA). Tout d'abord la trajectoire d'évolution est mise à jour :

$$p_{\sigma}^{(g+1)} = (1 - c_{\sigma}) p_{\sigma}^{(g)} + \sqrt{c_{\sigma}(2 - c_{\sigma}) \mu_{eff}} C^{(g)-0.5} \frac{m^{(g+1)} - m^{(g)}}{\sigma^{(g)}} \quad (\text{B.6})$$

où $p_{\sigma}^{(g)} \in \mathbb{R}^n$ est la trajectoire d'évolution conjuguée à la génération g , $c_{\sigma} < 1$, et $\sqrt{c_{\sigma}(2 - c_{\sigma}) \mu_{eff}}$ est une constante de normalisation. On obtient ensuite :

$$\sigma^{(g+1)} = \sigma^{(g)} \exp \left(\frac{c_{\sigma}}{d_{\sigma}} \left(\frac{\|p_{\sigma}^{(g+1)}\|}{E\|\mathcal{N}(0, I)\|} \right) - 1 \right) \quad (\text{B.7})$$

où d_{σ} est un paramètre d'amortissement. La taille de pas augmente seulement si $\|p_{\sigma}\|$ est supérieur à l'espérance $E\|\mathcal{N}(0, I)\|$, et inversement.

Les entrées de l'algorithme sont la moyenne initiale m , la taille de pas initiale σ , le nombre de particules λ ainsi que le nombre de générations. De plus, pour la première itération, on initialise C à la matrice identité I , $p_c = 0$ et $p_{\sigma} = 0$. Les paramètres suivants sont utilisés par défaut : $c_c \approx 4/n$, $c_{\sigma} \approx 4/n$, $c_1 \approx 2/n^2$, $c_{\mu} \approx \mu_{eff}/n^2$, $c_1 + c_{\mu} \leq 1$, $d_{\sigma} \approx 1 + \sqrt{\mu_{eff}/n}$, et $w_{i=1\dots\lambda}$ tel que $\mu_{eff} = 1 / \sum_{i=1}^{\mu} w_i^2 \approx 0.3\lambda$.

Plus on augmente le nombre de particules, plus on a de chance de ne pas s'enfermer dans un optimum local dès le début de la simulation. Plus le nombre de générations est élevé, plus les particules ont le temps de converger vers une solution unique (l'optimum trouvé). Dans la majorité des cas, les optimisations sont lancées avec 70 particules pour 1000 générations, avec une moyenne initiale 0.5 et une taille de pas initiale de 0.3.

Annexe C

Contrôleur

C.1 Relations force-longueur et force-vitesse d'un MTU

Les relations force-longueur $f_l(l_{ce})$ et force-vitesse $f_v(v_{ce})$ s'expriment comme suit (depuis [30]) :

$$f_l(l_{ce}) = \exp\left(c \left| \frac{l_{ce} - l_{opt}}{l_{opt}w} \right|^3\right) \quad (C.1)$$

où $w = 0.56$ représente la largeur de la cloche de $f_l(l_{ce})$, et $c = \ln(0.05)$ est tel que $f_l(l_{opt}(1 \pm w)) = 0.05$.

$$f_v(v_{ce}) = \begin{cases} \frac{v_{max} - v_{ce}}{v_{max} + K v_{ce}}, & \text{si } v_{ce} < 0 \\ N + (N - 1) \frac{v_{max} + v_{ce}}{7.56K v_{ce} - v_{max}}, & \text{si } v_{ce} \geq 0 \end{cases} \quad (C.2)$$

où $K = 5$ est une constante de courbe, $N = 1.5$ est la quantité de force F_{MTU}/F_{max} obtenue à une vitesse $v_{ce} = -v_{max}$, et la vitesse de contraction maximale $v_{max} = -12l_{opt}s^{-1}$.

C.2 Caractérisation des muscles

C.2.1 Muscles des jambes

Les paramètres caractérisant les muscles de la jambe sont présentés dans le tableau C.1. Ils sont issus de [65] (eux mêmes issus de [29]), excepté pour le muscle RF dont les données proviennent de [60]. Les valeurs ont ensuite été redimensionnées pour correspondre à la morphologie du COMAN ainsi qu'aux références des différentes articulations. Notons que pour le muscle RF, l'approximation pour calculer le bras de levier ainsi que la longueur du muscle a changé par rapport à [29] (formule 4.4 et 4.6) pour le genou et la cheville :

$$r_m(\phi) = r_{max} \cos[k(\phi - \phi_{max})] \quad (C.4)$$

$$\Delta l_{mtu} = \rho \frac{r_{max}}{k} [\sin([k(\phi - \phi_{max})]) - \sin([k(\phi_{ref} - \phi_{max})])] \quad (C.5)$$

avec $k = \cos^{-1} \frac{r_{min}/r_{max}}{\phi_{min} - \phi_{max}}$

Ces valeurs sont arrondies dans le code (elles sont en effet déjà basées sur des approximations des différents muscles).

| | SOL | TA | GAS | VAS | HAM | RF | GLU | HFL |
|---------------------------|------------|-----------|------------------------|------------|------------------------|---|------------|------------|
| F_{max} [N] | 1416 | 283.2 | 531 | 2124 | 1062 | 424.8 | 531 | 708 |
| v_{max} [l_{opt}/s] | 9.18 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 |
| l_{opt} [cm] | 1.708 | 2.562 | 2.135 | 3.416 | 4.27 | 0.034 | 4.697 | 4.697 |
| l_{slack} [cm] | 11.102 | 10.248 | 17.08 | 9.821 | 13.237 | 0.149 | 5.551 | 4.27 |
| r_0 [cm] | 2.135 | 1.708 | 2.135 (c) 2.135 (g) | 2.562 | 2.135 (g) 3.416 (h) | 2.562 (g max) 1.708 (g min) 3.416 (h) | 4.27 | 4.27 |
| ϕ_{max} [°] | 20 | -10 | 20 (c) 40 (g) | 15 | 0 (g) - (h) | 135 (g max) 15 (g min) - (h min) | - | - |
| ϕ_{ref} [°] | -10 | 20 | -10 (c) 15 (g) | 55 | 0 (g) -25 (h) | 55 (g) 10 (h) | -30 | 0 |
| ρ [-] | 0.5 | 0.7 | 0.7 (c) 0.7 (g) | 0.7 | 0.7 (g) 0.7 (h) | 0.5 (g) 0.3 (h) | 0.5 | 0.5 |
| m [kg] | 0.241 | 0.071 | 0.113 | 0.722 | 0.45 | 0.143 | 0.248 | 0.329 |
| λ [-] | 0.81 | 0.7 | 0.54 | 0.5 | 0.44 | 0.45 | 0.5 | 0.5 |

TABLE C.1 – Paramètres des muscles de la jambe.

C.2.2 Muscles du haut du corps

La caractérisation des muscles du haut du corps est listée dans les tables C.2 et C.3. La procédure pour trouver ces valeurs est expliquée dans l'annexe D.

| | BTR | BTL | BRR | BRL | BFL | BET |
|---------------------------|------------|------------|------------|------------|------------|------------|
| F_{max} [N] | 254.9 | 254.9 | 559.3 | 559.3 | 828.4 | 1062 |
| v_{max} [l_{opt}/s] | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 |
| l_{opt} [cm] | 4.27 | 4.27 | 4.697 | 4.697 | 4.796 | 5.124 |
| l_{slack} [cm] | 4.27 | 4.27 | 4.453 | 4.453 | 5.321 | 1.281 |
| r_0 [cm] | 4.911 | 4.911 | 1.498 | 1.498 | 3.482 | 2.306 |
| ϕ_{max} [°] | 45 | -45 | -34.3 | 34.3 | 40 | -45 |
| ϕ_{ref} [°] | 0 | 0 | 17.9 | -19.9 | 7.4 | 0 |
| ρ [-] | 1 | 1 | 1 | 1 | 1 | 1 |
| m [kg] | 0.108 | 0.108 | 0.261 | 0.261 | 0.394 | 0.54 |
| λ [-] | 0.5 | 0.5 | 0.51 | 0.51 | 0.5 | 0.57 |

TABLE C.2 – Paramètres des muscles du torse

Changements apportés aux valeurs trouvées initialement :

- Les muscles SET et EFL avaient à l'origine $|\phi_{max}| > 90^\circ$. Dès lors l'approximation de Geyer ne fonctionne plus, car le cosinus change de signe. Pour régler ce problème, un ϕ_{max} inférieur à 90° a été choisi (et r_0 a été adapté en conséquence).
- Le muscle EFL était beaucoup plus puissant que EET. Cela est tout à fait réaliste pour un être humain, mais dans le cas du COMAN cela ne l'est pas, car ses avant-bras sont plus légers (il n'a pas de main). Sa force maximale a donc été légèrement revue à la baisse pour correspondre à la physiologie du robot.

| | EET | EFL | SFL | SET | SAB | SAD | SIR | SER |
|---------------------------|------------|------------|------------|------------|------------|------------|------------|------------|
| F_{max} [N] | 459.5 | 392.4 | 525.8 | 182.5 | 809 | 138.3 | 649.5 | 428.6 |
| v_{max} [l_{opt}/s] | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 |
| l_{opt} [cm] | 5.338 | 4.959 | 4.335 | 5.914 | 4.385 | 5.914 | 4.916 | 3.224 |
| l_{slack} [cm] | 5.183 | 7.212 | 4.242 | 3.8 | 4.323 | 5.6336 | 2.242 | 1.315 |
| r_0 [cm] | 1.033 | 1.612 | 1.353 | 1.751 | 1.623 | 2.092 | 1.153 | 1.153 |
| ϕ_{max} [°] | -25 | -70 | -28.5 | -70 | ± 80 | ± 20 | ± 40.3 | ± 35 |
| ϕ_{ref} [°] | -57.8 | -62.1 | -16.1 | -120 | ± 57.5 | ± 155 | ± 23.6 | 0 |
| ρ [-] | 0.81 | 0.98 | 0.66 | 0.56 | 0.68 | 0.56 | 0.66 | 0.68 |
| m [kg] | 0.244 | 0.193 | 0.226 | 0.107 | 0.352 | 0.081 | 0.253 | 0.137 |
| λ [-] | 0.32 | 0.46 | 0.57 | 0.42 | 0.57 | 0.42 | 0.58 | 0.45 |

TABLE C.3 – Paramètres des muscles du bras

C.3 Contrôleur musculaire du torse et des articulations non sagittales des bras

Les muscles du torse ainsi que des articulations non sagittales des bras ne sont pas utilisés dans ce mémoire. Cependant, le contrôle de ceux-ci a été implémenté pour pouvoir utiliser ces muscles (pour un contrôleur 3D par exemple). Il s'agit d'un contrôleur proportionnel ou proportionnel dérivé. Pour faciliter la notation, on définit les fonctions $\text{pos}(x)$ et $\text{neg}(x)$:

$$\text{pos}(x) = \begin{cases} x & \text{si } x > 0 \\ 0.01 & \text{si } x \leq 0 \end{cases} \quad (\text{C.6})$$

$$\text{neg}(x) = \begin{cases} -x & \text{si } x < 0 \\ 0.01 & \text{si } x \geq 0 \end{cases} \quad (\text{C.7})$$

$$\text{pos}(x) = \begin{cases} x & \text{si } x > 0 \\ 0.01 & \text{si } x \leq 0 \end{cases} \quad (\text{C.8})$$

$$\text{neg}(x) = \begin{cases} -x & \text{si } x < 0 \\ 0.01 & \text{si } x \geq 0 \end{cases} \quad (\text{C.9})$$

C.3.1 Bras

Les stimulations à envoyer aux muscles des articulations frontales de l'épaule :

$$S_{BTL} = \text{pos} \left(K_p \left(\theta_{\text{epaule},ref}^{\text{front}} - \theta_{\text{epaule}}^{\text{front}} \right) \right) \quad (\text{C.10})$$

$$S_{BTR} = \text{neg} \left(K_p \left(\theta_{\text{epaule},ref}^{\text{front}} - \theta_{\text{epaule}}^{\text{front}} \right) \right) \quad (\text{C.11})$$

où $K_p = 10.0$.

Les stimulations à envoyer aux muscles des articulations transversales de l'épaule :

$$S_{BET} = \text{pos} \left(K_p \left(\theta_{\text{epaule},ref}^{\text{trans}} - \theta_{\text{epaule}}^{\text{trans}} \right) \right) \quad (\text{C.12})$$

$$S_{BFL} = \text{neg} \left(K_p \left(\theta_{\text{epaule},ref}^{\text{trans}} - \theta_{\text{epaule}}^{\text{trans}} \right) \right) \quad (\text{C.13})$$

où $K_p = 5.0$.

C.3.2 Torse

Les stimulations à envoyer aux muscles des articulations sagittales du torse :

$$S_{BET} = \text{pos} \left(K_p \left(\theta_{\text{torse},ref}^{\text{sag}} - \theta_{\text{torse}}^{\text{sag}} \right) + K_d \dot{\theta}_{\text{torse}}^{\text{sag}} \right) \quad (\text{C.14})$$

$$S_{BFL} = \text{neg} \left(K_p \left(\theta_{\text{torse},ref}^{\text{sag}} - \theta_{\text{torse}}^{\text{sag}} \right) + K_d \dot{\theta}_{\text{torse}}^{\text{sag}} \right) \quad (\text{C.15})$$

où $K_p = 20$, $K_d = -2.0$.

Les stimulations à envoyer aux muscles des articulations frontales du torse :

$$S_{BTL} = \text{pos} \left(K_p \left(\theta_{torse,ref}^{front} - \theta_{torse}^{front} \right) + K_d \dot{\theta}_{torse}^{front} \right) \quad (\text{C.16})$$

$$S_{BTR} = \text{neg} \left(K_p \left(\theta_{torse,ref}^{front} - \theta_{torse}^{front} \right) + K_d \dot{\theta}_{torse}^{front} \right) \quad (\text{C.17})$$

où $K_p = 20$, $K_d = -2.0$.

Les stimulations à envoyer aux muscles des articulations transversales du torse :

$$S_{BET} = \text{pos} \left(K_p \left(\theta_{torse,ref}^{trans} - \theta_{torse}^{trans} \right) + K_d \dot{\theta}_{torse}^{trans} \right) \quad (\text{C.18})$$

$$S_{BFL} = \text{neg} \left(K_p \left(\theta_{torse,ref}^{trans} - \theta_{torse}^{trans} \right) + K_d \dot{\theta}_{torse}^{trans} \right) \quad (\text{C.19})$$

où $K_p = 5.0$, $K_d = -0.5$.

Annexe D

Hill muscles characterisation

Ce document est écrit en anglais par Nicolas Van der Noot et n'a pas été publié. Il explique la démarche suivie pour trouver la caractérisation des muscles qui agissent sur les articulations des jambes (dans les plans sagittal, latéral et transversal). Les caractéristiques des muscles ou groupes de muscles sont déterminées soit à partir de la littérature (projet où les muscles ont déjà été caractérisés), soit en utilisant des modèles musculo-squelettiques humains et en les exploitant avec le programme OpenSim. Les muscles sont ensuite mis à échelle pour tenir compte de la différence de morphologie entre le modèle utilisé et le robot COMAN.

La procédure expliquée dans ce document est suivie pour caractériser les muscles du haut du corps du COMAN.

D.1 Goal

We want to implement new muscles, using the same muscle model as the one from [29] (partially described in [30]). More specifically, we want to fully actuate the robot legs : three degrees of freedom (DOFs) in the sagittal plane (pitch), two DOFs in the lateral plane (roll) and one in the transverse plane (yaw).

To this end, we are interested in the following muscles characteristics :

- F_{max} : maximum isometric force [N]
- v_{max} : maximum shortening velocity [l_{opt}/s]
- l_{opt} : optimal fiber length [cm]
- l_{slack} : tendon slack length [cm]
- r_0 : lever arm constant [cm]
- φ_{max} : joint [deg] producing the maximum lever arm
- φ_{ref} : joint [deg] at which $l_{mtu} = l_{opt} + l_{slack}$
- ρ : pennation factor[-]
- m : mass [kg]
- λ : the fraction of Type I (slow twitch) fibers [-]

The last two parameters are only useful to compute the metabolic energy as in [7].

D.2 Muscles

For all muscles in the sagittal plane (SOL, TA, GAS, VAS, HAM, GLU, HFL), we get the characteristics from [29]. For the two muscles acting on the roll action at the hip (hip abduction

(HAB) and adduction (HAD)), we get the data from [59]. Regarding m and λ , we get the data from [70].

Finally, for the remaining muscles, we get them using the OpenSim software [20], more precisely the lower limb model developed in [4]. This model is called *LowerLimbModel2009_OneLeg* and can be downloaded at https://simtk.org/project/xml/downloads.xml?group_id=468. To exploit it, we use the OpenSim visualization environment (https://simtk.org/project/xml/downloads.xml?group_id=91), which only works on Windows.

These remaining muscles are the following (see *LowerLimbModel2009_OneLeg*) :

HER : Hip external rotator (lateral rotation) : *quadfem, piri, gem* and *gmin3*

HIR : Hip internal rotator (medial rotation) : *glmed1, iliacus, gmin1, psoas, tfl*

EVE : Eversion muscles : *perlong, perbrev, edl*

INV : Inversion muscles : *tibpost, fdl, fhl, tibant*

The following images are from http://en.wikipedia.org/wiki/Anatomical_terms_of_motion.

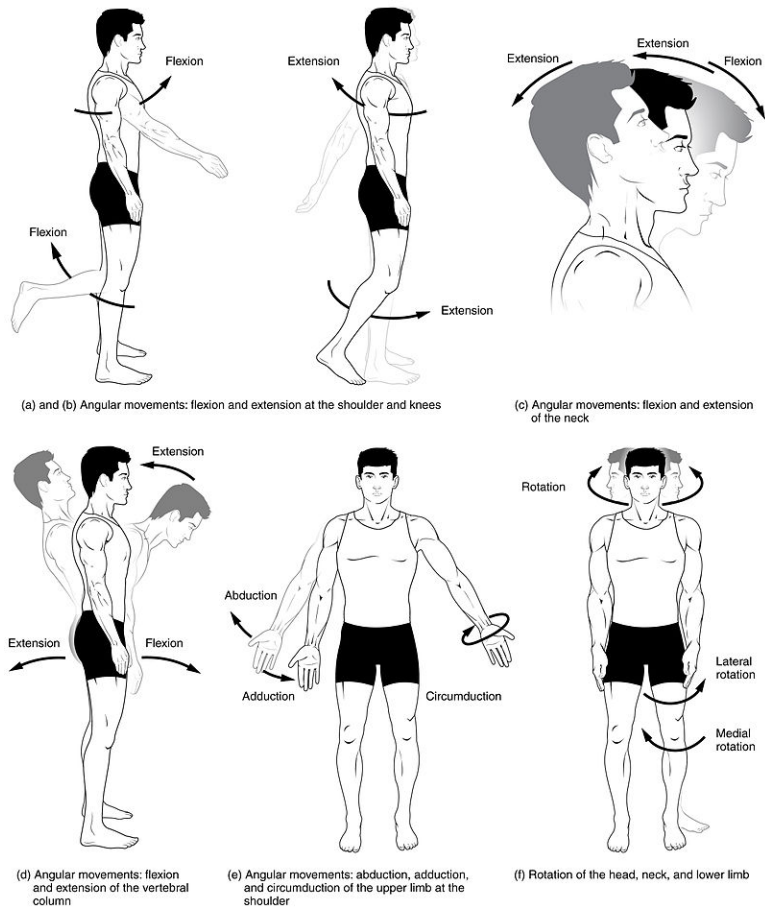


FIGURE D.1 – Body movements



FIGURE D.2 – Eversion and inversion

HER and HIR control the hip yaw angle while EVE and INV control the ankle roll one. This is of course an important simplification compared to real humans. Indeed, using these muscles, we decouple their effects on the three planes (lateral, sagittal and transverse) while, in real humans, most muscles have contributions in more than one plane.

D.3 Characteristics from OpenSim

We know explain how to find the HER, HIR, EVE and INV muscles characteristics using OpenSim.

First, the navigator of OpenSim is divided into different sections. Fortunately, the muscles are classified according to the type of motions they generate. Each muscle can appear in different classes in case it also affects other motions (most common case). In Fig. D.3, we select the four muscles of HER (*hip_extrot*) and show only these muscles in the visualization.

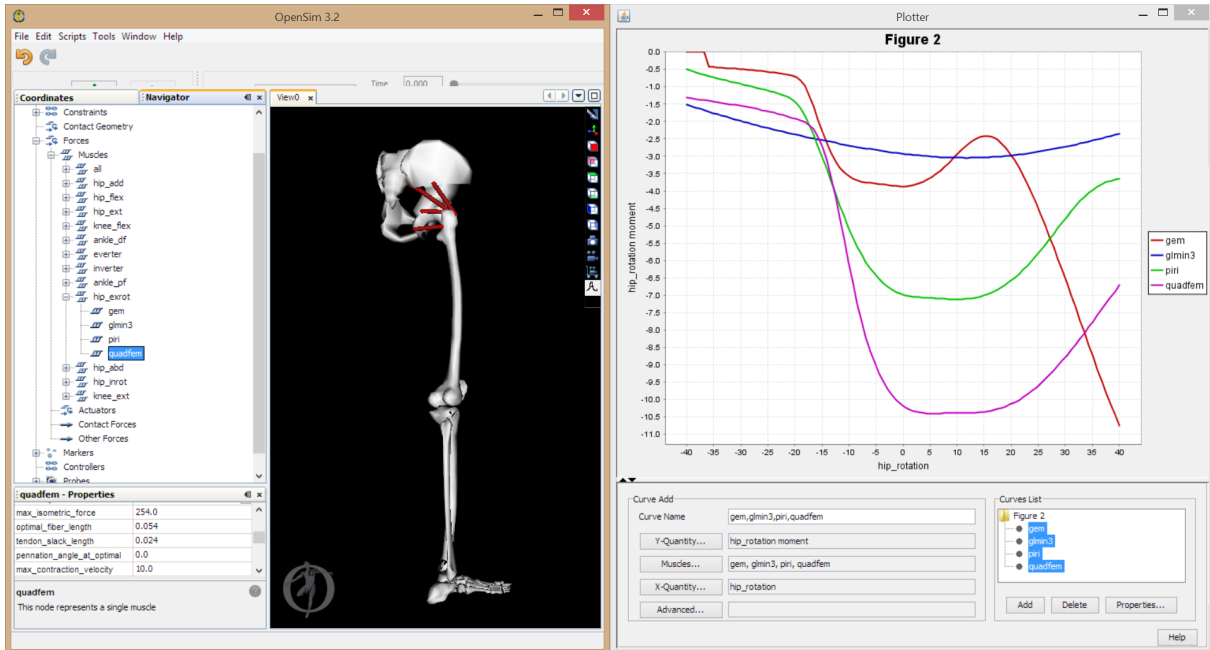


FIGURE D.3 – HER muscles torque contributions

The first thing to do is to get a *feeling* of the contribution of each muscle on the desired motion (hip external rotation in our case). To this end, we open *Plot* (in *Tools*). We let the default stimulations acting on the muscles because they seem to be the same for all muscles (to investigate). For the Y axis, we choose the moments generated at the requested degree of freedom (DOF) (*hip_rotation moment*). Then, we select the corresponding muscles. Finally, we choose the angle of interest as X coordinate (*hip_rotation* in this case).

In Fig. D.3, we see that the *quadfem* muscle seems to be the most important one for the hip external rotation movement. This is also coherent with Wikipedia : describing it (the quadratus femoris) as a strong external rotator (http://en.wikipedia.org/wiki/Quadratus_femoris_muscle). On top of that, this muscle, contrary to *gmin3* and *piri* only affect the hip external rotation motion. From Fig. D.3, we choose weight factors for the different muscles : (*quadfem* :1.0; *piri* :0.7; *gmin3* : 0.3 and *gem* : 0.3). Provided their parameters are not too different, this ponderation will not affect a lot the results.

Using these weights, we get the muscles properties of the resulting muscle group HER by applying the weighted average on their respective properties. Most of them are provided in the *Properties* section : maximum isotropic force ($max_isometric_force = F_{max}$), optimal fiber length ($optimal_fiber_length = l_{opt}$), tendon slack length ($tendon_slack_length = l_{slack}$) and complementary of the pennation factor ($pennation_angle_at_optimal = 1 - \rho$). F_{max} is an exception because we replace the weighted average by a weighted sum (i.e. we don't divide by the sum of the weights

at the end).

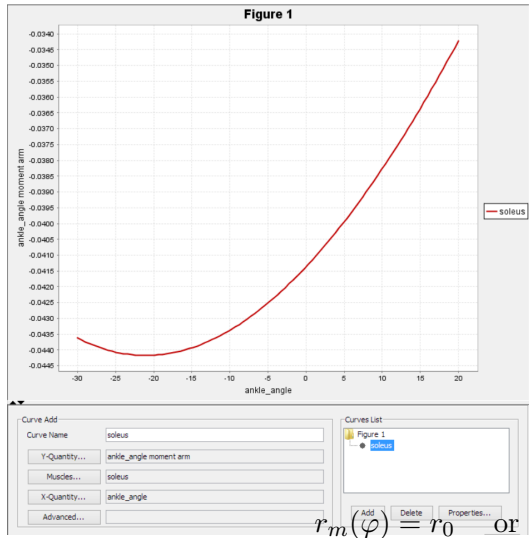


FIGURE D.4 – Soleus muscle moment arm

To get φ_{ref} , we know from [29] that it is the joint angle at which $l_{mtu} = l_{opt} + l_{slack}$. We already computed l_{opt} and l_{slack} . So, we just have to plot the muscle-tendon length to get the angle φ for which $y(\varphi) = l_{opt} + l_{slack}$. Again, a weighted average can be computed with the different muscles.

To get the mass m and the fraction of Type I (slow twitch) fibers λ , we use the same method as the one described in the supplemental material of [70] : we multiply the muscle volume by the bulk density 1060 kg/m^3 . As in [70], we approximate the muscle volume as the optimal fiber length l_{opt} multiplied by the cross sectional area. As a pressure of 25 N/cm^2 was used in [29], the cross sectional area is simply the maximum isometric force divided by 25 N/cm^2 .

To get the the fraction λ of Type I (slow twitch) fibers, we get the values from [74], applying the same weighted average as before. For the sagittal muscles, these values are already provided in [70]. Finally, for v_{max} , its value is set in [29] to $12 \text{ } l_{opt}/s$ for medium-fast twitch muscles, and to $6 \text{ } l_{opt}/s$ for slow ones. In [29] and [59], only the SOL ($\lambda = 0.81$) is considered as a slow one. The TA ($\lambda = 0.7$) is already considered as a medium-fast one. Because all the other muscles have a λ inferior to 0.7, all muscles (except the SOL) are considered as medium-fast twitch muscles, so with $v_{max} = 12 \text{ } l_{opt}/s$.

Fig. D.4 presents the soleus moment arm evolution with the ankle pitch angle. As demonstrated in [23], φ_{max} corresponds to the angle at which the moment arm (in absolute values) is maximum. This angle can be directly obtained from this Figure (-20° , corresponding to 20° in our frame references and to 110° in [29]). The lever arm constant r_0 is the corresponding y value, as can be seen in (D.2). Again, these parameters can be scaled according to the weights of the different muscles.

$$\tau_m = r_m(\varphi) F_m \quad (\text{D.1})$$

$$r_m(\varphi) = r_0 \cos(\varphi - \varphi_{max}) \quad (\text{D.2})$$

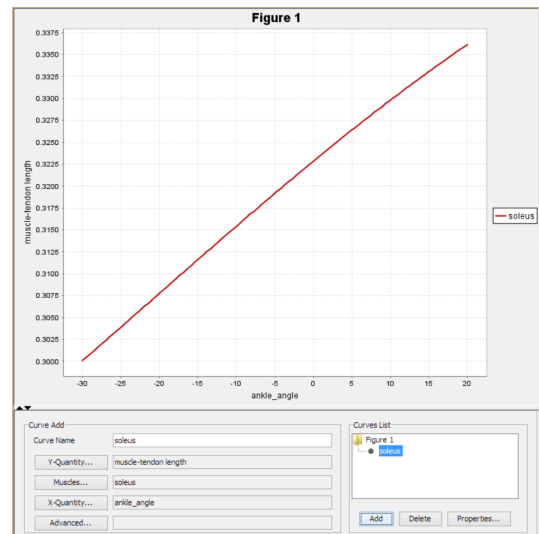


FIGURE D.5 – Soleus muscle length

D.4 Results for adult-like model

These are the results for an adult-like model of 1.8 m and 80 kg, as the one described in [29]. Fortunately, this is approximately the same model as the one from [4] (1.68 m for 82.7kg). The angles are given relative to frames attached to the different bodies. In home position (upright standing), these frames are all oriented in the same way : \hat{X} in forward direction, \hat{Y} on the left and \hat{Z} positively aligned with the gravitational axis. For non-sagittal muscles, the angles are given for the right leg.

| | SOL | TA | GAS | VAS | HAM | GLU | HFL | HAB | HAD | HER | HIR | EVE | INV |
|---------------------------|------|-----|--------------------|------|--------------------|------|------|------|------|------|------|------|------|
| F_{max} [N] | 4000 | 800 | 1500 | 6000 | 3000 | 1500 | 2000 | 3000 | 4500 | 500 | 800 | 700 | 900 |
| v_{max} [l_{opt}/s] | 6 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| l_{opt} [cm] | 4 | 6 | 5 | 8 | 10 | 11 | 11 | 9 | 10 | 4 | 8 | 5 | 5 |
| l_{slack} [cm] | 26 | 24 | 40 | 23 | 31 | 13 | 10 | 7 | 18 | 5 | 7 | 25 | 30 |
| r_o [cm] | 5 | 4 | 5 (a) 5 (k) | 6 | 5 (k) 8 (h) | 10 | 10 | 6 | 3 | 4 | 3 | 3 | 2 |
| φ_{max} [deg] | 20 | -10 | 20(a) 40 (k) | 15 | 0 (k) - (h) | - | - | - | - | - | - | -10 | 5 |
| φ_{ref} [deg] | -10 | 20 | -10 (a) 15 (k) | 55 | 0 (k) -25 (h) | -30 | 0 | -10 | -15 | 10 | -20 | -5 | -10 |
| ρ [-] | 0.5 | 0.7 | 0.7 (a) 0.7 (k) | 0.7 | 0.7 (k) 0.7 (h) | 0.5 | 0.5 | 0.7 | 1 | 1 | 0.7 | 0.7 | 0.7 |
| m [kg] | 0.68 | 0.2 | 0.32 | 2.04 | 1.27 | 0.7 | 0.93 | 1.14 | 1.91 | 0.09 | 0.27 | 0.15 | 0.19 |
| λ [-] | 0.81 | 0.7 | 0.54 | 0.5 | 0.44 | 0.5 | 0.5 | 0.5 | 0.57 | 0.5 | 0.5 | 0.57 | 0.55 |

D.5 Dynamic scaling

Dynamic scaling is presented in [6] and [57]. We present briefly the equations needed for our purpose : scaling the muscles properties for the size and the mass of the COMAN.

Suppose we have a body 1 with length L_1 and mass M_1 for which we know all the muscles characteristics. Now, we want to scale these characteristics for a body 2 having a length L_2 and a mass M_2 .

We express the time references for these two bodies as T_1 and T_2 (the two bodies can indeed have different time references, which might seem counter-intuitive at the beginning).

We will now try to express all the relations using only the international basis units : m , kg , s (A is not used as we do not play with current), so corresponding to L_i , M_i and T_i .

Gravity must be the same for the two bodies (G_1 and G_2) :

$$\frac{G_2}{G_1} = 1 = \frac{L_2/T_2^2}{L_1/T_1^2} = \frac{L_2}{L_1} \cdot \left(\frac{T_1}{T_2}\right)^2 \rightarrow \frac{T_2}{T_1} = \sqrt{\frac{L_2}{L_1}}$$

We can now compute the forces ratio :

$$\frac{F_2}{F_1} = \frac{M_2 \cdot (L_2/T_2^2)}{M_1 \cdot (L_1/T_1^2)} = \frac{M_2}{M_1}$$

We compute then the torques ratio :

$$\frac{\tau_2}{\tau_1} = \frac{L_2 \cdot F_2}{L_1 \cdot F_1} = \frac{L_2}{L_1} \cdot \frac{M_2}{M_1}$$

We then compute the velocity ratio :

$$\frac{V_2}{V_1} = \frac{L_2/T_2}{L_1/T_1} = \frac{L_2}{L_1} \cdot \frac{T_1}{T_2} = \frac{L_2}{L_1} \cdot \sqrt{\frac{L_1}{L_2}} = \sqrt{\frac{L_2}{L_1}}$$

Finally, we compute the velocity ratio normalized by the length :

$$\frac{v_2}{v_1} = \frac{1/T_2}{1/T_1} = \frac{T_1}{T_2} = \sqrt{\frac{L_1}{L_2}}$$

To get the length ratio, we compare the leg length of the human model from [29] (L_H) with the one of the COMAN (L_C). To get the mass ratio, we compare the total masses of these two models (M_H compared to M_C).

$$\frac{L_C}{L_H} = \frac{42.7 \text{ cm}}{100 \text{ cm}} = 0.427 \quad \frac{M_C}{M_H} = \frac{28.3 \text{ kg}}{80 \text{ kg}} \approx 0.354$$

Using these two ratios, we get :

$$\frac{F_C}{F_H} = \frac{M_C}{M_H} = 0.354$$

$$\frac{\tau_C}{\tau_H} = \frac{L_C}{L_H} \cdot \frac{M_C}{M_H} = 0.427 \cdot 0.354 \approx 0.151$$

$$\frac{V_C}{V_H} = \sqrt{\frac{L_C}{L_H}} = \sqrt{0.427} \approx 0.653$$

$$\frac{v_C}{v_H} = \sqrt{\frac{L_H}{L_C}} = \frac{1}{\sqrt{0.427}} \approx 1.53$$

D.6 Results for the COMAN

In the section D.5, we got the following ratios between COMAN (C) and the human model (H) :

$$\frac{L_C}{L_H} = 0.427 \quad \frac{M_C}{M_H} = 0.354 \quad \frac{F_C}{F_H} = 0.354 \quad \frac{\tau_C}{\tau_H} = 0.151 \quad \frac{V_C}{V_H} = 0.653 \quad \frac{v_C}{v_H} = 1.53$$

We now apply them on the muscles characteristics :

| | SOL | TA | GAS | VAS | HAM | GLU | HFL | HAB | HAD | HER | HIR | EVE | INV |
|---------------------------|--------|--------|------------------------|-------|------------------------|-------|-------|-------|-------|-------|-------|--------|-------|
| F_{max} [N] | 1416 | 283.2 | 531 | 2124 | 1062 | 531 | 708 | 1062 | 1593 | 177 | 283.2 | 247.8 | 318.6 |
| v_{max} [l_{opt}/s] | 9.18 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 | 18.36 |
| l_{opt} [cm] | 1.708 | 2.562 | 2.135 | 3.416 | 4.27 | 4.697 | 4.697 | 3.843 | 4.27 | 1.708 | 3.416 | 2.135 | 2.135 |
| l_{slack} [cm] | 11.102 | 10.248 | 17.08 | 9.821 | 13.237 | 5.551 | 4.27 | 2.989 | 7.686 | 2.135 | 2.989 | 10.675 | 12.81 |
| r_o [cm] | 2.135 | 1.708 | 2.135 (a) 2.135 (k) | 2.562 | 2.135 (k) 3.416 (h) | 4.27 | 4.27 | 2.562 | 1.281 | 1.708 | 1.281 | 1.281 | 0.854 |
| φ_{max} [deg] | 20 | -10 | 20(a) 40 (k) | 15 | 0 (k) - (h) | - | - | - | - | - | - | -10 | 5 |
| φ_{ref} [deg] | -10 | 20 | -10 (a) 15 (k) | 55 | 0 (k) -25 (h) | -30 | 0 | -10 | -15 | 10 | -20 | -5 | -10 |
| ρ [-] | 0.5 | 0.7 | 0.7 (a) 0.7 (k) | 0.7 | 0.7 (k) 0.7 (h) | 0.5 | 0.5 | 0.7 | 1 | 1 | 0.7 | 0.7 | 0.7 |
| m [kg] | 0.241 | 0.071 | 0.113 | 0.722 | 0.45 | 0.248 | 0.329 | 0.404 | 0.676 | 0.032 | 0.096 | 0.053 | 0.067 |
| λ [-] | 0.81 | 0.7 | 0.54 | 0.5 | 0.44 | 0.5 | 0.5 | 0.5 | 0.57 | 0.5 | 0.5 | 0.57 | 0.55 |

Of course, these values can be rounded. Indeed, they are based on rounded averages of different muscles, which were evaluated from different people. There is no single possibility, changing a bit the muscles properties (in a feasible range) is like giving the muscles from someone else.

Tests were performed on the COMAN in Robotran [55, 16]. The robot waist was constrained to a fixed position and orientation while the rest of the body was free to move. Trapezoidal stimulations were successively sent to all the non sagittal muscles. Pictures are presented in Figures D.6, D.7 and D.8.

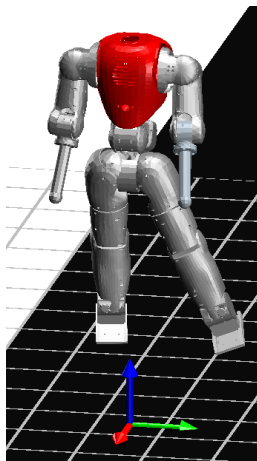


FIGURE D.6 – HAB tests

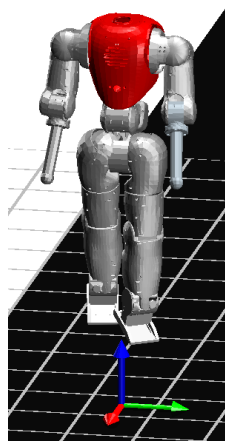


FIGURE D.7 – HER tests

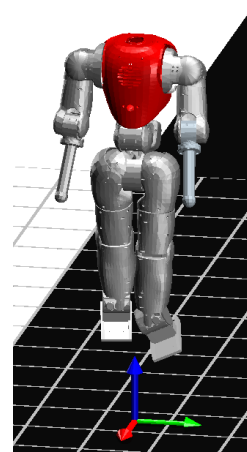


FIGURE D.8 – EVE tests

Annexe E

Résultats et analyses

E.1 Contrôleur musculaire du torse et des articulations non sagittales des bras

Comme expliqué à l'annexe C.3, le contrôleur musculaire du torse et des articulations non sagittales des bras est implémenté même s'il n'est pas utilisé dans ce projet. Il s'agit d'un contrôleur proportionnel pour le bras, et proportionnel-dérivé pour le torse. Nous utilisons une référence de position trapézoïdale afin de le tester.

L'évolution des positions des articulations frontale et transversale de l'épaule ainsi que les références trapézoïdales sont représentées sur la figure E.1 (en bas). Les stimulations envoyées aux muscles agissant sur ces articulations sont montrées sur la figure du dessus.

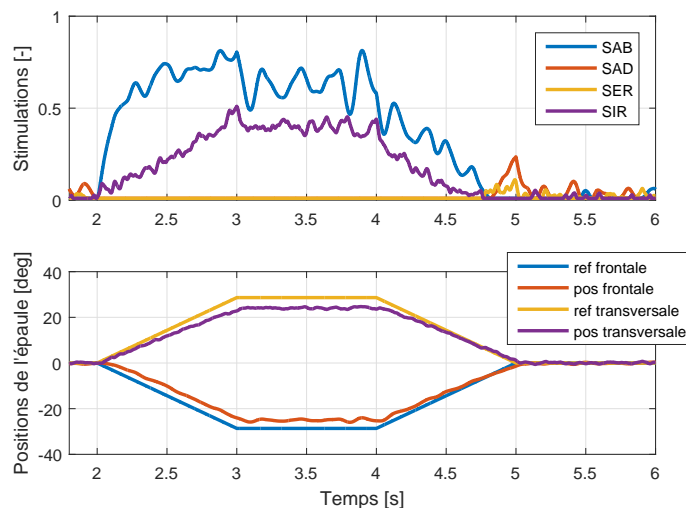


FIGURE E.1 – Évolution des positions non sagittales du bras droit pour des références trapézoïdales. Les stimulations envoyées aux muscles sont représentées à la figure du dessus.

De même, l'évolution des positions des articulations du torse (sagittale, frontale et transversale) ainsi que les références trapézoïdales sont représentées sur la figure E.2 (en bas). Les stimulations envoyées aux muscles agissant sur ces articulations sont montrées sur la figure du dessus.

On remarque que la référence de position est relativement bien suivie. Néanmoins, il persiste une petite erreur statique, mais cela n'est pas critique dans la mesure où le rôle du haut du corps est relatif par rapport à celui des jambes lors de la locomotion. On peut toutefois ajuster les

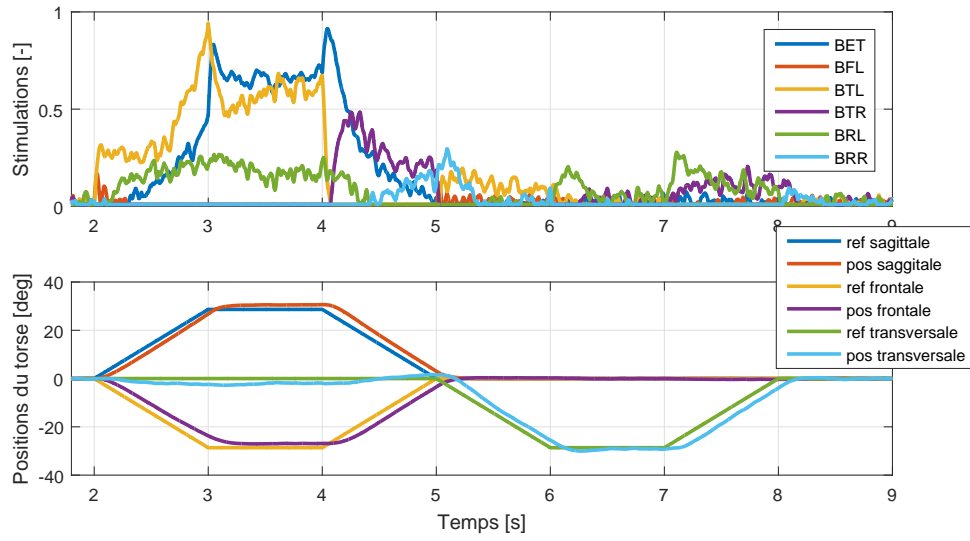


FIGURE E.2 – Évolution des positions du torse pour des références trapézoïdales. Les stimulations envoyées aux muscles sont représentées à la figure du dessus.

différents gains ainsi qu'ajouter un terme intégral pour avoir un meilleur suivi de la consigne si cela est nécessaire.

Bibliographie

- [1] R. S. Adelaar. The practical biomechanics of running. *The American journal of sports medicine*, 14(6) :497–500, 1986.
- [2] F. C. Anderson and M. G. Pandy. A dynamic optimization solution for vertical jumping in three dimensions. *Computer methods in biomechanics and biomedical engineering*, 2(3) :201–231, 1999.
- [3] F. C. Anderson and M. G. Pandy. Dynamic optimization of human walking. *Journal of biomechanical engineering*, 123(5) :381–390, 2001.
- [4] E. M. Arnold, S. R. Ward, R. L. Lieber, and S. L. Delp. A model of the lower limb for analysis of human movement. *Annals of biomedical engineering*, 38(2) :269–279, 2010.
- [5] A. Barrea and N. Van der Noot. Real-time control of a humanoid robot : Dynamic walking using a bio-inspired approach. Master’s thesis, École polytechnique de Louvain (EPL), 2013.
- [6] A. Bejan and J. H. Marden. Unifying constructal theory for scale effects in running, swimming and flying. *Journal of Experimental Biology*, 209(2) :238–248, 2006.
- [7] L. J. Bhargava, M. G. Pandy, and F. C. Anderson. A phenomenological model for estimating metabolic energy consumption in muscle contraction. *Journal of biomechanics*, 37(1) :81–88, 2004.
- [8] R. Blickhan. The spring-mass model for running and hopping. *Journal of biomechanics*, 22(11-12) :1217–1227, 1989.
- [9] R. Blickhan and R. Full. Similarity in multilegged locomotion : bouncing like a monopode. *Journal of Comparative Physiology A*, 173(5) :509–517, 1993.
- [10] D. M. Bramble and D. E. Lieberman. Endurance running and the evolution of homo. *Nature*, 432(7015) :345–352, 2004.
- [11] H. M. Co. Studies of leg/foot functions of the robot. <http://world.honda.com/ASIMO/P3/technology>, 2006.
- [12] H. M. Co. The honda humanoid robot asimo. <http://world.honda.com/ASIMO>, 2016.
- [13] S. H. Collins, P. G. Adamczyk, and A. D. Kuo. Dynamic arm swinging in human walking. *Proceedings of the Royal Society of London B : Biological Sciences*, page rspb20090664, 2009.
- [14] S. Coros, P. Beaudoin, and M. van de Panne. Generalized biped walking control. In *ACM Transactions on Graphics (TOG)*, volume 29, page 130. ACM, 2010.
- [15] H. Dallali, M. Mosadeghzad, G. A. Medrano-Cerda, N. Docquier, P. Kormushev, N. Tsagarakis, Z. Li, and D. Caldwell. Development of a dynamic simulator for a compliant humanoid robot based on a symbolic multibody approach. In *Mechatronics (ICM), 2013 IEEE International Conference on*, pages 598–603. IEEE, 2013.
- [16] H. Dallali, M. Mosadeghzad, G. A. Medrano-Cerda, N. Docquier, P. Kormushev, N. Tsagarakis, Z. Li, and D. Caldwell. Development of a dynamic simulator for a compliant humanoid robot based on a symbolic multibody approach. In *Mechatronics (ICM), 2013 IEEE International Conference on*, pages 598–603. IEEE, 2013.

- [17] A. Dasgupta and Y. Nakamura. Making feasible walking motion of humanoid robots from human motion capture data. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1044–1049. IEEE, 1999.
- [18] A. De Coninck. Bio-inspired walking controllers for a variety of robots having different sacles. Master’s thesis, École polytechnique de Louvain (EPL), 2016.
- [19] M. de Lasa, I. Mordatch, and A. Hertzmann. Feature-based locomotion controllers. In *ACM Transactions on Graphics (TOG)*, volume 29, page 131. ACM, 2010.
- [20] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen. Opensim : open-source software to create and analyze dynamic simulations of movement. *Biomedical Engineering, IEEE Transactions on*, 54(11) :1940–1950, 2007.
- [21] S. L. Delp, J. P. Loan, M. G. Hoy, F. E. Zajac, E. L. Topp, and J. M. Rosen. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *Biomedical Engineering, IEEE Transactions on*, 37(8) :757–767, 1990.
- [22] S. A. Dugan and K. P. Bhat. Biomechanics and analysis of running gait. *Physical medicine and rehabilitation clinics of North America*, 16(3) :603–621, 2005.
- [23] F. Dzeladini. *Implementation of a Human Feedback-based Locomotion and its Control by means of a Feedforward Component inspired by Central Pattern Generators*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 2013.
- [24] T. Erez and E. Todorov. Trajectory optimization for domains with contacts using inverse dynamics. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4914–4919. IEEE, 2012.
- [25] A. A. Faisal, L. P. Selen, and D. M. Wolpert. Noise in the nervous system. *Nature reviews neuroscience*, 9(4) :292–303, 2008.
- [26] J. Falconer. This humanoid robot gets pushed around but stays on its feet. <http://spectrum.ieee.org/automaton/robotics/humanoids/iit-coman-humanoid-robot>, 2013.
- [27] S. Faraji, S. Pouya, R. Moeckel, and A. J. Ijspeert. Compliant and adaptive control of a planar monopod hopper in rough terrain. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 4818–4825. IEEE, 2013.
- [28] R. J. Full and D. E. Koditschek. Templates and anchors : neuromechanical hypotheses of legged locomotion on land. *Journal of Experimental Biology*, 202(23) :3325–3332, 1999.
- [29] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(3) :263–273, 2010.
- [30] H. Geyer, A. Seyfarth, and R. Blickhan. Positive force feedback in bouncing gaits? *Proceedings of the Royal Society of London B : Biological Sciences*, 270(1529) :2173–2183, 2003.
- [31] S. R. Hamner, A. Seth, and S. L. Delp. Muscle contributions to propulsion and support during running. *Journal of biomechanics*, 43(14) :2709–2716, 2010.
- [32] N. Hansen. The CMA Evolution Strategy : A Tutorial. *ArXiv e-prints*, Apr. 2016.
- [33] K. Hase and N. Yamazaki. Computer simulation study of human locomotion with a three-dimensional entire-body neuro-musculo-skeletal model. i. acquisition of normal walking. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 45(4) :1040–1050, 2002.
- [34] F. Heremans, N. Van der Noot, A. J. Ijspeert, and R. Ronsse. Bio-inspired balance controller for a humanoid robot. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 441–448. IEEE, 2016.
- [35] A. Hill. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London B : Biological Sciences*, 126(843) :136–195, 1938.

- [36] K. R. Holzbour, W. M. Murray, and S. L. Delp. A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Annals of biomedical engineering*, 33(6) :829–840, 2005.
- [37] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa. The 3d linear inverted pendulum mode : A simple modeling for a biped walking pattern generation. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 1, pages 239–246. IEEE, 2001.
- [38] S. Kajita, S. Sakka, H. Hirukawa, K. Harada, and K. Yokoi. *Introduction à la commande des robots humanoïdes : De la modélisation à la génération du mouvement*. Springer Science & Business Media, 2009.
- [39] K. Kaneko, F. Kanehiro, S. Kajita, K. Yokoyama, K. Akachi, T. Kawasaki, S. Ota, and T. Isozumi. Design of prototype humanoid robotics platform for hrp. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 3, pages 2431–2436. IEEE, 2002.
- [40] I. Kato. Development of waseda robot—the study of biomechanisms at kato laboratory, 1991.
- [41] J.-Y. Kim, I.-W. Park, and J.-H. Oh. Experimental realization of dynamic walking of the biped humanoid robot khr-2 using zero moment point feedback and inertial measurement. *Advanced Robotics*, 20(6) :707–736, 2006.
- [42] S.-H. Lee, E. Sifakis, and D. Terzopoulos. Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)*, 28(4) :99, 2009.
- [43] R. A. Mann and J. Hagy. Biomechanics of walking, running, and sprinting. *The American journal of sports medicine*, 8(5) :345–350, 1980.
- [44] M. Mistry, J. Buchli, and S. Schaal. Inverse dynamics control of floating base systems using orthogonal decomposition. In *IEEE Int. Conf. on Robotics and Automation*, pages 3406–3412, 2010.
- [45] I. Mordatch, M. De Lasa, and A. Hertzmann. Robust physics-based locomotion using low-dimensional planning. In *ACM Transactions on Graphics (TOG)*, volume 29, page 71. ACM, 2010.
- [46] J. Morimoto, G. Cheng, C. G. Atkeson, and G. Zeglin. A simple reinforcement learning algorithm for biped walking. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 3, pages 3030–3035. IEEE, 2004.
- [47] T. F. Novacheck. The biomechanics of running. *Gait & posture*, 7(1) :77–95, 1998.
- [48] D. E. Orin and A. Goswami. Centroidal momentum matrix of a humanoid robot : Structure and properties. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 653–659. IEEE, 2008.
- [49] J. Park. Synthesis of natural arm swing motion in human bipedal walking. *Journal of biomechanics*, 41(7) :1417–1426, 2008.
- [50] R. R. Playter and M. H. Raibert. Control of a biped somersault in 3d. In *Intelligent Robots and Systems, 1992., Proceedings of the 1992 IEEE/RSJ International Conference on*, volume 1, pages 582–589. IEEE, 1992.
- [51] N. S. Pollard, J. K. Hodgins, M. J. Riley, and C. G. Atkeson. Adapting human motion for the control of a humanoid robot. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 2, pages 1390–1397. IEEE, 2002.
- [52] A. project. Compliant humanoid platform (coman). <https://www.amarsi-project.eu/coman>, 2013.
- [53] M. H. Raibert. *Legged robots that balance*. MIT press, 1986.
- [54] M. H. Raibert and J. K. Hodgins. Animation of dynamic legged locomotion. In *ACM SIGGRAPH Computer Graphics*, volume 25, pages 349–358. ACM, 1991.

- [55] J.-C. Samin and P. Fiset. *Symbolic modeling of multibody systems*. Springer, 2004.
- [56] K. Sasaki and R. R. Neptune. Differences in muscle function during walking and running at the same speed. *Journal of biomechanics*, 39(11) :2005–2013, 2006.
- [57] A. Schepelmann, H. Geyer, and M. Taylor. Development of a testbed for robotic neuromuscular controllers. In *Robotics : Science and Systems*, 2012.
- [58] E. Sifakis, I. Neverov, and R. Fedkiw. Automatic determination of facial muscle activations from sparse motion capture marker data. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 417–425. ACM, 2005.
- [59] S. Song and H. Geyer. Generalization of a muscle-reflex control model to 3d walking. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7463–7466. IEEE, 2013.
- [60] S. Song and H. Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of physiology*, 593(16) :3493–3511, 2015.
- [61] S. Song and H. Geyer. Regulating speed in a neuromuscular human running model. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 217–222. IEEE, 2015.
- [62] S. Sueda, A. Kaufman, and D. K. Pai. Musculotendon simulation for hand animation. In *ACM Transactions on Graphics (TOG)*, volume 27, page 83. ACM, 2008.
- [63] G. Taga, Y. Yamaguchi, and H. Shimizu. Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. *Biological cybernetics*, 65(3) :147–159, 1991.
- [64] R. Tedrake, T. W. Zhang, and H. S. Seung. Stochastic policy gradient reinforcement learning on a simple 3d biped. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2849–2854. IEEE, 2004.
- [65] N. Van der Noot. Hill muscles characterisation. document non publié, avril 2015.
- [66] N. Van der Noot, L. Colasanto, A. Barrea, J. van den Kieboom, R. Ronsse, and A. J. Ijspeert. Experimental validation of a bio-inspired controller for dynamic walking with a humanoid robot. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 393–400. IEEE, 2015.
- [67] N. Van der Noot, A. J. Ijspeert, and R. Ronsse. Biped gait controller for large speed variations, combining reflexes and a central pattern generator in a neuromuscular model. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 6267–6274. IEEE, 2015.
- [68] M. Vukobratović and B. Borovac. Zero-moment point—thirty five years of its life. *International Journal of Humanoid Robotics*, 1(01) :157–173, 2004.
- [69] J. M. Wang, D. J. Fleet, and A. Hertzmann. Optimizing walking controllers for uncertain inputs and environments. In *ACM Transactions on Graphics (TOG)*, volume 29, page 73. ACM, 2010.
- [70] J. M. Wang, S. R. Hamner, S. L. Delp, and V. Koltun. Optimizing locomotion controllers using biologically-based actuators and objectives. *ACM transactions on graphics*, 31(4), 2012.
- [71] P. M. Wensing. *Optimization and Control of Dynamic Humanoid Running and Jumping*. PhD thesis, The Ohio State University, 2014.
- [72] A. Witkin and M. Kass. Spacetime constraints. *ACM Siggraph Computer Graphics*, 22(4) :159–168, 1988.
- [73] J. K. H. W. L. Wooten. Animating human athletics. 1995.
- [74] G. Yamaguchi, A. Sawa, D. Moran, M. Fessler, and J. Winters. A survey of human musculotendon actuator parameters. *Multiple muscle systems : Biomechanics and movement organization*, pages 717–773, 1990.

- [75] G. T. Yamaguchi and F. E. Zajac. A planar model of the knee joint to characterize the knee extensor mechanism. *Journal of biomechanics*, 22(1) :1–10, 1989.
- [76] K. Yamane and Y. Nakamura. Dynamics filter-concept and implementation of online motion generator for human figures. *IEEE transactions on robotics and automation*, 19(3) :421–432, 2003.
- [77] K. Yin, K. Loken, and M. van de Panne. Simbicon : Simple biped locomotion control. In *ACM Transactions on Graphics (TOG)*, volume 26, page 105. ACM, 2007.

