Contribution to the Control of Biped Robots

Sylvain Finet

THÈSE DE DOCTORAT

de l'Université de recherche Paris Sciences et Lettres
PSL Research University

Préparée à MINES ParisTech

Contribution à la commande des robots bipèdes
Contribution to the Control of Biped Robots

Ecole doctorale SMI 432

Spécialité Mathématique et Automatique

COMPOSITION DU JURY :

M. Tarek HAMEL
Université de Nice, Président

M. Aaron AMES
Caltech, Rapporteur

M. Gabriel ABBA
ENIM, Rapporteur

M. Nicolas MANSARD
LAAS, Rapporteur

M. Laurent PRALY
Mines ParisTech, Membre du jury

M. Matthieu MASSELIN
Société Wandercraft, Membre du jury

Soutenue par Sylvain FINET
le 07 juin 2017

Dirigée par **Laurent PRALY**

MINES ParisTech | PSL RESEARCH UNIVERSITY PARIS

Résumé

Cette thèse porte sur le développement de lois de commande pour la marche des robots bipèdes. Le sous actionnement engendré par le basculement, volontaire ou involontaire, du pied en appui sur le sol représente une difficulté majeure. Nous abordons ce problème par l'étude de robots plans avec pieds ponctuels.

La première partie de la thèse est une compilation des informations issues de la littérature que nous avons jugées intéressantes. Nous traitons dans un premier temps de la modélisation adoptée, puis effectuons une revue des différentes méthodes existantes, et présentons la mise en oeuvre expérimentale de l'une d'entre elle : la méthode HZD.

Dans une deuxième partie, nous procédons à une étude de la dissipation relative de l'énergie cinétique du robot lorsque le pied impacte le sol. Nous utilisons les résultats issus de cette étude pour planifier des trajectoires de marche dissipant peu d'énergie. De telles trajectoires ont a priori le mérite de préserver la structure du robot et de générer moins de bruit. A contrario, des trajectoires dissipant la majorité de l'énergie du robot sont utilisées pour un arrêt rapide. Une étude numérique a montré que ces résultats sont robustes à des incertitudes de modèle.

Enfin, dans une dernière partie, afin de compenser les difficultés liées au sous actionnement, nous proposons d'utiliser le degré de liberté supplémentaire offert par un changement de l'échelle de temps dans les équations de la dynamique (Time Scaling) pour la classe de robots considérée. En utilisant par ailleurs un changement de coordonnées et de feedback, nous dérivons de nouvelles formes normales exactes et approximatives.

Mots clés : Robot bipède - Planification de trajectoires- Dynamique des zéros - Impact - Time Scaling

Abstract

This thesis addresses the general problem of the walking control of biped robots. The foot of the robot in contact with the ground may tip over and cause the robot to be undercatuated. This is a major difficulty in term of control. This problem is addressed by considering planar biped robots with point feet.

In a first part, we present a standard way of modeling such systems, a litterature review of the existing methods, and then report experimental results of the walking control of a biped robot using the HZD method.

In a second part, we perform an analytic and numeric study of the relative kinetic energy dissipation when the foot of the robot impacts the ground. Using this study, we design trajectories with low energy dissipation at impact, which a priori result in gaits preserving the hardware of the robot and causing less noise. On the contrary, trajectories dissipating almost all the kinetic energy are used to quickly stop the robot.

Finally, in an attempt to alleviate the burden due to underactuation, we propose to investigate the additional degree of freedom provided, in the control design, by a change of time scale in the dynamic equations (Time-Scaling) for the considered class of biped robots. Using feedback transformations, we derive new exact and approximative normal forms.

Keywords: Biped Robot - Motion Planning - Zero dynamics - Impact - Time Scaling

Remerciements

Je tiens à remercier en premier lieu mon directeur de thèse, Laurent Praly, pour son suivi et ses conseils réguliers. Quel régal que de pouvoir travailler avec une personne à l'imagination si fertile, et qui maîtrise avec tant d'aisance les mathématiques. J'aurai beaucoup appris à son contact, en particulier une extrême rigueur et la remise en cause d'affirmations qui paraissent évidentes, mais qui en réalité ne sont qu'approximatives, voire fausses. Il a eu aussi beaucoup de courage et de la patience pour se lancer dans ce sujet qui était nouveau pour nous. Nous avons passé beaucoup de temps à discuter, débattre, gribouiller, raturer de nombreuses pages, dans la bonne humeur et dans un esprit scientifique de respect mutuel. De grands moments ! Merci à l'équipe du CAS, en particulier à Nicolas Petit de m'avoir fait confiance et de m'avoir proposé cette thèse. Du côté de Wandercraft, merci à Matthieu qui a su me donner toute la liberté nécessaire pour mener à bien cette thèse.

Cette thèse a eu une dimension tout à fait exceptionnelle grâce à l'ambiance incroyable et unique qui règne à Wandercraft. Que de bons moments passés avec des gens que je considère plus comme des amis que des collègues. Je retiendrai particulièrement : les Wanderpots (jusqu'à un par soir à la belle époque !), les apéros du lundi, les Wanderskis, les sorties escalade, le séjour au Cap Ferret, le séjour à Séville, la Loire à vélo, le parc Astérix, une balade dans les (vraies) catacombes, les soirées poker, la pendaison de crémaillère (la meilleure des soirées que j'ai connue !), et j'en passe ... Merci à vous tous.

Merci à Jean Lévine et Frédéric Jean qui ont su être disponibles pour répondre à toutes mes interrogations concernant la géométrie différentielle. Je remercie Aaron Ames, Gabriel Abba et Nicolas Mansard qui m'ont fait l'honneur d'accepter d'être les rapporteurs de cette thèse, et Tarek Hamel d'être le président du jury. Je remercie aussi très chaleureusement Jessy Grizzle et Christine Chevallereau d'avoir toujours accepté de discuter ouvertement, et de répondre rapidement et clairement aux questions que je leur ai posées tout au long de ma thèse. Quel plaisir d'avoir pu être en contact avec des scientifiques aussi brillants et humbles.

Enfin, je remercie ma famille pour tout le soutien et l'amour qu'elle m'a donnée. Merci à Myriam, qui par son amour et sa gentillesse, a illuminé cette fin de thèse.

Contents

Preamble

My PhD was funded by Wandercraft, a French startup company founded in 2012, and headed by three newly graduated engineers[1]. Today, a team of more than 20 engineers and doctors are working hard for the design, control, and marketing of a novel medical active lower-limb exoskeleton for rehabilitation. Active lower-limb exoskeletons are anthropomorphic robotic devices that fit around the legs of its user. The exoskeleton of Wandercraft aims at recovering the ability to walk for the people who lost the control of their lower limbs, such as paraplegics or myopathics. Contrary to the currently existing exoskeletons, it will be able to stabilize and carry the user at the average human walking speed, without requiring crutches or stabilizing mechanisms. Originally designed to be used in rehabilitation centers, the exoskeleton will be next commercialized to individual customers.

When I arrived at Wandercraft, the company had no employees and was located in a small room in a startup incubator. The faith that the founders had in this project could have seemed disproportionate. They had strictly no experience, but were willing to change the life of people with disabilities. Nonetheless, the appeal and the technical challenges of the project convinced me to join this new adventure.

It was not possible to define an outline of the work to be done, since we started from scratch. Obviously, the ultimate goal was to develop a control law for the exoskeleton. But, at that time, the exoskeleton did not exist, and we did not know how to achieve this ambitious objective.

As a first step, we decided to look at the rich existing literature on robotic bipedal locomotion. We believed that these techniques could be applied to the control of exoskeletons. We especially noticed the method of virtual constraints and hybrid zero dynamics (HZD method), which clearly and rigorously explains how to design controllers for underactuated bipedal locomotion. Two of their main contributors, Professor Jessy Grizzle[2] and Christine Chevallereau[3] always kindly answered our questions. In order to gain more experience in this method, Wandercraft designed a planar robot with point feet[4], called VS, an almost replica of the RABBIT robot, on which the HZD method was successfully tested for the

[1] Nicolas Simon, Alexandre Boulanger, Matthieu Masselin. Jean-Louis Constanza has recently joined the head as business manager.

[2] University of Michigan (USA)

[3] IRRCyN, Nantes (France)

[4] *Planar* means that the robot is equipped with a lateral stabilizing mechanism to facilitate its control. Then, the controller has to stabilize the robot in the sagittal plane "only" (plane dividing the robot into left and right halves). *Point feet* means that the robot has no feet.

first time in 2002.

I was tasked with the design and the implementation of HZD control laws on the VS robot. It took 6 months to get a continuous robust walk on the VS robot. I especially encountered repetitive hardware problems which slowed down the experiments: cable faults, breakdown of the treadmill on which the robot evolves, deterioration of the contact sensors in the feet. But finally, successful experiments were realized.

As for the theoretical work, as a first step, inspired by the HZD method, I focused on planar robots with point feet, since these models, which are simpler than tridimensional robots, are well suited for beginning a study on biped robots and the development of new control strategies. Some new ideas to control this class of robots are introduced in this manuscript. Unfortunately, due to a lack of time, I did not implement them on the real VS robot, and I was not able to exhibit new results for tridimensional biped robots.

Even though this is the core of the project of Wandercraft, the control of a lower-limb exoskeleton was not explicitly addressed. Firstly, some maturity had to be gained in the probably simpler problem of controlling a standard biped robot. Indeed, the problem of the interaction of a user with the exoskeleton is not obvious. The question raises whether the user will be able to learn how to help to stabilize the exoskeleton by an appropriate motion of its torso and upper limbs, or if the exoskeleton will have to compensate for the perturbations exerted by the user. This may depend on the pathology.

This PhD work was guided by the two-fold goal of being as useful as possible for the development of Wandercraft, and producing innovative and releasable scientific results. I am not mincing my words when I say that it was not an easy task to be innovative in a field that has been researched by brilliant researchers for decades, and starting from scratch. Most of the results presented in this manuscript were obtained during the last year of the PhD. At the end of the day, I hope that this dissertation brings a little contribution to the field.

Introduction

Context

The general problem under consideration in this thesis is the walking control of biped robots. Bipedal walking is a way of locomotion that consists in moving forward the feet alternately so that there is always one foot on the ground. Humans are generally considered as the most agile biped walkers in the animal world. Human walking gaits are *robust*, *energy-efficient* and *dynamic*. *Robust* means that humans are able to handle perturbations, such as an external push or a slippage of the foot. *Energy-efficient* means that the walk consumes little energy. *Dynamic* means that human is capable of performing fast walking motions. This is achieved by the fact that he is falling during a part of the walking cycle. This happens for instance when the foot rolls on the ground. Actually human walking can be seen as "controlled falling".

Robots are machines operating automatically. They are equipped with sensors to perceive their own state (position and velocity of the joints, absolute orientation...) and to sense their environment (contact sensor, camera...). The information provided by the sensors is processed by an algorithm embedded in a computer. The algorithm makes a decision on the actions that the robot must realize. Orders are then sent by the computer to the actuators of the robot to perform the desired task. Biped robots are robots equipped with two actuated legs.

The design and the control of biped robots have raised a strong interest in the past decades, since it is believed that they will ultimately be able to realize some tasks that others mobile robots, like wheeled robots, have difficulties to realize: moving on a rough terrain, crossing an obstacle, climbing a ladder or more generally evolving in a human environment such as a house. The applications of robot bipedal locomotion are numerous. Biped robots capable to evolve in real and unknown environments could assist humans in everyday life. They could work in hazardous areas such as in a fire disaster or a radioactive place. The use of anthropomorphic robotic devices that fit around the legs of its user, called lower limb exoskeletons, could restore the mobility of paraplegics.

Brief State-of-the-art

Researches on biped robots started at the beginning of the 1970s with the works of Miomir Vukobratovic (Mialjo Pupin Institude, Belgrade) on the design and the

control of a leg exoskeleton for paraplegics [121]. The exoskeleton was capable of providing a walking motion to the legs of a user, but the user had to use crutches to keep balanced. These researches highlighted the important role played by the center of pressure of the ground reaction forces, called ZMP, in the balance of bipeds. A necessary condition so that the foot/feet in contact with the ground do not tip over is that the ZMP stays strictly inside the convex hull of the contact points with the ground (the support polygon). This lead to the emergence of a method called *flat foot walking*, or *ZMP walking*, that consists in designing a walking controller constraining the robot to always keep its feet flat on the ground. Since the feet are flat on the ground and do not slip, the robot is then entirely actuated, i.e. the number of actuators is greater than or equal to the number of degrees of freedom, which greatly facilitates its control. But, this requires high ankle torques. This approach has been successfully implemented for the first time in 1984 on the japanese robot WL-10RD by Ichiro Kato et al. Later, Kajita et al. introduced a popular approach for generating flat foot walking gaits online using the linear inverted pendulum as an approximative model of the robot [59]. The ZMP method has been widely used on humanoid robots (ASIMO [54], HUBO [123], HRP2 [61], NAO [44]...). Nonetheless, the absence of foot rotation limits the possibility of getting fast and energy-efficient gaits. Additionally, the robustness to disturbances of the walk is limited since the robot falls down when the feet tip over.

Marc Raibert (MIT) pioneered dynamic legged motion in the early 1980s. He first designed one legged robots. These "hoppers" were kept balanced by continuously hopping and calculating where to place the leg onto the ground between hops to maintain their balance. He generalized this new strategy, called *foot placement* to multi-legged robots. His robots were able to realize agile motions such as back flips [94]. Raibert founded the company Boston Dynamics in 1992, which is known worldwide for its impressive agile quadruped robot BigDog and humanoid robot ATLAS.

In 1988, Tad McGeer highlighted that biped mechanisms with no actuators are able to naturally walk down slopes by converging to a periodic walking (limit cycle walking) thanks to the counteractive action of the gravity (gain of energy) and of the impact of the feet onto the ground (loss of energy). They are called passive walkers [73]. Starting from this observation, in the early 2000s, Christine Chevallereau (IRCCyN, Nantes) and Jessy W. Grizzle (University of Michigan) introduced a new approach so that actuated biped robots be controlled to converge to a periodic walking cycle. It is called *method of virtual constraints and hybrid zero dynamics (HZD method)* [127]. Contrary to the previous approaches, which use heuristics and simplified models of the robot, the HZD method uses a complete model and allows the design of walking trajectories for underactuated robots that are provably stable. Underactuated means that the robot has more degrees of freedom than actuators. Then, some degrees of freedom may describe an unstable trajectory (non minimum phase zero dynamics). The HZD method especially addresses the control of point foot robots, that is robots with no feet. The use of point feet robots makes the ZMP heuristics not applicable and requires the

design of a controller that cope with the underactuation resulting from the non actuation of the leg contact with the ground. The method was first tested in 2002 on the planar[5] robot with point feet RABBIT. It has since then been extended and successfully tested on real tridimensional biped robots with point feet (MARLO) and with actuated feet (DURUS).

All these methods do not require to sense the environment to make the robot walk (blind robot). But, laser range finders or cameras can be used to improve the performances of the gait. Today, the community of biped robot locomotion is very active and major improvements have been made over the past few years. Several biped robots have been tested recently outdoor: ATLAS, ATRIAS, Cassie... Here are some videos illustrating the current state-of-the-art in robotic bipedal locomotion:

- ZMP Method. ASIMO robot (HONDA, 2014).
  `https://www.youtube.com/watch?v=_kzgk4Rnpqc`.
- Method of Foot Placement. ATLAS robot (Boston Dynamics, February 2016).
  `https://www.youtube.com/watch?v=rVlhMGQgDkY`.
- HZD Method. DURUS robot (Georgia Tech, July 2016).
  `https://www.youtube.com/watch?v=1fC7b2LjVW4`.

These results are quite impressive, but there is still work to do to get biped robots walking as well as humans can.

Issues in Robotic Bipedal Locomotion

The control of biped robots is challenging and several issues still have to be addressed.

A first issue concerns the mecatronical design. The structure of biped robots is far less advanced than the humans one. While robots are equipped with a few actuators, hundreds of muscles act together during human walking. Muscles store and release energy during the walk. As a consequence, humans have greater power-to-mass ratios and more energy efficient walks than any existing biped robot. The design of control algorithms for biped robots is constrained by these inherent physical limitations. Things get even more complicated for commercialized robots, where the price should be as low as possible.

Additionally, in spite of more than 40 years of research, while walking seems to be easy and natural for humans, the mechanisms of bipedal locomotion are not fully understood. The analysis of biped locomotion is rendered difficult by the fact that the dynamics of biped robots are complex and nonlinear. See the five pages-length Appendix E of [127] that gives the equations of motion of a (yet

[5]This means that the robot was equipped with a lateral stabilizing mechanism to facilitate its control. Then, the controller had to stabilize the robot in the sagittal plane "only" (plane dividing the robot into left and right halves).

simple) planar five-link walker with point feet. Additionally, biped robots are subject to an impact, which adds complexity to the modeling and to the control. In spite of this, the researches mentioned above highlighted some aspects of biped locomotion. The dynamics of a biped are similar to those of an inverted pendulum. The control of the ZMP and a proper placement of the foot on the ground plays an important role in the balance of bipeds. Walking can be viewed as the convergence of the walker to a periodic limit cycle. But, to date, no control method unifies these aspects of biped locomotion.

During walking, the robot may become, voluntarily or involuntarily, underactuated. Then, controllers must cope with underactuation. In addition to this difficulty, the design of such controllers is constrained by the fact that the computational capacities of computers are limited. Thus, for the time being (but things should change with computer performance improvement), any sophisticated control algorithm cannot be used, especially to replan online the trajectory of the robot.

Objectives and Thesis Outline

Since underactuation is inherent to bipedal locomotion and that current controllers do not result in gaits which are as agile and energy efficient as human gaits, in this PhD work, we address the control of underactuated biped robots. Following the HZD method, we especially consider planar one degree underactuated biped robots with point feet, since this class of system is simpler than tridimensional underactuated biped robots, but sufficiently complex to embody some problems due to underactuation in bipedal locomotion. We deemed necessary to begin with these simpler models, in particular for developing some new ideas. The study of point feet robots may be confusing, but the results obtained for such robots can be translated to robots with feet. Indeed, the dynamic behavior of a robot the foot of which is rolling on the ground is close to the one of a point feet robot. Then, in this thesis, we address the following issue:

How to design control algorithms for obtaining a robust and energy efficient walking gait on planar one degree underactuated biped robots with point feet?

We addressed the problem in three steps:

- Reviewing the literature of robot biped locomotion.
- Testing some existing methods on a real testbed.
- Proposing improvements to the state-of-the-art.

The manuscript follows this approach. In Chapter 1, we present a standard framework, especially described in [50], to model biped robots as hybrid systems,

that is systems ruled by continuous and discrete dynamics. Then, in Chapter 2 we give a review of the existing methods for the control of biped robots. We identified three major methods: the ZMP method, the method of foot placement, and the method of virtual constraints and hybrid zero dynamics (HZD method). Some works do not claim to belong to one of these three categories. And many contributions have been made across the years, which may introduce confusion to know which approach offers the best performances. We give here our own analysis of the-state-of-the-art and gather the information that we could find and that we deemed to be relevant. We especially focus our attention to the HZD method since we found it is the most documented and rigorous approach for the control of underactuated robots.

Then, in Chapter 3, we describe how we applied the HZD method, starting from the experimental reports in [127], to design walking trajectories for a real planar biped robot with point feet, called robot VS.

We found that the obtained experimental walking trajectories dissipate more than half of the kinetic energy at impact, causing noise and vibration in the robot. This may be acceptable for a testbed, but not for an industrial product, especially for a lower-limb exoskeleton, where the comfort of the user is crucial. In light of this, in Chapter 4, we study and propose to control the relative kinetic energy dissipation at impact for biped robots. First, we perform a theoretical study using the standard rigid impact model introduced by Hurmuzlu in [56]. Then, we apply our study in simulation to a planar five-link walker with point feet. The control of the energy dissipation allows for the robot to walk with little energy loss at impact, or to stop in "one step" by dissipating most of its energy at impact. We show that our approach is robust to model uncertainties.

In Chapter 5, in an attempt to alleviate the burden due to underactuation, we propose to investigate the additional degree of freedom provided, in the control design, by a change of the time scale in the dynamical equations (Time-Scaling) of planar one degree underactuated biped robots with point feet. It is well known that the conjoint use of Time-Scaling and coordinate and feedback transformations may lead to new normal forms [103], that is to dynamic equations written under a form suitable for the design of control laws. Time-Scaling seems to be especially appropriate for bipedal locomotion since there generally exist geometric quantities, functions of the configuration of the robot, evolving strictly monotonically during a walking step [119]. These quantities, called phase variables in the HZD method, define a natural choice for a new time scale. Using this particular choice of time scale and feedback transformations, we derive new normal forms with two and one-dimensional zero dynamics. Then, using the first order approximation of the dynamics along a reference trajectory in this new time scale, we design a linear quadratic regulator under the form of a state feedback controller, that is a controller depending only on the state and not on the ordinary time t, to stabilize a walking trajectory. The performances of the controller are evaluated on a numerical example of a planar five-link walker with point feet.

Publication

The work described in Section 5.5 of Chapter 5 has been published:

- S. Finet and L. Praly. Feedback Linearization of the Transverse Dynamics for a Class of One Degree Underactuated Systems. *2015 IEEE 54th Annual Conference on Decision and Control (CDC 2015), Dec 2015, Osaka, Japan. pp.7802-7807, Proceedings of the 54th IEEE Conference on Decision and Control (CDC 2015).*

The work realized in Chapter 4 and in the others sections of Chapter 5 will probably be submitted in a near future.

Chapter 1

Modeling of Biped Robots

Résumé français Un bon modèle est un prérequis à la synthèse de lois de contrôle pour un système. Dans ce premier chapitre, nous introduisons le vocabulaire usuel de la locomotion bipède. Puis, nous décrivons un modèle standard utilisé pour la modélisation d'un robot bipède. Un robot bipède est modélisé comme un système hybride, c'est-à-dire régi par une dynamique continue, et une dynamique discrète. Cette dernière est désignée aussi sous le terme de dynamique d'impact, car elle décrit le saut sur l'état lorsque le pied du robot impacte le sol. Elle est obtenue par l'utilisation d'un modèle standard qui suppose un impact instantané et un contact rigide avec le sol. Une description complète et détaillée de cette modélisation se trouve dans [50] et dans les références citées. Nous indiquons aussi comment procéder pour avoir un modèle plus fidèle à la réalité. Ce chapitre ne contient aucune nouveauté par rapport à la littérature. Il présente un assemblage de divers aspects de la modélisation d'un robot bipède et de leur discussion, à partir de notre propre expérience.

A right model is a prerequisite for the design of control of any system. In this chapter, we will first introduce some common vocabulary used in biped locomotion. Then, we will present a standard way of modeling biped robots that we used in this dissertation. Biped robots will be modeled as hybrid systems, that is systems ruled by continuous and discrete dynamics. The continuous dynamics are derived using the method of Lagrange. The discrete dynamics, or impact dynamics, since they describe the change of state when a leg of the robot hits the ground, are derived using a standard model assuming an instantaneous and rigid foot-ground contact. A complete and comprehensive review of this modeling can be found in [50] and references therein. We will finally present others modeling aspects useful to have a model closer to the reality. There is absolutely nothing new in the components of this chapter. Our work consists in the "assembling" of these components and their discussion, this being based on our experience with using them for design and testing.

1.1 Useful Concepts for the Study of Biped Locomotion

Whether it is for the control of humanoid robots or the control of leg exoskeletons, human walking has been a source of inspiration and a model. We introduce here briefly biped walking, and especially human walking, as well as some related vocabulary.

1.1.1 Biped Locomotion

Bipedal walking is a way of locomotion that consists in the displacement of the center of mass of the walker along a given direction thanks to a motion of the legs. During walking, the altitude of the center of mass is always above a certain value, that is the walker does not fall. The motion of the center of mass is almost sinusoidal [129]. Human walking has been studied for more than one century. But, its underlying mechanisms are still misunderstood. Many works agree to say that human walking results of the optimal motion of the center of mass [67, 106]. Yet, it is not clear which function cost human walking minimizes [6]. It seems that the motion of arms plays a role in the reduction of the energetic consumption and of the risk of slippage of the foot [24].

1.1.2 Vocabulary

Human walking on flat ground is a cyclic motion. Three planes are defined to describe the motion of a biped (see figure 1.1):

- The sagittal plane divides the body into right and left halves.
- The frontal plane (or coronal plane) is any vertical plane that divides the body into dorsal and ventral sections.
- The transverse plane divides the body into superior and inferior parts.

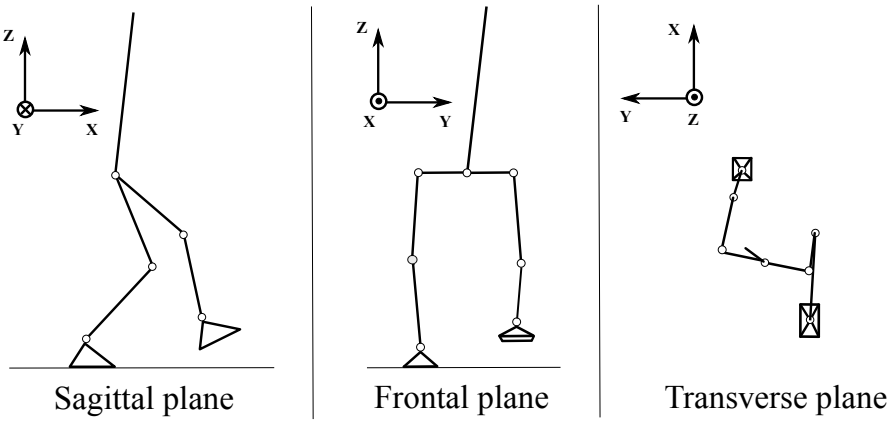


Figure 1.1: The three planes used to describe the motion of a biped walker.

The motion of walking essentially occurs in the sagittal plane. That's why the study of planar robot, as it is done in this PhD thesis, is carried out in the sagittal plane.

Human walking can be divided into two phases (see figure 1.2):

- The single support phase: only one foot is in contact with the ground.
- The double support phase: both feet are in contact with the ground.

A leg is said to be in *stance phase* when it is in contact with the ground. It is said to be in *swing phase* when off the ground.

During the stance phase, the swing foot first impacts the ground with the heel. The foot rotates about the heel. Next, the foot lays flat on the ground. Then, the heel lifts from the ground and the foot rotates about the toe. The stance phase finishes when the toe lifts from the ground. This behavior seems to play an important role in human walking.

When the walking speed increases, the duration of the double support phase diminishes until disappearing. In this latter case, the double support phase is replaced by the *flight phase*, during which both legs are off the ground. The biped is running.

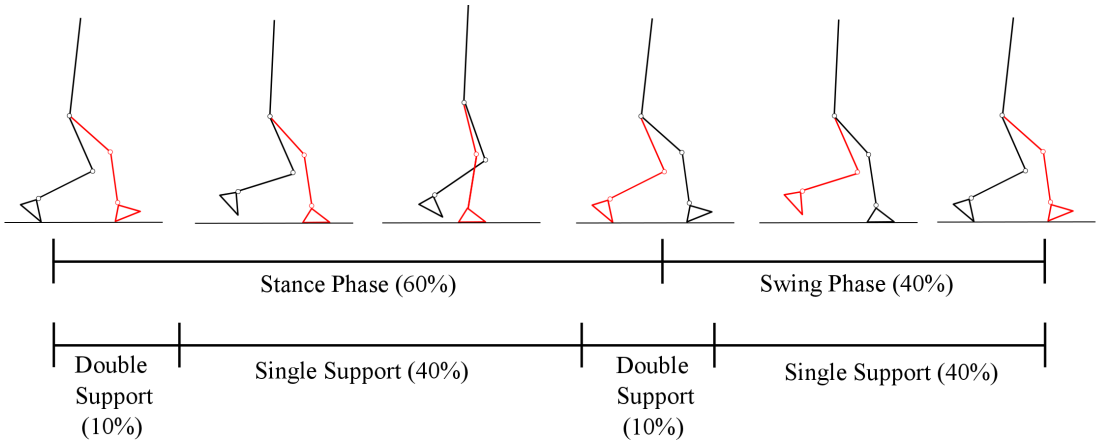


Figure 1.2: The different phases in human walking

1.2 Dynamic Model in Flight Phase

From now, we will present the mathematical models and the notations used in this dissertation. This is a standard modeling described in [50, 127]. In this section, we derive the dynamics of the robot in flight phase.

1.2.1 Model Assumptions and Notations

We consider here a general model of biped robots. The robot can be planar or tridimensional and have feet or not. In this latter case, the robot is said to have

point feet. The robot is composed of rigid body links forming a N_b degrees of freedom tree structure. We assume that $N_a \leq N_b$ joints are independently actuated. One example of such a model is depicted in Figure 1.3. Some others examples are the planar five-link walker described in Chapter 4 or the models used in [23, 1, 127]. To distinguish between the 2 legs, we label them by the indices "1" and "2".

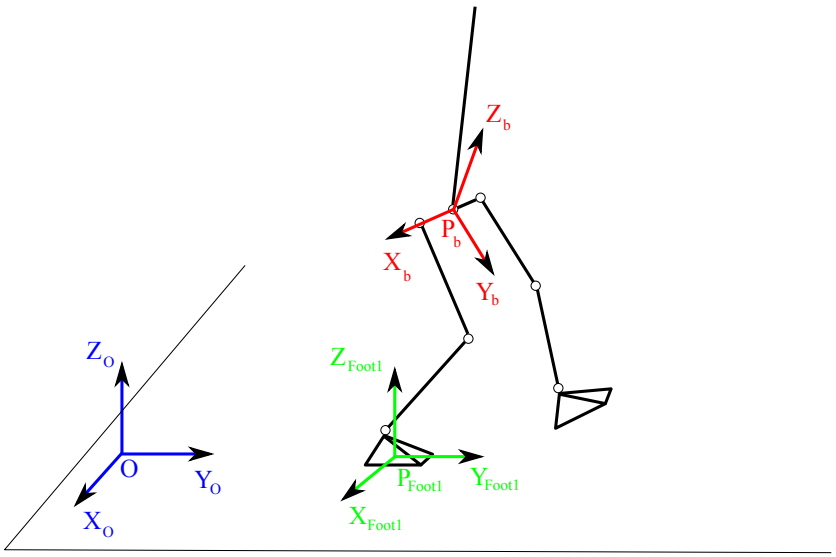


Figure 1.3: Model of a biped robot and the frames used to describe its configuration.

Let $\mathcal{R}_o = (O, X_o, Y_o, Z_o)$ be a frame attached to the world (world frame). Let $p_b \in \mathbb{R}^3$ ($\mathbb{R}^2$ for a planar model), be[1] the cartesian coordinates of a point P_b attached to the robot and expressed in $\mathcal{R}_o$. Let $\mathcal{R}_b$ be a frame attached to the robot and centered at P_b: $\mathcal{R}_b = (P_b, X_b, Y_b, Z_b)$. Let $\phi_b \in \mathbb{T}^3$ (= the 3-torus) be the orientation of $\mathcal{R}_b$ relative to $\mathcal{R}_o$. ϕ_b corresponds to the Euler angles (roll, pitch, yaw). For a planar model, $\phi_b \in \mathbb{T}$ and represents the pitch angle.

The vector (p_b, ϕ_b) describes the absolute position and orientation of the robot.

Let q_b be the vector of the N_b joint angles of the robot in $\mathcal{R}_b$, called body coordinates. Without loss of generality, we assume that the N_a first components of q_b are the actuated joints.

Let $N = N_b + 6$ ($N = N_b + 3$ for a planar model). The N-dimensional vector q_e forms a set of generalized coordinates[2], i.e. a set of variables that completely describes the configuration of the robot, in $\mathcal{R}_o$

$$q_e = \begin{pmatrix} q_b \\ p_b \\ \phi_b \end{pmatrix}. \tag{1.1}$$

$q_e \in \mathcal{Q}_e$, where $\mathcal{Q}_e \subset \mathbb{R}^N$ is the *configuration space* of the robot and $\mathcal{TQ}_e$ is the corresponding tangent bundle, also called *state space*. Denoting $\dot{q}_e$ as the generalized velocities, the state of the robot is

[1] Subscript "b" refers to "body".

[2] "e" stands for "extended", since we will define further from q_e a set of *reduced coordinates*.

$$x_e = \begin{pmatrix} q_e \\ \dot{q}_e \end{pmatrix} \in \mathcal{TQ}_e \,. \tag{1.2}$$

1.2.2 Flight Phase Dynamics

Let us assume that the robot is not in contact with the environment (*flight phase*) and is only subject to the gravity. We derive here the dynamical equations using the Lagragian mechanics.

1.2.2.1 Lagrangian Mechanics

Lagrangian mechanics offers a convenient way to derive the dynamical equations of mechanical systems, and is widely used in Mechanics and Robotics. We introduce briefly this method. For further details, see [127, 80, 38].

The scalar function $\mathcal{L} : \mathcal{TQ}_e \to \mathbb{R}$, defined as the difference between the kinetic and the potential energy of the system, is called *Lagrangian*. We have

$$\mathcal{L} = \mathrm{K}(q_e, \dot{q}_e) - \mathrm{V}(q_e) \,, \tag{1.3}$$

where $\mathrm{K}(q_e, \dot{q}_e)$ is the kinetic energy and $\mathrm{V}(q_e)$ is the potential energy.

The *Lagrange's equations* give the dynamical equations of the system

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}_e} - \frac{\partial \mathcal{L}}{\partial q_e} = \Gamma \,, \tag{1.4}$$

where Γ is the vector of the *generalized forces and torques*.

This expression can be put in a more useful form when the kinetic energy is quadratic in velocity, that is

$$\mathrm{K}(q_e, \dot{q}_e) = \frac{1}{2}\dot{q}_e^T D(q)\dot{q}_e \,, \tag{1.5}$$

where $D(q_e)$ is a positive definite matrix, called *mass matrix*. This is the case for most of mechanical systems, including biped robots. Then, the Lagrange's equations become

$$\frac{d}{dt}\Big(D(q_e)\dot{q}_e\Big) - \frac{1}{2}\Big(\frac{\partial}{\partial q_e}\Big(\dot{q}_e^T D(q_e)\dot{q}_e\Big)\Big)^T - \frac{\partial \mathrm{V}}{\partial q_e} = \Gamma \,, \tag{1.6}$$

which can be written under the standard form

$$D(q_e)\ddot{q}_e + C(q_e, \dot{q}_e)\dot{q}_e + G(q_e) = \Gamma \,, \tag{1.7}$$

where $C(q_e, \dot{q}_e)$ is the *Coriolis matrix* and $G(q_e)$ is the vector of the potential energy. We have

$$G(q_e) = -\frac{\partial V}{\partial q_e}(q_e)\,, \tag{1.8}$$

and

$$C(q_e, \dot{q}_e) = \Big(\frac{\partial}{\partial q_e}\Big(D(q_e)\dot{q}_e\Big)\Big) - \frac{1}{2}\Big(\frac{\partial}{\partial q_e}\Big(D(q_e)\dot{q}_e\Big)\Big)^T\,. \tag{1.9}$$

The Coriolis matrix is not defined uniquely. But, it is generally defined as

$$C(q_e, \dot{q}_e) = \sum_{i=1}^{N} \frac{1}{2}\Big(\frac{\partial D_e^{k,j}}{\partial q_i} + \frac{\partial D_e^{k,i}}{\partial q_j} - \frac{\partial D_e^{i,j}}{\partial q_k}\Big)\dot{q}_i\,, \tag{1.10}$$

where $D_e^{k,j}$ is the kj entry of the matrix D_e.

The vector of generalized forces and torques Γ defines the contribution of each external forces and torques on the dynamics of the coordinates q_e. It is calculated using the principle of virtual works, also called the d'Alembert's principle (see page 52 in [38]).

More precisely, if a force F is applied on a point of coordinates $p(q)$, the corresponding generalized force is

$$\Gamma_F = \Big(\frac{\partial p}{\partial q}(q)\Big)^T F\,. \tag{1.11}$$

If a torque u is exerted along an axis parameterized by the angle $\theta(q)$, the generalized torque is

$$\Gamma_u = \Big(\frac{\partial \theta}{\partial q}(q)\Big)^T u\,. \tag{1.12}$$

1.2.2.2 Flight phase Dynamics

Using the Lagrange's equations, the equations of dynamics for the considered models of biped robots are

$$D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e)\dot{q}_e + G(q_e) = B_e u\,, \tag{1.13}$$

where $u \in \mathbb{R}^{N_a}$ is the vector of inputs (motor torques), $G(q_e)$ is the gravity vector, and B_e is a $N \times N_a$ matrix defining how the motor torques act on the coordinates and obtained using (1.12). Since q_b corresponds to the joints of the robot, and its N_a first components correspond to the actuated joints, we have

$$B_e = \begin{pmatrix} I_{N_a \times N_a} \\ 0_{(N-N_a)\times N_a} \end{pmatrix}\,. \tag{1.14}$$

The equations in *state-variable* form are

$$\dot{x}_e = \begin{bmatrix} \dot{q}_e \\ D_e^{-1}(q_e)[-C_e(q_e, \dot{q}_e)\dot{q}_e - G(q_e) + B_e u] \end{bmatrix}. \tag{1.15}$$

They can be written as

$$\dot{x}_e = f_e(x_e) + g_e(x_e)u. \tag{1.16}$$

1.3 Dynamics With Ground Contact

During walking, the robot has one or both feet on the ground. We present here how to model the foot-ground contact and how to include it in the dynamical equations. The foot-ground contact is commonly modeled using holonomic constraints, that is relations under the form $\eta(q) = 0$. This provides an easy way to compute the corresponding ground reaction efforts. This is a standard modeling described in [**?**, 127].

1.3.1 Modeling the Foot-Ground Contact

Without loss of generality, let us consider the foot of leg 1. The foot-ground contact is modeled as a kinematic joint parameterized[3] by six degrees of freedom: three of them describe the foot position, and the three others describe the foot orientation relative to the ground.

Let $p_{Foot1} \in \mathbb{R}^3$ be[4] the cartesian coordinates of a point of leg 1, called P_{Foot1} and expressed in $\mathcal{R}_o$

$$p_{Foot1}(q_e) = \left(p^x_{Foot1}(q_e);\, p^y_{Foot1}(q_e);\, p^z_{Foot1}(q_e) \right), \tag{1.17}$$

where $p^i_{Foot1}(q_e)$ is the component of p_{Foot1} along axis i.

Let $\mathcal{R}_{Foot1}$ be a frame attached to the foot and centered at p_{Foot1}. $\phi_{Foot1} \in 0, 2\pi^3$ is [5] the orientation of $\mathcal{R}_{Foot1}$ relative to $\mathcal{R}_o$

$$\phi_{Foot1}(q_e) = \left(\phi^x_{Foot1}(q_e);\, \phi^y_{Foot1}(q_e);\, \phi^z_{Foot1}(q_e) \right), \tag{1.18}$$

where $\phi^i_{Foot1}(q_e)$ the orientation of the frame along axis i.

The vector $(p_{Foot1}(q_e); \phi_{Foot1}(q_e))$ regroups all the possible degrees of freedom of the joint between the foot and the ground. Depending on the nature of the contact, some of them are locked. For example, ignoring friction, Figure 1.4 represents the

[3] For a planar model, it is parameterized by three degrees of freedom: two of them describe the foot position, and one the foot orientation.

[4] $p_{Foot1} \in \mathbb{R}^2$ for a planar model

[5] $\phi_{Foot1} \in 0, 2\pi^2$ for a planar model.

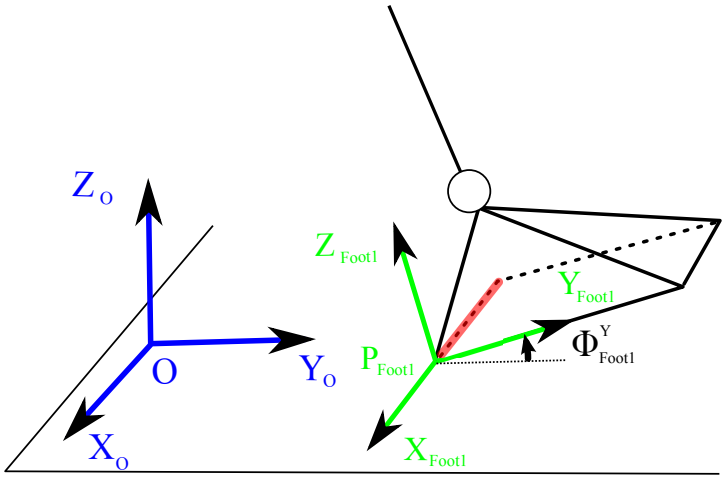


Figure 1.4: Foot-ground contact modeled as a pivot. The contact surface is the red line and the foot is assumed to rotate along this line. The rotation is parameterized by the angle $\phi^y_{Foot_1}$.

heel-ground contact modeled as a pivot (only one rotational degree of freedom is free).

Without loss of generality, let us assume that the nature of the contact locks $k_{p_{Foot1}} \in \{1, 3\}$ translational degrees of freedom, and $k_{\phi_{Foot1}} \in \{1, 3\}$ rotational degrees of freedom. By changing, if necessary, the order of its components, p_{Foot1} can be written as [6]

$$p_{Foot1}(q_e) = \left(p^f_{Foot1}(q_e);\, p^l_{Foot1}(q_e)\right), \tag{1.19}$$

where $p^f_{Foot1}(q_e)$ is a $(3 - k_{p_{Foot1}})$-dimensional column vector (resp. $p^l_{Foot1}(q_e)$ is a $k_{p_{Foot1}}$-dimensional row vector) corresponding to the free translational degrees of freedom (resp. translational degrees of freedom locked by the contact with the ground).

Similarly, we have

$$\phi_{Foot1}(q_e) = \left(\phi^f_{Foot1}(q_e);\, \phi^l_{Foot1}(q_e)\right), \tag{1.20}$$

where $\phi^f_{Foot1}(q_e)$ is a $(3 - k_{\phi_{Foot1}})$-dimensional column vector (resp. $\phi^l_{Foot1}(q_e)$ is a $k_{\phi_{st}}$-dimensional row vector) corresponding to the free rotational degrees of freedoms (resp. rotational degrees of freedom locked by the contact with the ground).

The degrees of freedom of foot 1 are subject to kinematic constraints. These constraints are gathered in the vector of dimension $k_{p_{Foot1}} + k_{\phi_{Foot1}}$

[6] Indice "f" refers to "free" and "l" to "locked".

$$\eta_{Foot1}(q_e) = \left(p^l_{Foot1}(q_e) - p^{l*}_{Foot1};\ \phi^l_{Foot1}(q_e) - \phi^{l*}_{Foot1} \right) = 0\,, \tag{1.21}$$

where p^{l*}_{Foot1} and ϕ^{l*}_{Foot1} are real constants corresponding to the value taken by the locked degrees of freedom.

$\eta_{Foot1}(q_e) = 0$ is a mechanical constraint that only depends on the positions q_e. It is called a *holonomic constraint* [80].

Remark: In the numerical examples of this thesis, we consider only planar biped robots with point feet. The contact of a point foot with the ground is modeled as an ideal (= no friction) revolute joint and is unactuated.

1.3.2 Ground Reaction Wrench on One Foot

The robot is subject to constraint forces ensuring that the holonomic constraints $\eta_{Foot1}(q_e) = 0$ are respected. The interaction between the foot and the ground can be modeled as a wrench [7] acting on the foot . Let $\mathcal{T}_{Foot1}$ be the wrench exerted by the ground on foot 1 expressed at P_{Foot1}. $\mathcal{T}_{Foot1}$ is a six-dimensional vector gathering the ground reaction force $\mathcal{F}_{Foot1}$ and the moment exerted by $\mathcal{F}_{Foot1}$ at P_{Foot1}, denoted $\mathcal{M}_{Foot1}$

$$\mathcal{T}_{Foot1} = (\mathcal{F}^x_{Foot1};\ \mathcal{F}^y_{Foot1};\ \mathcal{F}^z_{Foot1};\ \mathcal{M}^x_{Foot1};\ \mathcal{M}^y_{Foot1};\ \mathcal{M}^z_{Foot1})\,, \tag{1.22}$$

where $\mathcal{F}^i_{Foot1}$ is the component of the ground reaction force $\mathcal{F}_{Foot1}$ along axis i, and $\mathcal{M}^i_{Foot1}$ is the moment of $\mathcal{F}_{Foot1}$ along axis i at P_{Foot1}.

The joint between the foot and the ground is assumed to be ideal (no friction). Then, only the components of $\mathcal{T}_{Foot1}$ related to the locked degrees of freedom p^l_{Foot1} and ϕ^l_{Foot1} are generally non zero. Thus, we introduce the $(k_{p_{Foot1}} + k_{\phi_{Foot1}})$-dimensional vector

$$\lambda_{Foot1} = (\mathcal{F}^l_{Foot1};\ \mathcal{M}^l_{Foot1})\,, \tag{1.23}$$

where $\mathcal{F}^l_{Foot1}$ is the column vector of dimension $k_{p_{Foot1}}$ gathering the components of the force $\mathcal{F}_{Foot1}$ along the locked degrees of freedom p^l_{Foot1}. $\mathcal{M}^l_{Foot1}$ is the column vector of dimension $k_{\phi_{Foot1}}$ gathering the components of the moment $\mathcal{M}_{Foot1}$ about the axes for which rotations are locked, i.e. rotations parameterized by the locked degrees of freedom ϕ^l_{Foot1}.

In the case where the foot is assumed to be fixed and flat on the ground, all the degrees of freedom between the foot and the ground are locked, i.e. $\eta_{Foot1}(q_e) = (p_{Foot1}(q_e) - p^*_{Foot1};\ \phi_{Foot1}(q_e) - \phi^*_{Foot1})$, and $\lambda_{Foot1} = (\mathcal{F}_{Foot1};\ \mathcal{M}_{Foot1})$.

[7] A wrench is a mathematical object gathering a force and its corresponding moment. See page 61 in [80] for further details.

1.3.3 Dynamic Model With Ground Contact

The previous approach can be applied to the foot of leg 2, by replacing the subscript "Foot1" by "Foot2" in the equations. The constraints exerted on the two feet are

$$\eta(q_e) = \begin{pmatrix} \eta_{Foot1}(q_e) \\ \eta_{Foot2}(q_e) \end{pmatrix} = 0\,. \tag{1.24}$$

The number of degrees of freedom locked for the two feet, or equivalently the size of $\eta(q_e)$, is

$$p = k_{p_{Foot1}} + k_{\phi_{Foot1}} + k_{p_{Foot2}} + k_{\phi_{Foot2}}\,. \tag{1.25}$$

We assume that the jacobian of $\eta(q_e)$, a rectangular matrix of size $p \times N$, is full rank, that is

$$\frac{\partial \eta}{\partial q_e}(q_e) \text{ is full rank (equal to } p)\,. \tag{1.26}$$

As we will see later, with this assumption, the holonomic contraints $\eta(q_e) = 0$ allow for the definition of a reduced set of generalized coordinates.

The non zero components of the ground reaction wrenches exerted on both feet are put in a vector of size p

$$\lambda = \begin{pmatrix} \lambda_{Foot1} \\ \lambda_{Foot2} \end{pmatrix}. \tag{1.27}$$

λ is also called the vector of *Lagrange multipliers* [80].

Let us now calculate the equations of motion of the robot in contact with the ground. The robot is assumed to be subject to the gravity and to the ground reaction forces. The efforts related to the holonomic contraints are computed using the principle of virtual works (see (1.11)). The dynamical equations are obtained using the dynamical equations in flight phase (1.13) to which the ground reaction efforts are added

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e)\dot{q}_e + G(q_e) = B_e(q_e)u + \left(\dfrac{\partial \eta}{\partial q_e}(q_e)\right)^T \lambda \\ \eta_(q_e) = 0 \end{cases} \tag{1.28}$$

where $u \in \mathbb{R}^{N_a}$ is the vector of inputs (motor torques).

1.3.3.1 Calculating λ

The non zero component of the contact wrenches are obtained by enforcing the holonomic constraints to zero and using the equations of motion [80]. Differentiating (1.24) twice with respect to time gives

$$\frac{\partial \eta}{\partial q_e}(q_e)\ddot{q}_e + \frac{\partial}{\partial q_e}\Big(\frac{\partial \eta}{\partial q_e}(q_e)\dot{q}_e\Big)\dot{q}_e = 0_{p\times 1}\,. \tag{1.29}$$

Then, we have the following equations to solve for λ

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e,\dot{q}_e)\dot{q}_e + G(q_e) = B_e(q_e)u + \Big(\dfrac{\partial \eta}{\partial q_e}(q_e)\Big)^T\lambda \\ \dfrac{\partial \eta}{\partial q_e}(q_e)\ddot{q}_e = -\dfrac{\partial}{\partial q_e}\Big(\dfrac{\partial \eta}{\partial q_e}(q_e)\dot{q}_e\Big)\dot{q}_e \end{cases} \tag{1.30}$$

Given that $\frac{\partial \eta}{\partial q_e}(q_e)$ is assumed to be full rank and $D_e^{-1}(q_e)$ is positive definite, $\frac{\partial \eta}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^T}{\partial q_e}(q_e)$ is invertible.

Then

$$\begin{aligned}\lambda = \Big(\frac{\partial \eta}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^T}{\partial q_e}(q_e)\Big)^{-1}\Bigg(&-\frac{\partial}{\partial q_e}\Big(\frac{\partial \eta}{\partial q_e}(q_e)\dot{q}_e\Big)\dot{q}_e \\ &+ \frac{\partial \eta}{\partial q_e}(q_e)D_e^{-1}(q_e)\Big[C_e(q_e,\dot{q}_e)\dot{q}_e + G_e(q_e) - B_e u\Big]\Bigg).\end{aligned} \tag{1.31}$$

Hence, the contact wrenches depend on q_e, $\dot{q}_e$ and u. They are linear with the input u.

1.3.4 Dynamics in a Reduced Set of Coordinates

The dynamical equations (1.28) are expressed using the coordinates q_e of the configuration manifold $\mathcal{Q}_e$. But, due to the p holonomic constraints (1.24), the system is constrained to move in the submanifold

$$\mathcal{Q}_e^\eta = \{q_e \in \mathcal{Q}_e \,|\, \eta(q_e) = 0\} \subset \mathbb{R}^N\,. \tag{1.32}$$

Then, the generalized coordinates q_e are not independent from each other and we show here that it is possible to find a reduced set of $(N-p)$ generalized positions, denoted q, for the system. We then express the dynamical equations in the new set of coordinates $(q,\dot{q})$, that we call *reduced coordinates*.

1.3.4.1 A Reduced Set of Coordinates

From a classical result of differential geometry, given that $\frac{\partial \eta}{\partial q_e}(q_e)$ is full rank equal to p, $\mathcal{Q}_e^\eta$ is a $N-p$ dimensional embedded manifold of $\mathcal{Q}_e$. The implicit functions theorem allows to find a set of $N-p$ independent coordinates $q \in \mathcal{Q}_f^\eta \subset \mathrm{R}^{N-p}$, and p constrained coordinates $q_c \in \mathcal{Q}_c^\eta \subset \mathrm{R}^p$ such that the diffeomorphism $\Phi^\eta : \mathcal{Q}_f^\eta \times \mathcal{Q}_c^\eta \to \mathcal{Q}_e$ exists. Then, the mapping from a point of $\mathcal{Q}_f^\eta$ to a point of $\mathcal{Q}_e$

$$\begin{cases} E^\eta : \mathcal{Q}_f^\eta \to \mathcal{Q}_e^\eta \subset \mathcal{Q}_e \\ \qquad q \quad \to \; q_e = E^\eta(q) \end{cases} \tag{1.33}$$

is an *embedding*, that is $E^\eta(Q_f^\eta)$ is diffeomorphic to Q_e^η. Differently said, the coordinates $(q_e; \dot{q}_e)$ can be reconstructed from the reduced coordinates $(q; \dot{q})$, and it is possible to use either one or the other set of coordinates on the condition that $\eta(q_e) = 0$.

Since $\mathcal{Q}_f^\eta$ is a manifold of dimension $N - p$, its corresponding tangent bundle $\mathcal{TQ}_f^\eta$ is a manifold of dimension $2(N - p)$. The state of the robot evolves in the image of $\mathcal{TQ}_f^\eta$ under E^η. It corresponds to the tangent bundle of Q_e^η, that is $\mathcal{TQ}_e^\eta = E^\eta(\mathcal{TQ}_f^\eta)$. Then, we have

$$\begin{aligned}\mathcal{TQ}_e^\eta &= \left\{ \begin{pmatrix} q_e \\ \dot{q}_e \end{pmatrix} \in \mathcal{TQ}_e \,|\, \eta(q_e) = 0,\, \frac{\partial \eta}{\partial q_e}(q_e)\dot{q}_e = 0 \right\} \\ &= \left\{ \begin{pmatrix} q_e \\ \dot{q}_e \end{pmatrix} \in \mathcal{TQ}_e \,|\, q_e = E^\eta(q),\, \dot{q}_e = \frac{\partial E^\eta}{\partial q}(q)\dot{q},\, (q, \dot{q}) \in \mathcal{TQ}_f^\eta \right\}.\end{aligned} \tag{1.34}$$

Hence, the state of the system can be described using the reduced coordinates $(q, \dot{q})$, and the study can be performed in the lower dimensional manifold $\mathcal{TQ}_f^\eta$.

1.3.4.2 Dynamical Equations in the Reduced Coordinates

The dynamical equations using the reduced coordinates $(q, \dot{q})$ can be derived using the method of Lagrange, that is computing the kinetic and potential energies in these coordinates. More simply, they can also be determined from the dynamic model in the extended coordinates $(q_e, \dot{q}_e)$ (1.28). Using the embedding from the extended coordinates to the reduced coordinates, we have

$$\begin{cases} q_e = E^\eta(q) \\ \dot{q}_e = \dfrac{\partial E^\eta}{\partial q}(q)\dot{q} \\ \ddot{q}_e = \dfrac{\partial}{\partial q}\Big(\dfrac{\partial E^\eta}{\partial q}(q)\dot{q}\Big)\dot{q} + \dfrac{\partial E^\eta}{\partial q}(q)\ddot{q} \end{cases} \tag{1.35}$$

Replacing the expression of q_e and of its time derivatives in the dynamical equations gives

$$\begin{aligned}&\Big(D_e(q_e)\frac{\partial E^\eta}{\partial q}(q)\Big)\ddot{q} + \Big(D_e(q_e)\frac{\partial}{\partial q}\Big[\frac{\partial E^\eta}{\partial q}(q)\dot{q}\Big] + C_e(q_e, \dot{q}_e)\frac{\partial E^\eta}{\partial q}(q)\Big)\dot{q} + G_e(q_e) = \\ &\quad B_e u + \left(\frac{\partial \eta}{\partial q_e}(q_e)\right)^T \lambda\,,\end{aligned} \tag{1.36}$$

To eliminate the Lagrange multipliers λ, we use

$$\left(\frac{\partial E^\eta}{\partial q}(q)\right)^T \left(\frac{\partial \eta}{\partial q_e}(q_e)\right)^T = \left(\frac{\partial \eta}{\partial q_e}(q_e)\frac{\partial E^\eta}{\partial q}(q)\right)^T = 0\,, \tag{1.37}$$

since $\frac{\partial \eta}{\partial q_e}(q_e)\frac{\partial E^\eta}{\partial q}(q) = \frac{d}{dq_e}\Big(\eta\Big(E^\eta(q)\Big)\Big) = 0$.

Then, multiplying the equations by $\Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T$ give

$$\Big(\Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T D_e(q)\Big(\frac{\partial E^\eta}{\partial q}(q)\Big)\Big)\ddot{q} + \Big(\Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T\Big(D_e(q)\frac{\partial}{\partial q}\Big[\frac{\partial E^\eta}{\partial q}(q)\dot{q}\Big] + C_e(q,\dot{q})\Big)\Big)\dot{q} + \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T G_e(q_e) = \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T B_e u\,. \tag{1.38}$$

Hence, the dynamical equations in the coordinates $(q,\dot{q})$ can be written under the classical form

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = B(q)u\,, \tag{1.39}$$

with :

$$\begin{cases} D(q) = \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T D_e\Big(E^\eta(q)\Big)\Big(\frac{\partial E^\eta}{\partial q}(q)\Big) \\ C(q,\dot{q}) = \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T\left(C_e(q_e,\dot{q}_e)\frac{\partial E^\eta}{\partial q}(q) + D_e(E(q))\frac{\partial}{\partial q}\Big(\frac{\partial E^\eta}{\partial q}\Big)\right) \\ G(q) = \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T G_e(q_e) \\ B = \Big(\frac{\partial E^\eta}{\partial q}(q)\Big)^T B_e \end{cases} \tag{1.40}$$

It can be also written under the classical form

$$\dot{x} = f(x) + g(x)u\,, \tag{1.41}$$

whith $x = (q;\dot{q})$, $f(x) = D(q)^{-1}\Big(-C(q,\dot{q})\dot{q} - G(q)\Big)$, and $g(x) = D(q)^{-1}B(q)u$.

It is usual to use the reduced coordinates $(q;\dot{q})$, since they define a reduced-order model for the system. This reduced-order model is equivalent to (1.28).

1.3.5 Domain of Admissibility of the Model

The previously derived dynamic models (1.28) or (1.39) are valid as long as the holonomic constraints are respected, or differently said, that the nature of the foot-ground contact is preserved.

For that, the ground reaction wrench(es) must respect some constraints. Let us consider that the foot-ground contact is described by the set of holonomic constraints $\eta(q_e) = 0$. Let us assume that the foot 1 is in contact with the ground (the case is the same for the foot 2). In order to avoid that the foot lifts off the ground, the vertical component of the ground reaction force must stay positive

$$\mathcal{F}^z_{Foot1} > 0\,. \tag{1.42}$$

Furthermore, to avoid slipping, using a simple standard model (the Coulomb model), the ground reaction force must stay in the cone of friction, i.e.

$$\sqrt{(\mathcal{F}^x_{Foot1})^2 + (\mathcal{F}^y_{Foot1})^2} < \mu \mathcal{F}^z_{Foot1}\,, \tag{1.43}$$

where μ is the Coulomb static friction coefficient. It was taken equal to 0.6 for the robots RABBIT [127] or MARLO [46]. This corresponds to the friction coefficient between rubber and dry concrete.

If the ground contact wrench respect these constraints, then the foot does not translate relatively to the ground.

But, if the size of the feet is non zero, additional constraints on the ground reaction wrenches must be verified. Especially, if the foot is flat on the ground, the center of pressure of the ground reaction force must stay inside the convex hull of the contacting points. Otherwise, the foot can rotate about one of its edges. This constraint is known as the Zero Moment Point (ZMP) constraint [50].

All these constraints can be gathered into a vector of inequalities that must be respected by the non zero components of the contact wrenches λ

$$P^\eta(\lambda) > 0\,. \tag{1.44}$$

Furthermore, for a robot in flight phase or in single support phase, if the swing foot impacts the ground, the holonomic constraints are modified and the model is not valid anymore. Then, the swing foot height must stay strictly greater than the ground height h

$$alt(q_e) > h\,, \tag{1.45}$$

where $q_e \to alt(q_e)$ is the swing foot altitude.

As a result, for a given set of holonomic constraints $\eta(q_e) = 0$, the domain of admissibility $\mathcal{D}_\eta$ can be described as bounds on the state and on the contact wrenches λ, or on the inputs u (see (1.31)), that result on inequality constraints H_η, called *unilateral constraints*

$$\mathcal{D}_\eta = \left\{ \begin{pmatrix} q_e \\ \dot{q}_e \\ u \end{pmatrix} \in \mathcal{TQ}^\eta_e \times \mathbb{R}^{N_a} |\, H_\eta(q_e, \dot{q}_e, u) > 0 \right\}. \tag{1.46}$$

If the unilateral constraints are violated, the dynamical equations (1.28) with the constraints $\eta(q_e) = 0$ are not valid anymore. An other model must be considered. Especially, if the new foot-ground contact can be described by an other set of holonomic constraints $\tilde{\eta}(q_e) = 0$, the new model is (1.28) (or (1.39)) with $\tilde{\eta}(q_e) = 0$.

1.4 Impact dynamics

Until now, we derived a continuous dynamical model of the robot using Lagrangian dynamics, and assuming that the foot-ground contact remains unchanged. We

modeled the foot-ground contact using holonomic constraints. This implicitly assumes that no deformation of the foot and of the ground occur during the contact. This hypothesis is often referred as the *rigid ground model*.

We focus here on the dynamics when the foot-ground contact is modified, for instance when the swing leg of the robot hits the ground. When the foot-ground contact is modified, the holonomic constraints are modified. This modification is of very short duration, and can be considered as instantaneous. This event is called an *impact*. The *impact dynamics* is the discrete dynamics describing the evolution of the state when the impact occurs. We assume that the position is continuous and that the velocity may undergo a discontinuity (or jump) when subject to this dynamics. The impact dynamics allows to switch between two continuous phases subject to different sets of holonomic constraints.

Remark: Note that we adopt here a general definition of what an impact is. Especially, we consider that an impact does not necessarily involve a collision with the ground, even though this is our main concern.

1.4.1 Derivation of the impact map

We derive here the impact dynamics using the standard approach introduced by Hurmuzlu [55, 56]. This model is commonly used in bipedal locomotion. It is especially used in the HZD method to design walking trajectories [127]. The model considers that the impact of one leg onto the ground is instantaneous and inelastic (coefficient of restitution equal to 0). Others impact models exist, but they are more complicated, and less appropriate for an analytical study. See [13, 75, 56] and references therein for further details.

We will consider that the impact makes the transition between the two following phases. Any quantities related to the model before impact (resp. after impact) will be superscripted by "-" (resp. "+"). The derivation must be performed in a set of coordinates that is valid before and after impact. Hence, we use the extended coordinates $(q_e; \dot{q}_e)$.

Let us consider that the robot is subject to p^- holonomic constraints $\eta^-(q_e) = 0$ before impact. Then, the configuration space is a $(N - p^-)$-dimensional embedded manifold of $\mathcal{Q}_e$

$$\mathcal{Q}_e^{\eta^-} = \{q_e \in \mathcal{Q}_e \,|\, \eta^-(q_e) = 0\}\,. \tag{1.47}$$

The state evolves in the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ and the dynamical equations are

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e) + G_e(q_e) = B_e u + \left(\dfrac{\partial \eta^-}{\partial q_e}(q_e)\right)^T \lambda \\ \eta^-(q_e) = 0 \end{cases} \tag{1.48}$$

Let us consider that the phase after impact is defined by p^+ holonomic constraints $\eta^+(q_e) = 0$. The configuration space is a $(N - p^+)$-dimensional embedded manifold of $\mathcal{Q}_e$

$$\mathcal{Q}_e^{\eta^+} = \{q_e \in \mathcal{Q}_e \,|\, \eta^+(q_e) = 0\}\,. \tag{1.49}$$

The state evolves in the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^+}$ and the dynamical equations are

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e) + G_e(q_e) = B_e u + \left(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\right)^T \lambda \\ \eta^+(q_e) = 0 \end{cases} \tag{1.50}$$

We will consider that a switch between (4.2) and (4.4), that is an impact, occurs when the state evolving in $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ crosses the *impact surface*

$$\mathcal{S}_{\eta^- \to \eta^+} = \left\{ x_e := (q_e; \dot{q}_e) \in \mathcal{T}\mathcal{Q}_e^{\eta^-} \,|\, \eta^+(q_e) = 0 \right\}, \tag{1.51}$$

and therefore $\eta^-(q_e) = 0$ and $\eta^+(q_e) = 0$.

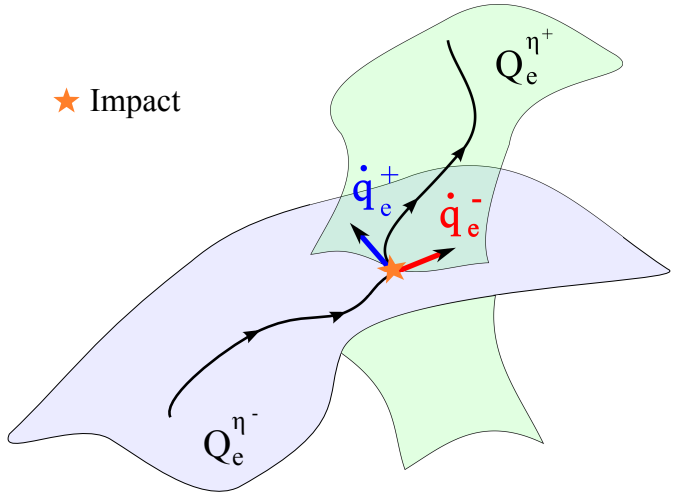


Figure 1.5: Geometrical interpretation of the impact dynamics. A jump on the velocity (from $\dot{q}_e^-$ to $\dot{q}_e^+$) may be necessary so that the state after impact be in $\mathcal{T}\mathcal{Q}_e^{\eta^+}$. The impact map can be viewed as a jump from the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ to the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^+}$.

1.4.1.1 The Impact Map: a Jump From $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ to $\mathcal{T}\mathcal{Q}_e^{\eta^+}$

Let $x_e^- = (q_e^-; \dot{q}_e^-)$ be the state just before the impact, and $x_e^+ = (q_e^+; \dot{q}_e^+)$ be the state just after the impact. The *impact map* is the function mapping a point from $\mathcal{S}_{\eta^- \to \eta^+}$ to a point of $\mathcal{T}\mathcal{Q}_e^{\eta^+}$

$$\begin{cases} \Delta : \mathcal{S}_{\eta^- \to \eta^+} & \to & \mathcal{T}\mathcal{Q}_e^{\eta^+} \\ \quad x_e^- & \to & x_e^+ = \Delta(x_e^-) \end{cases} \tag{1.52}$$

By assumption the position q_e is continuous at impact: $q_e^+ = q_e^- = q_e$. On the contrary, the velocity may undergo a jump. Indeed, prior to the impact, the

position of the robot evolves in the manifold $\mathcal{Q}_e^{\eta^-}$ and its velocity is tangent to this manifold. At impact, the position intersects $\mathcal{Q}_e^{\eta^+}$. But, the velocity is not necessarily tangent to $\mathcal{Q}_e^{\eta^+}$

$$\frac{\partial \eta^+}{\partial q_e}(q_e^-)\dot{q_e}^- \neq 0\,. \tag{1.53}$$

Hence, so that the position remains in $\mathcal{Q}_e^{\eta^+}$, the velocity must undergo a jump to become tangent to $\mathcal{Q}_e^{\eta^+}$. The velocity after impact must verify

$$\frac{\partial \eta^+}{\partial q_e}(q_e^+)\dot{q_e}^+ = 0\,. \tag{1.54}$$

Hence, geometrically, as illustrated in figure 1.5, the impact map is a jump from the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ to the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^+}$, with $q_e^- = q_e^+$.

1.4.1.2 Derivation of the Impact Map Δ

Let us now derive the impact map. We assume that the constraints $\eta^+(q_e) = 0$ are verified "during" the impact. Then, using (1.6) the dynamical equations "during" the impact are

$$\frac{d}{dt}\Big(D(q_e)\dot{q}_e\Big) - \frac{1}{2}\Big(\frac{\partial}{\partial q_e}\Big(\dot{q}_e^T D(q_e)\dot{q}_e\Big)\Big)^T - \frac{\partial \mathrm{V}}{\partial q_e} = \Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T \lambda + B_e u\,. \tag{1.55}$$

The integration of the dynamical equations (1.55) over the impact duration gives

$$\begin{aligned} &D(q_e)\dot{q}_e^+ - D(q_e)\dot{q}_e^- - \frac{1}{2}\int_{t^+}^{t^-}\Big(\frac{\partial}{\partial q_e}\Big(\dot{q}_e^T D(q_e)\dot{q}_e\Big)\Big)^T \mathrm{d}t - (V(q_e^+) - V(q_e^-)) \\ &\quad = \int_{t^+}^{t^-}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T \lambda\,\mathrm{d}t + B_e\int_{t^+}^{t^-} u\,\mathrm{d}t\,, \end{aligned} \tag{1.56}$$

where t^+ (resp. t^-) is the end (resp. beginning) of the impact.

Assuming that inputs cannot generate impulses, there exits $M_u \in \mathbb{R}^+$ such that

$$\Big|\int_{t^+}^{t^-} u\,\mathrm{d}t\Big| \leq (t^- - t^+)M_u\,. \tag{1.57}$$

Since $(q_e, \dot{q}_e) \to \frac{\partial}{\partial q_e}\Big(\dot{q}_e^T D(q_e)\dot{q}_e\Big)$ is a continuous function, then there exists M such that

$$\Big|\frac{1}{2}\int_{t^+}^{t^-}\Big(\frac{\partial}{\partial q_e}\Big(\dot{q}_e^T D(q_e)\dot{q}_e\Big)\Big)^T \mathrm{d}t\Big| \leq (t^- - t^+)M\,. \tag{1.58}$$

Then, reducing to zero the duration of the impact, that is setting $t^- = t^+$ gives what is called *the equation of conservation of momentum*

$$D(q_e)\dot{q}_e^+ - D(q_e)\dot{q}_e^- = \left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T I\,, \tag{1.59}$$

where $I = lim_{t^- \to t^+} \int_{t^-}^{t^+} \lambda \, \mathrm{d}t$ is called the *ground reaction impulse*.

Actually, the following derivation of the equation of conservation of momentum, even though intuitive, is not rigorous. The reader can find a more complex, but correct presentation, using the theory of distributions, chapter 1 of [13].

Under the hypothesis on the continuity of the position, we know that $q_e^+ = q_e^- = q_e$. The equation of conservation of momentum gives N equations with $N + p^+$ unknowns ($\dot{q}_e^+$ and I). But, the constraint stating that $\dot{q}_e^+$ belongs to $\mathcal{TQ}_e^{\eta^+}$ (1.54) brings p^+ additional equations. This allows to solve for $\dot{q}_e^+$ and I. We have

$$\begin{cases} D_e(q_e)\dot{q}_e^+ - D_e(q_e)\dot{q}_e^- = \left(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\right)^T I \\ \dfrac{\partial \eta^+}{\partial q_e}(q_e)\dot{q_e}^+ = 0_{p^+ \times 1} \end{cases} \tag{1.60}$$

Assuming that $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank, and solving for the system (1.60) , we get

$$\begin{cases} \dot{q}_e^+ = \Delta_{\dot{q}_e}(q_e)\dot{q}_e^- \\ I = \Delta_I(q_e)\dot{q}_e^- \end{cases} \tag{1.61}$$

with

$$\Delta_I(q_e) = -\left(\frac{\partial \eta^+}{\partial q_e}(q_e) D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right)^{-1} \frac{\partial \eta^+}{\partial q_e}(q_e) \tag{1.62}$$

$$\Delta_{\dot{q}_e}(q_e) = I_{(N+6)\times(N+6)} + D_e(q_e)^{-1}\left(\frac{\partial \eta_1^+}{\partial q_e}(q_e)\right)^T \Delta_I(q_e) \tag{1.63}$$

And the impact map is

$$\begin{cases} q_e^+ = q_e^- \\ \dot{q}_e^+ = \Delta_{\dot{q}_e}(q_e)\dot{q}_e^- \end{cases} \tag{1.64}$$

1.4.2 Validity of The Impact Model

The impact model is not necessarily valid. Indeed, it was built assuming that the holonomic constraints $\eta^+(q_e) = 0$ are verified "during" the impact. Then, several hypotheses on the ground reaction impulse I and on the post-impact velocity $\dot{q}_e^+$ (1.61) must be checked.

1.4.2.1 Conditions on the Ground Reaction Impulse I

The impulsive contact wrench I must satisfy the same hypotheses as the standard ground reaction wrenches, such as the no slipping and the no take off conditions (see 1.3.5).

Especially, let us assume that the foot 1 is in contact with the ground during the impact and immobile. Let

$$I^j_{Foot1} = lim_{t^- \to t^+} \int_{t^-}^{t^+} \mathcal{F}^j_{Foot1} \, \mathrm{d}t \tag{1.65}$$

be the component of the ground reaction impulse associated to the ground reaction force along axis j exerted on the foot 1 during the impact, and denoted $\mathcal{F}^j_{Foot1}$. Then, so that the foot does not take off, one must have

$$I^z_{Foot1} > 0 \, . \tag{1.66}$$

Furthermore, to avoid slipping, using the standard Coulomb model, the ground reaction impulse must stay in the friction cone, i.e.

$$\sqrt{(I^x_{Foot1})^2 + (I^y_{Foot1})^2} < \mu I^z_{Foot1} \, . \tag{1.67}$$

Additionally, the ZMP conditions must be checked.

1.4.2.2 Conditions on the Post-Impact Velocity

The post-impact velocity $\dot{q}_e^+$ must be consistent with the assumed foot-ground contact $\eta^+(q_e) = 0$. For instance, if the robot is assumed to be in single support phase after impact, the vertical velocity of the former stance leg must be upward after impact.

1.4.2.3 Impact Model Hypotheses

The hypotheses of the impact model presented above can be formulated as constraints on the state of the robot before impact $(q_e^-, \dot{q}_e^-)$, and gathered into the vector of constraints

$$C_\Delta^{\eta^+}(q_e^-, \dot{q}_e^-) \geq 0 \, . \tag{1.68}$$

1.4.2.4 Determining the Correct Foot-Ground Contact After Impact

Note that there exist several possible phases after impact. For instance, the robot may be in single support phase, or in double support phase after impact. Actually, for a given set of holonomic constraints, $\eta^-(q_e) = 0$, and a state x_e^- before impact, the nature of the foot-ground contact after impact, described by the set of holonomic constraints $\eta^+(q_e) = 0$, is not known a priori.

Finding the proper phase is an iterative process, as explained in [56]. A model under the form (1.61) is derived for each possible set of holonomic constraints $\eta^+(q_e) = 0$. The set of holonomic constraints satisfying the model hypotheses

(1.68) is selected as the good one. This process cannot be done analytically, but numerically.

But, there is no guarantee that a single set of holonomic constraints after impact $\eta^+(q_e) = 0$ satisfy (1.68). Then, if none or several of them verify the impact model hypotheses, the impact model does not allow to conclude on the behavior of the robot after impact. This is a limitation of the model. For instance, Mu and Wu [78] showed that there exist states for which the impact model predict both a single and a double support phase for the planar five-link walker with point feet.

Remarks:

- In the particular case where $\frac{\partial \eta^+}{\partial q_e}(q_e)\dot{q}_e^- = 0$, that is when the holonomic constraints are modified without collision, then there is no jump on the velocity ($\dot{q}_e^+ = \dot{q}_e^-$). Additionally, $\Delta_I(q_e)\dot{q}_e^- = 0$, and therefore $I = 0$. Hence, $\dot{q}_e^+$ and I are consistent with the holonomic constraints after impact. They describe a correct foot-ground contact after impact.

- The impact dynamics was derived on the impact surface $\mathcal{S}_{\eta^- \to \eta^+}$. But, since no particular property of this surface was used, this derivation is valid in the whole state space $\mathcal{TQ}_e^{\eta^-}$. This is useful since the impact may occur at any state, as it may happen when the robot evolves on a rough terrain.

1.4.3 Kinetic Energy Variation at Impact

The mechanical energy of the robot E_e, is the sum of the kinetic energy K_e and the potential energy V_e. We assume that the potential energy V_e depends only on the position q_e. This is generally the case for biped robots. Then

$$\mathrm{E}_e(q_e, \dot{q}_e) = \mathrm{K}_e(q_e, \dot{q}_e) + \mathrm{V}_e(q_e)\,, \tag{1.69}$$

with

$$\mathrm{K}_e(q_e, \dot{q}_e) = \frac{1}{2}\dot{q}_e^T D_e(q_e)\dot{q}_e\,. \tag{1.70}$$

At impact, given that the position q_e is assumed to be preserved, the potential energy is unchanged. Then, the mechanical energy variation at impact is equal to the kinetic energy variation

$$\mathrm{E}_e(q_e, \dot{q}_e^+) - \mathrm{E}_e(q_e, \dot{q}_e^-) = \frac{1}{2}\dot{q}_e^{+T} D_e(q_e)\dot{q}_e^+ - \frac{1}{2}\dot{q}_e^{-T} D_e(q_e)\dot{q}_e^-\,. \tag{1.71}$$

Multiplying the equation of conservation of momentum (1.59) by $\dot{q}_e^+$ and $\dot{q}_e^-$, and using (1.54) gives

$$\begin{cases} \dot{q}_e^{+T} D_e(q_e)\dot{q}_e^+ - \dot{q}_e^{+T} D_e(q_e)\dot{q}_e^- = \dot{q}_e^{+T}\left(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\right)^T I = 0 \\ \dot{q}_e^{-T} D_e(q_e)\dot{q}_e^+ - \dot{q}_e^{-T} D_e(q_e)\dot{q}_e^- = \dot{q}_e^{-T}\left(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\right)^T I \end{cases} \tag{1.72}$$

Adding the 2 equations leads to

$$\begin{aligned} \mathrm{K}_e(q_e, \dot{q}_e^+) - \mathrm{K}_e(q_e, \dot{q}_e^-) &= \frac{1}{2}\dot{q}_e^{-T}\Big(\frac{\partial \eta}{\partial q_e}(q_e)\Big)^T I \\ &= \frac{1}{2}\Big(\frac{d}{dt}\Big(\eta^+(q_e)\Big)\Big)^T I \end{aligned} \tag{1.73}$$

Hence, without surprise, the kinetic energy variation is proportional to the "work" of the impulse I.

Using the expression of I in (1.62), the kinetic energy variation is

$$\begin{aligned} &\mathrm{VarK}_e(q_e^-, \dot{q}_e^-) = \\ &-\frac{1}{2}\dot{q}_e^{-T}\left[\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\left(\frac{\partial \eta^+}{\partial q_e}(q_e)D_e(q_e)^{-1}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\right)^{-1}\frac{\partial \eta^+}{\partial q_e}(q_e)\right]\dot{q}_e^- \,. \end{aligned} \tag{1.74}$$

Since $\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)D_e(q_e)^{-1}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T$ is positive definite

$$\mathrm{VarK}_e(q_e^-, \dot{q}_e^-) \leq 0\,. \tag{1.75}$$

Then, the kinetic energy level cannot increase at impact, it can only remain constant or decrease, which is physically consistent.

1.4.3.1 Expression of the variation of energy in the reduced coordinates.

Instead of using the set of positions q_e that are not independent each others due to the p^- holonomic constraints $\eta^-(q_e) = 0$, one can use a set of $N - p^-$ coordinates $q \in \mathcal{Q}_f^{\eta^-} \subset \mathbb{R}^{N-p^-}$ (see Section 1.3.4.1). The mapping between q and q_e is given by the embedding

$$\begin{cases} E^{\eta^-} : \mathcal{Q}_f^{\eta^-} \to \mathcal{Q}_e^{\eta^-} \\ \qquad\quad q \;\to\; q_e = E^{\eta^-}(q) \end{cases} \tag{1.76}$$

We have $E^{\eta^-}(\mathcal{Q}_f^{\eta^-}) = \mathcal{Q}_e^{\eta^-}$, and $(q, \dot{q})$ defines a set of $2(N - p^-)$ independent coordinates for the system subject to the holonomic constraints $\eta^-(q_e) = 0$.

From (1.5), and using the the embedding, the expression of the kinetic energy in the reduced coordinates $(q, \dot{q}) \in \mathcal{TQ}_f^{\eta^-}$ is

$$\mathrm{K}(q, \dot{q}) = \frac{1}{2}\dot{q}^T D(q)\dot{q}\,, \tag{1.77}$$

where

$$D(q) = \Big(\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)^T D_e(q_e)\Big(\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)\,, \tag{1.78}$$

with $q_e = E^{\eta^-}(q)$.

From (1.74), the expression of the variation of energy in the reduced coordinates is

$$\mathrm{VarK}(q, \dot{q}) := \frac{1}{2}\dot{q}^T A(q)\dot{q}\,, \tag{1.79}$$

where

$$\begin{aligned} A(q) = -\Big(\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)^T \Bigg[\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T \Bigg(\frac{\partial \eta^+}{\partial q_e}(q_e) D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\Bigg)^{-1} \\ \times \Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)\Bigg]\Big(\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)\,, \end{aligned} \tag{1.80}$$

with $q_e = E^{\eta^-}(q)$.

1.4.3.2 Discussion

From (1.74), one can easily remark that there is no energy dissipation at impact if and only if

$$\frac{\partial \eta^+}{\partial q_e}(q_e)\dot{q}_e^- = 0\,. \tag{1.81}$$

This is in agreement with the analysis performed in Section 1.4.1.1, where we explained that a jump on the velocity was necessary when $\frac{\partial \eta^+}{\partial q_e}(q_e)\dot{q}_e^- \neq 0$.

Note that

$$v = \frac{d}{dt}\Big(\eta^+(q_e)\Big) = \frac{\partial \eta^+}{\partial q_e}(q_e)\dot{q}_e\,, \tag{1.82}$$

is the velocity of the degrees of freedom which are locked after impact. Hence, an other point of view, is to say that there is no dissipation of energy if and only if this velocity is equal to zero just before the impact. For example, a biped with a flat foot walk dissipates no energy at impact if and only if it lays its foot flat down on the ground with no velocity.

1.5 Modeling Biped Robots as Hybrid Systems

From the modeling that we adopted, a biped robot is ruled by continuous dynamics and discrete dynamics. It is an *hybrid system* [104]. The discrete dynamics allows to make the transition between the continuous models.

The *hybrid model* of the robot is:

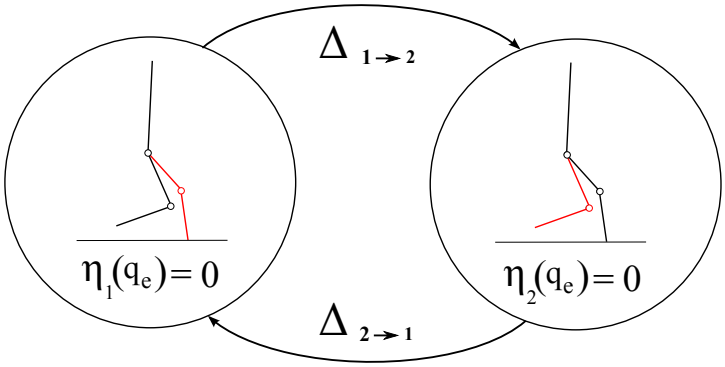


Figure 1.6: Diagram representing the two phases of a walk with an instantaneous double support phase for a planar biped with point feet. The two phases are: "leg 1 is the stance leg ($\eta_1(q_e) = 0$)" and "leg 2 is the stance leg ($\eta_2(q_e) = 0$)".

If $x_e \in D_{\eta_i}$:

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e) + G_e(q_e) = B_e u + (\dfrac{\partial \eta_i}{\partial q_e}(q_e))^T \lambda \\ \eta_i(q_e) = 0 \end{cases} \tag{1.83}$$

If $x_e \in \mathcal{S}_{\eta_i \to \eta_j}$:

$$x_e^+ = \Delta_{\eta_i \to \eta_j}(x_e)\,, j \in J \subset \mathbb{N}\,, \tag{1.84}$$

where the set $\{\eta_j(q_e) = 0, j \in J \subset \mathbb{N}\}$ corresponds to all the holonomic constraints the system may switch to.

Figure 1.6 illustrates the hybrid model of a walk composed of two phases: leg 1 flat on the ground and leg 2 flat on the ground. In this case, two continuous and two discrete models decribe the dynamics of the system. Considering only these two phases is a common choice in the HZD method (see next chapter). In the case where the legs of the robot are symmetric, the model of the robot when the two continuous models can be deduced from each other by a simple coordinates relabeling: leg 1 $\to$ leg 2, leg 2 $\to$ leg 1. Then, the hybrid model can be written as one continuous and one discrete dynamics (see [127] for further details)

$$\begin{cases} \dot{x}_e = f_e(x_e) + g_e(x_e)u, & x \notin \mathcal{S} \\ x_e^+ = \Delta(x_e), & x_e \in \mathcal{S} \end{cases} \tag{1.85}$$

where $\mathcal{S}$ is the impact surface, and Δ the impact map. The hybrid model can also be derived in the reduced coordinates.

1.6 Considering the Inertia of a Rigid Mechanical Transmission System

An other modeling aspect to be taken into account for practical applications is the consideration of the inertia of the mechanical transmission systems in the

dynamical equations. In this section, we discuss this issue. Generally, actuators are not directly mounted on the joints of robots. Mechanical components, named *mechanical transmission system* or *drivetrain*, are placed between the motor rotor shaft and the joint for 2 reasons:

- To transfer the motion from the actuator to the joint, when mechanical design constraints prevent from mounting the actuator directly on the joint.
- To increase the torque produced by the actuator by means of components such that gear reducers or pulley belt systems.

The ratio between the output speed and the input speed of the mechanical transmission system is called *transmission ratio* or *gear ratio*. We will assume that the elasticity of the components is negligible (rigid transmission), and the mechanical backlash too. Under these assumptions, for most of mechanical transmission systems, the transmission ratio is constant. From now on, we will consider that this is the case.

We show here that the transmission inertia cannot be neglected in the case of a high transmission ratio. We then present the exact and approximate ways of proceeding to consider it in the dynamical equations. Adding the transmission inertia times the square of the transmission ratio to the diagonal terms of the mass matrix is generally enough.

1.6.1 Why Considering the Inertia of a Mechanical Transmission System?

We highlight the importance of considering the Inertia of a mechanical transmission system using the example of a simple joint. Let us consider a mechanical transmission system assumed to be fixed in the world frame. See for example figure 1.7. Let q_M be the actuator position (input of the mechanical transmission system), and q be the joint position (output of the mechanical transmission system). We denote I_{load} the inertia of the load attached to the joint and I_T the sum of the inertia of the mechanical components of the transmission. The transmission ratio is denoted r. Then, we have

$$\dot{q}_M = r\dot{q} \text{ and } u = ru_M \,, \tag{1.86}$$

where u is the joint torque and u_M is the actuator torque. Note that if $r > 1$, the mechanical transmission system diminishes the velocity, but increases the torque by the same factor.

Since the mechanical transmission system is assumed to be in an inertial frame, the kinetic energy of the system is

$$\begin{aligned} K &= \frac{1}{2} I_{Load} \dot{q}^2 + \frac{1}{2} I_T \dot{q_M}^2 \\ &= \frac{1}{2} \Big(I_{Load} + r^2 I_T \Big) \dot{q}^2 \,. \end{aligned} \tag{1.87}$$

The quantity

$$I_{eq} = I_{Load} + r^2 I_T \tag{1.88}$$

is called the *reflected inertia to joint*. This is the inertia viewed from the joint, that is the inertia felt by an operator when moving the load. Note that the inertia of the transmission is multiplied by a factor r^2.

Generally, the inertia of the transmission components I_T is small compared to I_{Load}. Then, common sense would incite us to neglect I_T. However, from the expression of the reflected joint inertia (1.88), when the transmission factor r is large, the effects of the transmission on the kinetic energy, and so on the dynamics, cannot be neglected. They may even dominate the effects of the load inertia!

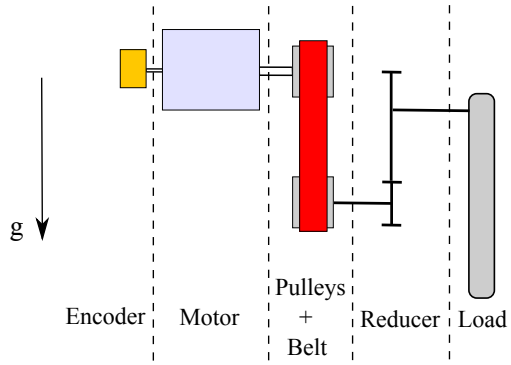


Figure 1.7: Example of a mechanical transmission system.

The dynamical equations of the load with the mechanical transmission system are obtained using the Lagrange's equations

$$I_{eq}\ddot{q} = u + \Gamma_e \,, \tag{1.89}$$

where Γ_e is the vector of the torques exerted on the system (gravity, friction,...). Then, for this simple case, modeling the effects of the transmission consists in adding the term $r^2 I_T$ in the expression of the (scalar) mass matrix I_{Load}.

1.6.2 How to Consider the Mechanical Transmission System ?

We have highlighted the importance of considering the inertia of a transmission system in case of a high transmission factor. An illustration was given for the simple case of a mechanical transmission system fixed in the inertial frame. Unfortunately, for more complex systems, like for biped robots, this hypothesis is not true anymore, and modeling exactly the effects of the transmission is harder. We explain here how to consider it exactly and by making some classical approximations. One can both use the extended or the reduced coordinates. For the sake of simplicity, we will use the reduced coordinates.

1.6.2.1 Direct drive model

As a preliminary, we introduce the model that does not take into account the dynamics of the transmission, sometimes called *direct drive model*. In this case, the mechanical transmission system is assumed to be rigidly fixed to the link to which the actuator is attached. Thus, the kinetic energy due to the rotationnal motion of the transmission system is not considered. The dynamical equations are obtained using the equations of Lagrange. Thus

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = Bu\,. \tag{1.90}$$

1.6.2.2 Considering the Transmission Mechanical System Exactly

The derivation of the exact dynamical equations considering the transmission inertia is described using the Lagrangian formalism and the Newton Euler's formalism in [127] ((B.222) page 434), [71] and [18]. Each mechanical component (rotor of the motor, transmission components) is considered as a separate entity. For the Lagrangian formalism, the kinetic and potential energy for each component are calculated. This allows to compute the Lagrangian and to get the equations of motion using the Lagrange's equations. The advantage of this approach is to make no approximations. But, it results in more complex dynamics equations, increasing the computation cost when they are numerically integrated. That's why an approximated model is often used in practice.

1.6.2.3 Considering the Transmission Mechanical System Approximately

Given that the exact model leads to complex equations, a common assumption is to neglect the coupling effects between the transmission and the link motion (page 201 of [64]). Thus, it is assumed that the mechanical transmission system is in an inertial frame. Under this assumption, we fall into the simple case presented before. Then, the kinetic energy of the transmission is $\frac{1}{2}r^2 I_T \dot{q}^2$. No proof of validity of this approximation has been found in the litterature. But, it is commonly used and considered valid in case of high transmission ratio. This model was especially used for the robot RABBIT (page 218 in [127]). [18] compares the exact model with the approximate model for a specific model of robot and conclude that the approximate model is satisfactory.

Then, the new model is obtained from the direct drive model by adding the reflected transmission inertia to joint

$$I_{eq} = \begin{pmatrix} I_{eq,1} & 0 & \cdots & 0 \\ 0 & I_{eq,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I_{eq,n} \end{pmatrix}, \tag{1.91}$$

where $I_{eq,i}$ is the reflected transmission inertia to the joint parameterized by q_i,

the i^{th} component of vector q.

Hence, the dynamical equations are

$$\tilde{D}(q)\ddot{q} + C(q,\dot{q}) + G(q) = Bu\,, \tag{1.92}$$

with

$$\tilde{D}(q) = D(q) + I_{eq}\,. \tag{1.93}$$

If q_i is an actuated coordinate, $I_{eq,i}$ is

$$I_{eq,i} = r_i^2 I_{T,i}\,, \tag{1.94}$$

where $I_{T,i}$ is the inertia of the transmission for the joint parameterized by q_i, and r_i is the transmission factor.

If q_i is an unactuated coordinate

$$I_{eq,i} = 0. \tag{1.95}$$

1.6.3 Decoupling Effect of High Transmission Ratios

We show here that mechanical transmission systems have the interesting property of approximately decoupling the actuated dynamics.

To show this, we partition the generalized positions q into the actuated positions q_a and the unactuated positions q_u, i.e. $q = (q_a; q_u)$. The dynamics expressed in joint level can be written as

$$\begin{aligned}&\begin{pmatrix} D_{11}(q) + r^2 I_T & D_{12}(q) \\ D_{21}(q) & D_{22}(q) \end{pmatrix} \begin{pmatrix} \ddot{q}_a \\ \ddot{q}_u \end{pmatrix} + \begin{pmatrix} C_{1,1}(q,\dot{q}) & C_{1,2}(q,\dot{q}) \\ C_{2,1}(q,\dot{q}) & C_{2,2}(q,\dot{q}) \end{pmatrix} \begin{pmatrix} \dot{q}_a \\ \dot{q}_u \end{pmatrix} + \begin{pmatrix} G_1(q) \\ G_2(q) \end{pmatrix} \\ &\quad = \begin{pmatrix} r u_M \\ 0 \end{pmatrix} + \begin{pmatrix} \Gamma_{f,1} \\ \Gamma_{f,2} \end{pmatrix}\,,\end{aligned} \tag{1.96}$$

where Γ_f is the vector of the generalized efforts corresponding to the perturbations exerted on the system (joint friction, external force,...), u_M ($u = r u_M$) is the vector of motor torques, I_T is a diagonal matrix gathering the transmission inertia for each transmission chain expressed at motor level, and $r \in \mathbb{N}^*$ is the transmission reduction factor. For the sake of clarity, we assume here that the reduction factor is the same for each joint.

The equations of motion give

$$\begin{cases} I_T\ddot{q}_a + \dfrac{D_{11}(q)}{r^2}\ddot{q}_a + \dfrac{D_{12}(q)}{r^2}\ddot{q}_u + \dfrac{C_{1,1}(q,\dot{q})}{r^2}\dot{q}_a + \dfrac{C_{1,2}(q,\dot{q})}{r^2}\dot{q}_u + \dfrac{G_1(q)}{r^2} \\ \qquad = \dfrac{u_M}{r} + \dfrac{\Gamma_{f,1}}{r^2}. \\ D_{21}(q)\ddot{q}_a + D_{22}(q)\ddot{q}_u + C_{2,1}(q,\dot{q})\dot{q}_a + C_{2,2}(q,\dot{q})\dot{q}_u + G_2(q) = \Gamma_{f,2}\,. \end{cases} \tag{1.97}$$

Considering that the system describes a given motion $t \to (q(t), \dot{q}(t))$ with the motor torques $t \to u_M(t)$, we have

$$u_M \sim r I_T \ddot{q}_a \text{ as } r \to \infty \,. \tag{1.98}$$

This motivates the approximative dynamic model

$$\begin{cases} r^2 I_T \ddot{q}_a = u \\ D_{21}(q)\ddot{q}_a + D_{22}(q)\ddot{q}_u + C_{2,1}(q,\dot{q})\dot{q}_a + C_{2,2}(q,\dot{q})\dot{q}_u + G_2(q) = \Gamma_{f,2} \,. \end{cases} \tag{1.99}$$

Therefore, the dynamics of the actuated coordinates are approximately decoupled. The time-varying nonlinear inertia of the load, and the external perturbations are neglictible in comparison to the time-invariant inertia of the transmission $r^2 I_T$.

This property is interesting for the design of controllers. This especially proves that the use of a decoupled controller like a high gain PD controller on the actuated coordinates may give good stabilization results. See Section 2.4.7.2 for further details.

1.7 Others Modeling Aspects

In this dissertation, we used the previous modeling to derive control laws in theory and to test them in simulation. Actually, this is a "simplified modeling", since it ignores some modeling aspects that may be not negligible, such as the compliance of the ground, joint friction, or mechanical backlash. Considering these aspects complicates the model and slows down simulations.

Then, it is preferable to investigate a posteriori their influence on the results obtained using the "simplified modeling" in simulation. If the numerical results obtained using the "sophisticated model" are close to those obtained the with "simplified model", then the latter model is enough. This is the approach used in this dissertation. We especially investigated the effects of the compliance of the ground, and joint friction. Especially, see Sections 3.4 and 4.4.5.1.

1.7.1 A Compliant Ground Model

The previous modeling assumes that the foot-ground contact is rigid. Under this assumption, an expression of the ground reaction wrenches can be obtained by differentiating the holonomic constraints. This model has the advantage to be simple and to give an analytical expression of the ground reaction wrenches (see (1.31)). But, in reality, the foot-ground contact is compliant, that is the foot and the ground are being slightly deformed due to the contact.

To be closer to the reality, a compliant ground model, introduced by Plestan et al. [88], can be used. In the compliant ground model, there is no assumption on the equivalent kinematic link modeling the foot-ground contact. Then, holonomic constraints are not used anymore, and with this compliant model, the system is modeled as a continuous and not as an hybrid system.

We consider here only the case of a planar biped robot with point feet. See [16] for a more general case. The dynamics of the robot is

$$D_e(q_e)\ddot{q}_e + C_e(q_e,\dot{q}_e)\dot{q}_e + G(q_e) = B_e u + \left(\frac{\partial p_1}{\partial q_e}(q_e)\right)^T F_1 + \left(\frac{\partial p_2}{\partial q_e}(q_e)\right)^T F_2\,, \quad (1.100)$$

where $F_1 = (F_1^N; F_1^T)$ (resp. $F_2 = (F_2^N; F_2^T)$) is the vector gathering the normal (overscript "N") and tangential (overscript "T") components of the ground reaction force acting on the end of leg 1 P_1 (resp. leg 2 P_2). p_1 (resp. p_i) is the vector of the cartesian position of the extremity of the leg 1 (resp. 2).

The foot ground-contact interactions are assumed to be as those of a mass spring damper system. The ground reaction forces are calculated in the following way. Let z_G be the penetration depth, i.e. the altitude of one foot relative to the altitude of the non deformed ground. The ground reaction force $F = (F^N; F^T)$ acting on the foot is zero when $z_G > 0$ and is given by the following formula when $z_G \leq 0$

$$\begin{cases} F_N = -\lambda_v |z_G|^n \dot{z}_G + k|z_G|^n \\ F_T = -\mu(d,v)|F_N| \end{cases} \quad (1.101)$$

$\mu(d,v)$ is the coefficient of friction. A possible model for it is obtained by integrating the differential equation (LuGre model)

$$\begin{cases} \dot{d} = v - |v|\dfrac{\sigma_{h0}}{\alpha_{h0}} d \\ \mu(v,d) = \sigma_{h0} d + \sigma_{h1}\dot{d} + \alpha_{h2} v \end{cases} \quad (1.102)$$

where v is the horizontal velocity of the foot relative to the ground, and d is an internal state. For the robot RABBIT, the friction coefficient was saturated to 0.7 to allow the foot to slip. The parameters of the model depend on the geometry of the feet and the type of material used for the feet and the ground. For the simulations presented in this dissertation, they were first taken from the model of the robot RABBIT [88], and then adjusted to obtain a behavior close to the one observed experimentally for the robot VS

$$\begin{aligned} &\lambda_v = 9\,000\,000\,, k = 6\,000\,000\,, n = 1.5\,, \\ &\sigma_{h0} = 260\,, \sigma_{h1} = 0.6\,, \alpha_{h0} = 0.285\,, \alpha_{h2} = 0.18\,. \end{aligned} \quad (1.103)$$

Note that this model does not allow to obtain an analytical expression of the ground reaction forces, since a differential equation has to be integrated. This model requires more computer resources than the rigid ground model. Then, it is not suited for motion planning and control design. But, it can be used a posteriori to evaluate the performances of the controllers obtained with the rigid ground model, prior to an experimental implementation.

1.7.2 Joint Friction

Due to the interaction between the mechanical components of the mechanical transmission system, friction phenomena appear. A complete review on the friction models can be found in [85, 12]. Friction phenomena can be modeled as a friction torque at the joint level. A standard model for the joint friction torque in robotics is

$$\Gamma_f = -F_s sign(\dot{q}) - F_v \dot{q}\,, \tag{1.104}$$

where $F_s > 0$ is the *Coulomb static friction term*, $F_v > 0$ is *the viscous friction term*, and $\dot{q}$ is the joint velocity [63]. Friction compensation is commonly realized by identifying the model (1.104) [113].

1.7.3 Mechanical Backlash and Joint Flexibility

Phenomena of mechanical backlash and elasticity may exist in the mechanical transmission system, that is the actuator can move without moving the corresponding joint, and vice versa. If these phenomena are too important, the relations (1.86) are uncorrect. A modeling of joint flexibly can be found in [122].

Chapter 2

Literature on the Control of Biped Robots

Résumé français Ce chapitre fait une revue des approches les plus utilisées pour le contrôle de robots bipèdes. Nous en avons recensé trois :

- La méthode de "marche pied plat" ou "marche ZMP" [59].
- La méthode dite de "Foot placement" [94, 92] .
- La méthode des contraintes virtuelles et de la dynamique des zéros hybride (méthode HZD) [127].

Chacune de ces méthodes utilise certaines propriétés caractéristiques de la marche humaine. La méthode de "marche pied plat" utilise le fait que durant une partie de la phase d'appui, l'humain a le pied posé à plat au sol, et est donc entièrement actionné. Cette méthode utilise des modèles de type pendule inverse pour planifier des trajectoires de marche en ligne. La méthode de "Foot placement" utilise aussi de tels modèles pour planifier en ligne où le robot doit placer son pied de vol au sol pour garder son équilibre. Il est en effet bien connu que le placement du pied joue un rôle crucial dans la locomotion bipède. Enfin, la méthode HZD utilise le fait que la marche sur sol plat est un mouvement périodique, et que les bipèdes ont la propriété de converger naturellement vers un cycle de marche périodique (limit cycle walking).

De nombreuses contributions ont été réalisées ces dernières années, ce qui peut rendre difficile de savoir quelle approche offre les meilleures performances. Nous donnons ici notre propre analyse de l'état de l'art, et rassemblons les informations que nous avons jugées pertinentes. Nous nous attardons en particulier sur la méthode HZD, puisque nous avons trouvé que c'est la méthode la plus documentée et la plus rigoureuse pour le contrôle des robots bipèdes.

2.1 Introduction

This chapter presents a review on the most popular approaches for the control of biped robots. We identified three main methods for the control of biped robots:

- Flat foot walking or ZMP walking [59].
- Foot Placement [94, 92] .
- The method of virtual constraints and hybrid zero dynamics (HZD method) [127].

Each of these three methods focuses on some specific properties of human walking. Flat foot walking uses the fact that during a period of the stance phase, human has the foot flat on the ground and is then an entirely actuated system. It also exploits the inverted pendulum-like dynamics of human to realize online motion planning. Foot placement uses inverted pendulum-like models to online estimate where the robot should place its swing foot to preserve its balance. Actually, the placement of the foot plays a crucial role in biped locomotion, especially for preventing a fall. Finally, the HZD method exploits the fact that walking on flat ground is a periodic motion and that bipeds have the natural property to converge to periodic limit cycles (limit cycle walking).

Note that this classification may be somewhat arbitrary, since some works do not claim to belong to one of these three categories. Additionally, recent works [25, 27] blur the line between them. Actually, these three methods, even though exploiting different aspects of human walking, do not exclude each other. Unifying them is probably a good way to get agile and versatile biped robots.

Many contributions have been made across the years, which may introduce confusion to know which approach offer the best performances. We give here our own analysis of the-state-of-the-art and gather the information that we could find and that we judged to be relevant. We especially focus our attention to the HZD method since we found it is the most documented and rigorous approach for the control of underactuated robots.

Remark: A classical framework commonly used in Automation consists in dividing the problem of the control of a system into 2 steps:

- **Motion planning**. Motion planning consists in finding a trajectory that the system must follow to achieve a desired behavior, for example going from point A to point B, walking without falling... Ideally, motion planning should be realized online, which may greatly improve the reactiveness and robustness of the system. But, due to the computational burden, this not always possible, especially for complex systems like biped robots. Motion planning is generally the hardest step and solved online.
- **Trajectory stabilization**. Trajectory stabilization consists in designing a controller that stabilizes the system around the planned trajectory, even in the presence of disturbances and modeling errors. It is realized online.

This framework is used in the method of ZMP and of foot placement. We will see that the HZD method does not exactly follow this approach.

2.2 Flat Foot Walking or ZMP Walking

Flat foot walking, also called ZMP walking, consists in designing a walking controller constraining the robot to always keep its feet flat on the ground. Since the feet are flat on the ground, the system is then entirely actuated which greatly facilitates its control. This approach has been successfully implemented for the first time in 1984 on the japanese robot WL-10RD by Ichiro Kato et al. of the Waseda University. Since then, it has become a popular approach and has been widely used on humanoid robots (ASIMO [54], HUBO [123], HRP2 [61], NAO [44]...).

2.2.1 The Zero Moment Point

A sufficient, but not necessary, condition to keep the feet of the robot flat on the ground, is that the center of pressure of the ground reaction forces, called ZMP (Zero Moment Point) be strictly in the convex hull of the contact points with the ground. The convex hull is called *support polygon* (see figure 2.1). If the ZMP reaches the boundary of the support polygon, the feet may risk to rotate around its edges. This was first highlighted by M. Vukobratovic in 1968 [120]. The denomination Zero Moment Point comes from the fact that this is the point on the floor where the horizontal components of the moment of the ground reaction forces is zero.

The ZMP condition is less restrictive than the condition of static balance that was used for the first humanoid robots and for toys. A robot is said to be statically balanced if the projection of its center of mass onto the ground is always strictly inside the support polygon. Then, the robot does not fall if it suddenly stops.

Assuming that the biped is moving on an horizontal ground, from Wieber et al. (Chapter 48 in [128]), the following dynamic equation gives the relation existing between the ZMP and the center of mass of the robot

$$\begin{pmatrix} x \\ y \end{pmatrix} - \frac{z}{\ddot{z}+g}\begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} + \frac{1}{m(\ddot{z}+g)}\begin{pmatrix} -\dot{L}^y \\ \dot{L}^x \end{pmatrix} = \frac{1}{\sum_i f_i^z}\sum_i f_i^z \begin{pmatrix} p_i^x \\ p_i^y \end{pmatrix}, \qquad (2.1)$$

where $(x; y; z)$ is the position of the center of mass of the robot, $(L^x; L^y)$ is the vector of the horizontal components of the angular momentum of the robot, $p_i = (p_i^x; p_i^y; 0)$ are the contact points of the robot with the ground, and f_i^z is the vertical component of the ground reaction forces exerted on p_i. The right hand side of the equation is the center of pressure of the ground reaction forces, or ZMP. Since $f_i^z \geq 0$, it corresponds to the barycenter of the contact points p_i weighted by f_i^z, and so necessarily belongs to the convex hull of the contact points, i.e. to the support polygon.

2.2.2 Designing Flat Foot Walking Gaits

The literature is rich on how to generate offline and online walking gaits that respect the ZMP condition. But, it is not always clear on how the experimental

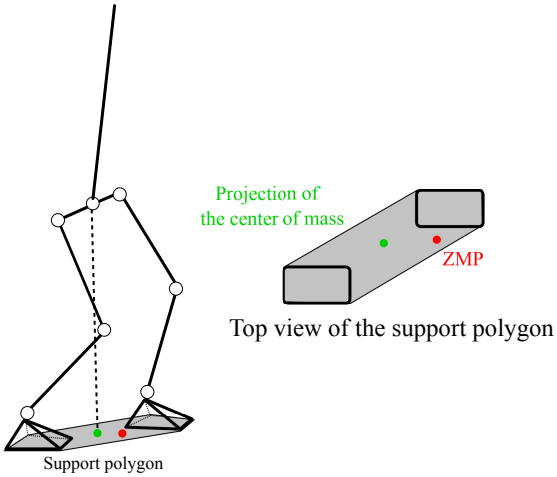
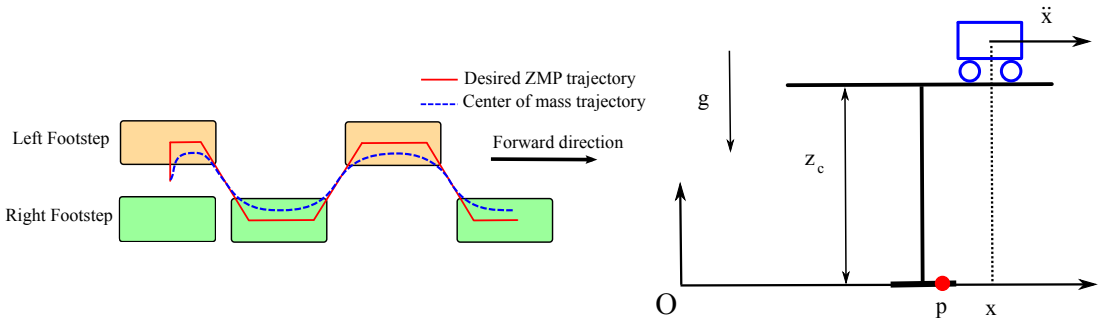


Figure 2.1: Illustration of the support polygon for a biped robot in double support phase. When the robot is immobile, the ZMP and the projection of the center of mass onto the support polygon are the same. But, when the acceleration of the robot is not zero, their locations are generally different.

gaits are achieved. A simplified model of the robot is generally used to generate walking gaits online [105, 62, 60, 115]. It is often based on the dynamics of an inverted pendulum.

2.2.2.1 Flat foot walking using the method of Kajita



(a) Illustration of a desired ZMP trajectory. (b) The cart table model.

Figure 2.2: (a) : Illustration of a desired ZMP trajectory and a trajectory for the center of mass obtained using the cart-table model [59]. (b) : The cart-table model.

A commonly used approach, that is implemented on the humanoid robot HRP2, was introduced by Kajita et al. in 2003 [59]. It follows the current methodology.

Step 1: Footstep and ZMP planning In a first step, the location of the feet on the floor is planned by possibly taking into account the environment such as obstacles. A desired trajectory of the ZMP is then derived from the footstep locations. It simply consists in drawing a line on the ground which stays inside

the support polygon defined by the footstep locations. The ZMP stays under the support in single support phase, and switches feet in double support phase. Generally, to maximize the robustness of the walk, the ZMP trajectory is chosen such that the ZMP is at any time as far as possible from the edges of the support polygon. The duration of each step, of the single and the double support phases are chosen. This defines the travel speed of the ZMP trajectory. All this process can be done by hand tuning or using software to easily adapt the ZMP trajectory to a change of parameters.

Step 2: Generation of a trajectory for the center of mass of the robot It is not obvious to directly track the ZMP trajectory using the actuators of the robot. Then, Kajita proposed to design a trajectory for the center of mass of the robot compatible with the desired ZMP trajectory. Indeed, the dynamics of the ZMP and of the center of mass of the robot are coupled (see (2.1)). Additionally, the center of mass can be easily controlled via the actuators by inverse dynamics [1].

The exact coupling of the ZMP and the center of mass is complicated (see (2.1)). This equation can be simplified by assuming that the altitude of the center of mass z is kept constant and that the angular momentum of the robot is constant ($\dot{L}^x = \dot{L}^y = 0$). Using these hypotheses (2.1) gets

$$\ddot{x} = \frac{g}{z_c}(x - p)\,, \tag{2.2}$$

where p is the position of the ZMP, z_c is the (constant) altitude of the cart and g is the gravitational constant. The center of mass of the robot is generally assumed to be located at the hip. This dynamic system is called the "linear inverted pendulum model" or "cart table model" and depicted in Figure 2.2.

The cart-table model corresponds to a cart translating on a table at a fixed altitude z_c. The cart represents the center of mass of the robot, and the table foot represents the support polygon of the robot. The table is not necessarily statically stable, since the projection of the center of mass of the cart on the ground may be outside the table foot. But, the table can be kept balanced (the table foot stays flat on the ground) through the acceleration of the cart. The dynamics (2.2) can be written under the standard form

$$\begin{cases} \dot{z} = Az + Bu \\ y = Cz \end{cases} \tag{2.3}$$

[1] Efficient inverse dynamics software exist to compute torques online: pinocchio [17], mujoco [117],...

where

$$\begin{cases} z = \begin{pmatrix} x \\ \dot{x} \\ \ddot{x} \end{pmatrix} \\ A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ C = \begin{pmatrix} 1 & 0 & -\frac{z}{g_c} \end{pmatrix} \end{cases} \tag{2.4}$$

Note that the output y is the ZMP

$$y = p\,, \tag{2.5}$$

and the input is the derivative of the acceleration, called jerk

$$u = \dddot{x}\,. \tag{2.6}$$

To generate a trajectory for the center of mass, a standard linear quadratic problem is solved with the cost function minimizing the tracking error of the ZMP and the jerk

$$J = \int_{t=0}^{t=T} Q\Big(p(s) - p^{ref}(s)\Big)^2 + Ru^2(s)\,ds\,, \tag{2.7}$$

where Q and R are positive weight matrices, y^{ref} is the reference trajectory for the ZMP, and T is the duration of the motion.

Step 3: Modification of the trajectory of the center of mass to compensate for the mismatch between the real model and the cart-table model The trajectory of the center of mass generated using this model may not exactly result in the desired ZMP trajectory when the hypotheses of the cart table model are not respected. Using the full model (2.1), the error on the ZMP can be computed: $\Delta p = p - p^{ref}$. Then, solving once again the quadratic problem of the previous paragraph (step 2), allows to find the corrections on the trajectory of the center of mass to get the desired ZMP. In practice, this dramatically increases the performances of the tracking of the desired ZMP.

Step 4: Computing the joint torques for tracking the trajectory of the center of mass The joint torques for tracking the center of mass are computed by inverse dynamics. Since the robot has generally more actuators than the dimension of x, the solution is not unique. Then, constraints can be added. Especially, to be closer to the cart-table model, it is preferable to constrain the altitude of the center of mass of the robot, generally assumed to be located at the hip, to be close to the altitude z_c. If the robot must accomplish some tasks, such as grasping or crossing a constrained environment, the tasks are translated into constraints on the state of the robot and are included into the inverse dynamics problem (see [32] and references therein).

2.2.2.2 Some Others Methods

To increase the robustness of the gait [4, 112] modify online the ZMP reference trajectory during a step. Nishiwaki and Kagami [83] propose to adjust the ZMP by using three strategies: changing the ZMP reference, changing the position of the next step, or changing the duration of the step. This allows the robot to walk through complex terrain. It is worth mentioning [126] that, contrary to the previously mentioned works, uses the HZD method (see section 2.4) to plan offline flat foot walking gaits using the full model of the robot and rigorously prove the stability of the gait.

2.3 Foot Placement

The placement of the swing leg on the ground has a strong influence on the balance of biped robots. Foot placement strategies use simple control laws, based on a simplified model of the robot, to estimate online where to place the swing foot to achieve the desired behavior. This general philosophy has been utilized in many ways for the control of biped robots.

Marc Raibert was a pioneer in legged locomotion and in developing this approach. In the late 1980s, his 2D and 3D hoppers were able to walk, run and jump in a robust way [94]. Since 2005, Boston Dynamics, a company founded by Marc Raibert, has unveiled impressive robots such as BigDog, PETMAN, ATLAS which are capable to robustly walk in real environments and to perform numerous tasks [95],[107]. ATLAS is probably the most advanced humanoid robot. Unfortunately, the rare publications on these robots do not explain in details how they are controlled.

Foot placement strategies have also been developed on several others biped robots: M2V2 (Jerry Pratt et al. in 2007) [92], Tulip (2008) [11], the DLR biped (2011) [33]). Since 2015, the heavily underactuated robot ATRIAS of the team of Jonathan Hurst has demonstrated robust humanlike gaits outdoor using the natural mass-spring dynamics of the robot [101].

2.3.1 An Intuitive and Decoupled Control

Foot placement strategies decouple the task of walking or running in several decoupled and simple tasks. The control of running for the Marc Raibert's planar hoppers is divided into three decoupled tasks: control of the body height, control of the forward speed by proper foot placement, control of the posture of the robot. The same strategy with little modifications proved successful on his 3D hoppers.

Jerry Pratt et al. used a controller for the swing leg and a controller for the stance leg [92]. Each controller is divided into subcontrollers achieving a given task (swing foot trajectory generation, control of the center of pressure, maintaining robot posture...). Virtual Model Control is used to maintain the posture of the robot. This empirical approach consists in using virtual components such as springs and dampers connected between the robot and the environment, and in generating

the corresponding joint torques [91]. A vertical spring-damper "granny-walker" is especially used to control the center of mass height. Denoting q as the joint angles, the joint torques u are computed using the principle of virtual works

$$u = J(q)^T F\,, \tag{2.8}$$

where F is the force exerted by the virtual components and acting on the point of coordinates p, and $J = \frac{\partial p}{\partial q}(q)$.

The robot ATRIAS was designed to maximally embody a reduced-order model derived from a spring-mass model. For that purpose, the robot has lightweight legs with springs. Rezazadeh et al. used this reduced-order model to design control laws for the robot [101]. They highlighted that the lateral and frontal plane control are of the same nature and approximately decoupled for the reduced-order model. Hence, they proposed to independently control the neutral lengths of the springs in the legs, the swing foot position and the rotational motion (attitude) of the robot. The control laws are derived using the reduced-order model, and are directly used for the control of the actuators of the real robot. The robot has 6 actuators (3 per leg). Two actuators of the swing leg are used to control its horizontal position. One actuator per leg is used to control the spring neutral position. The two remaining actuators are used to control the rotational degrees of freedom of the robot. The swing foot controller is detailed in the next paragraph.

2.3.2 Simple Models to Estimate Swing Foot Placement

Biological studies showed that human, and more generally animal walking and running, is similar to the dynamics of a spring-mass model, i.e. a point mass attached to a massless spring [10]. This inspired the use of simple models to estimate swing foot placement. Raibert introduced the “neutral point”, which is the point on the ground leading to a symmetric walk [94]. It was estimated empirically. Pratt et al. introduced the theory of capturability and used *capture points* to control the robot M2V2 [29]. Capture points are the points on the grounds making the center of mass come to rest over the stance foot (i.e. come to a complete stop) if the swing foot is placed on one of them. Their location is estimated on models derived from the 3D Linear Inverted Pendulum Model and introduced by Kajita [60]. The robot PETMAN of Boston Dyncamics utilized both capture points and Virtual Model Control [107].

The foot placement location of the robot ATRIAS [101] is calculated using a modified PID discrete controller

$$P_{SF} = K_P(\dot{\xi} - v^d) + K_D(\dot{\xi} - \dot{\xi}_{n-1}) + K_I(\xi - v^d t) + K_v\dot{\xi}\,, \tag{2.9}$$

where ξ is the 2D-horizontal-plane location of the center of mass of the robot with respect to a stationary point on the ground, $\dot{\xi}_{n-1}$ is the velocity in the previous step (either at the end, or the average velocity), v^d is the desired horizontal velocity of the center of mass, and P_{SF} is the desired 2D-horizontal-plane location of the swing foot. This control law was validated on a reduced order-model derived from the spring mass model. It works for small velocities. For lager velocities, since the

energy dissipation is higher, the neutral length of the springs is modified to inject energy after midstance. This control law works on the real robot.

2.3.3 Concluding Remarks

The theory of foot placement does not provide a rigorous theory for controlling any biped robots. It is essentially based on intuition and approximative models. Yet, impressive results could be showed off on some platforms. Unfortunately, these works are either poorly documented (the robot ATLAS of Boston Dynamics), or cannot be directly implemented on others robots, as is the case for ATRIAS, which was designed to maximally embody a simple reduced-order model.

2.4 The Method of Virtual Constraints and Hybrid Zero Dynamics

The previously introduced two methods (ZMP walking and foot placement) are popular approaches for the control of biped robots. But, they have some limitations. They use a simplified model of the robot (inverted pendulum) and are rather empirical. Furthermore, flat foot walking does not allow anthropomorphic walking, given that it constrains the robot to keep its feet flat on the ground.

Then, in the early 2000s, a new approach was introduced to generate dynamic motions using the full model of the robot that are provably stable, to explicitly and rigorously take into account the underactuaction of the robot, and ultimately to understand the mathematical principles governing biped locomotion. It started from the observation that passive biped mechanisms, called passive walkers, are able to walk down a slope by naturally converging to a periodic limit cycle [130], and proposed to explore the ways of constraining the nonactuated dynamics of underactuated biped robots so that they converge to a limit cycle. This phenomenon is called *limit cycle walking*.

Two major contributors of this new approach are Jessy W. Grizzle (University of Michigan) and Christine Chevallereau (IRCCyN, Nantes) [49, 127]. The method was first successfully implemented on a planar five-link robot with one degree of underactuation in single support phase, named RABBIT in 2002 [20]. This robot had the particularity of having no feet. It necessarily fell when in single support phase. Using such a robot was a way to get free of the flat foot walking paradigm. Later, in 2009, the compliant planar robot MABEL was able to walk, run and negotiates some obstacles [111]. Since 2014, MARLO, a 3D point feet biped robot with several degrees of underactuation has been able to walk indoor [15]. And recently, using the latest improvement of the method, MARLO is able to walk outdoor on grassfields and parklots [47]. The method also proved to be successful for robots with actuated ankles [125], especially for the humanoid robot DURUS of the team of Aaron Ames (AMBERLab, Georgia Tech) [97].

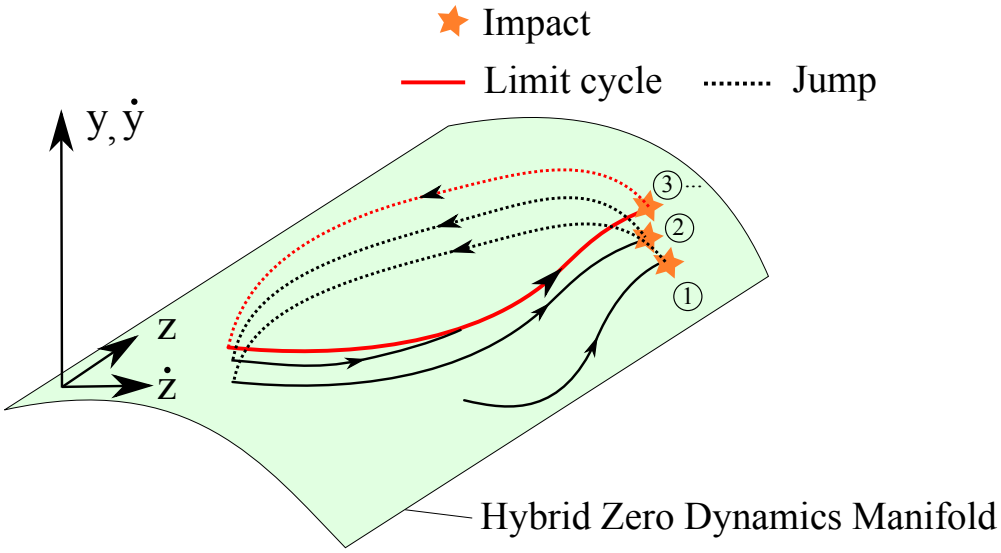


Figure 2.3: Illustration of a limit cycle in the hybrid zero dynamics manifold. The feedback law is used to constrain the system to evolve into the hybrid zero dynamics manifold (depicted in green). A correct choice of the virtual constraints (=outputs) allows the hybrid zero dynamics, i.e. the dynamics restricted to the hybrid zero dynamics manifold, to converge to a limit cycle.

2.4.1 Overview of the HZD Method

For the sake of brevity, we will call the approach "HZD method", as it is sometimes done in the literature. But, we warn the reader that this name is probably too restrictive and misleading.

The method is based on the use of two major tools: virtual constraints and hybrid zero dynamics. Virtual constraints are relations on the state components of a mechanical system imposed through feedback control. They are called *outputs* in the field of control theory.

The hybrid zero dynamics is an extension of the so called zero dynamics to hybrid systems. It defines an exact reduced-dimensional dynamics of the model describing the dynamics of the robot when it is constrained to evolve into the submanifold described by the virtual constraints. This submanifold is called *hybrid zero dynamics manifold*. Then, this approach is generally called *virtual constraints and hybrid zero dynamics method*.

The HZD method consists in finding a set of virtual constraints such that the hybrid zero dynamics asymptotically converges to a periodic orbit with some desired properties. See the illustration in figure 2.3. For that purpose, the virtual constraints, or equivalently the hybrid zero dynamics manifold, are parameterized by a polynomial. The coefficients of the polynomial are selected by solving a nonlinear programming problem, such that the hybrid zero dynamics is asymptotically stable to a periodic orbit, the model hypotheses are respected, and the corresponding walking trajectory has some desired properties. This design is performed offline. Then, a control law is computed online to constrain the robot to evolve into the hybrid zero dynamics manifold.

Hence, the HZD method does not follow the standard approach of motion planning and trajectory stabilization described in the remark at the beginning of this Chapter, and used by the ZMP method and the foot placement method. Instead of designing and stabilizing a reference trajectory, it seeks to design a submanifold of the state space, which is generally multidimensional, in which the dynamics asymptotically converge to a periodic orbit. Then, it is enough to design a feedback law to stabilize the robot into this submanifold to get a periodic walking trajectory. There is no need to design a feedback law that stabilizes the reference walking trajectory, i.e. that stabilizes all the state components of the robot.

2.4.2 Considered Mathematical Model

The HZD method uses the modeling described in Chapter 1. We consider here a biped robot with N degrees of freedom and N_a independently actuated degrees of freedom. The robot is assumed to be subject to p holonomic constraints due to the foot-ground contact. Then, the robot has

$$N_p = N - p \tag{2.10}$$

degrees of freedom. For the sake of brevity, but without loss of generality, we assume that the hybrid model is made of a single phase, as in the example in section 1.5.

We assume that the equations of motion can be written using the reduced coordinates [2] $q \in \mathcal{Q} \subset \mathbb{R}^n$, i.e. in a set of independent coordinates that take into account the p holonomic constraints exerted on the robot (see 1.3.4). We assume moreover that the generalized positions $q = (q_a; q_u)$ have been selected such that q_a are actuated, and q_u are unactuated, that is the right hand side in the Lagrange's equations is zero

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}_u} - \frac{\partial \mathcal{L}}{\partial q_u} = 0\,. \tag{2.11}$$

The continuous dynamic model is

$$D(q)\ddot{q} + H(q, \dot{q}) = B(q)u\,, \tag{2.12}$$

The hybrid model is

$$\begin{cases} \dot{x} = f(x) + g(x)u, & x \notin \mathcal{S} \\ x^+ = \Delta(x), & x \in \mathcal{S} \end{cases} \tag{2.13}$$

where $\mathcal{S}$ is the impact surface, i.e. the surface when impact occurs, Δ is the impact map, $f(x) = -D(q)^{-1}H(q, \dot{q})$, and $g(x) = D(q)^{-1}B$.

[2] It is generally true for the considered class of robots when in single support phase. The hybrid zero dynamics could also be written in the extended coordinates, i.e. with coordinates that are not all independent. But, this makes things more complex and so makes the use of the hybrid zero dynamics less interesting. That's why it is preferable to use a model written in the reduced coordinates.

2.4.3 Virtual Constraints

Physical constraints under the form $h(x) = 0$, such that constraints exerted by the environment or a physical mechanism, constrain the state x of the robot to evolve in a lower dimensional manifold (see 1.3.4).

The evolution of the state can also be "artificially" constrained in a lower dimensional manifold by the use of a feedback law zeroing the output

$$y : x \to h(x)\,. \tag{2.14}$$

In this case, the constraints $h(x) = 0$ are said to be *virtual*, since they are not induced by a physical mechanism. Virtual constraints are relations on the state components of a mechanical system imposed through feedback control. They have the strong interest to be reprogrammable on the fly without any physical modification.

2.4.3.1 Holonomic Virtual Constraints.

Holonomic virtual constraints correspond to outputs depending on the positions only

$$y = h(q)\,. \tag{2.15}$$

Differentiating twice the vector of outputs and using (2.13) gives

$$\ddot{y} = L_f^2 h + (L_g L_f h)u\,, \tag{2.16}$$

where

$$\begin{cases} L_f^2 h = -\dfrac{\partial h}{\partial q}(q)D^{-1}(q)H(q,\dot{q}) + \dfrac{\partial}{\partial q}\Big[\dfrac{\partial h}{\partial q}(q)\dot{q}\Big]\dot{q} \\ L_g L_f h = \dfrac{\partial h}{\partial q}(q)D^{-1}(q)B \end{cases} \tag{2.17}$$

The notation $L_f h$ corresponds to the Lie derivative of h relative to f [3], i.e. $L_f h(x) = \frac{\partial h}{\partial x}(x)f(x)$.

Since the acceleration $\ddot{q}$ appears only in the second derivative of y, via (2.12) it is the same for the inputs. Then, the relative degree of the outputs is at least two.

2.4.3.2 Nonholonomic Virtual Constraints.

Nonholonomic virtual constraints are outputs depending both on the position and on the velocity

$$y = h(q,\dot{q})\,. \tag{2.18}$$

Due to the dependence on the velocity $\dot{q}$, such outputs have generally a relative degree equal to one. But, it is possible to get a relative degree two by introducing the momenta conjugate to q_u [46]

$$\sigma = \frac{\partial \mathcal{L}}{\partial \dot{q}_u}(q, \dot{q}) \,, \tag{2.19}$$

where $\mathcal{L}$ is the Lagrangian of the system.

From (2.11), we have

$$\frac{d}{dt}\sigma = \frac{\partial \mathcal{L}}{\partial q_u}(q, \dot{q}) \,. \tag{2.20}$$

Then, the components of σ have a relative degree at least equal to two. Hence, outputs under the form

$$\begin{aligned} y &= h(q, \sigma) \\ &=: \tilde{h}(q, \dot{q}) \,, \end{aligned} \tag{2.21}$$

have a relative degree at least equal to two. Such nonholonomic virtual constraints were recently introduced in the HZD method and proved to outperform holonomic virtual constraints.

2.4.4 An Exact Reduced-Order Model: The Hybrid Zero Dynamics

Since N_a is the number of actuators of the robot, N_a virtual constraints can be generated by the actuators. Denoting r as the vector relative degree of the corresponding outputs y, the state is constrained to evolve in a $(2N_p - r)$-dimensional manifold, called *hybrid zero dynamics manifold*. This defines an exact reduced-order for the model, called *hybrid zero dynamics*. The higher the vector relative degree of y is, the smaller the dimension of the hybrid zero dynamics is. But, it is unclear whether getting the smallest dimension is the best.

Generally, the HZD method uses outputs with a relative degree equal to two [127, 111, 47]. In this case, the hybrid zero dynamics of dimension $2(N_p - N_a)$. Robust walking cycles could be generated for both planar and tridimensional biped robots.

The choice of the virtual constraints has a strong influence on the hybrid zero dynamics, and especially on the convergence of the hybrid zero dynamics to a limit cycle. To facilitate the selection of virtual constraints leading to a limit cycle behavior, the HZD method proposes to restrict the study to the exact reduced-order model defined by the hybrid zero dynamics. Actually, the *hybrid zero dynamics* is the extension to hybrid systems of one key concept of control theory, called the *zero dynamics* [3].

2.4.4.1 The Zero Dynamics

Virtual constraints define an output vector $y = h(x)$ of dimension N_a. Let r be the sum of the relative degrees of each components of y. Then, denoting $\eta = (y, \dot{y})$,

and assuming that the decoupling matrix $L_g L_f h(x)$ is invertible (see Appendix G), the dynamics can be written

$$\begin{cases} \dot{\eta} = f_\eta(\eta, z) + g_\eta(\eta, z)u \\ \dot{z} = f_{zero}(\eta, z) \end{cases} \tag{2.22}$$

When the virtual constraints are perfectly tracked, i.e. when η is zero, the dynamics $\dot{z} = f_{zero}(0, z)$ are called the *zero dynamics* [3]. The zero dynamics are of dimension $2(N_p - r)$. The *zero dynamics manifold* $\mathcal{Z}$ is defined as the submanifold of the state space where the state is constrained to evolve due to the virtual constraints

$$\mathcal{Z} = \Big\{ x \in \mathcal{TQ} \,|\, \eta(x) = 0_{r\times 1} \Big\} . \tag{2.23}$$

Deriving the Zero Dynamics for Virtual Holonomic Constraints. We consider here the particular case of *virtual holonomic constraints*, that is of outputs depending on the positions only [3]

$$y = h(q) . \tag{2.24}$$

Then, the relative degree of each components is at least two [4]. We assume that the vector relative degree of the output is $(2, 2, ..., 2)$.

The zero dynamics manifold is

$$\mathcal{Z} = \Big\{ x \in \mathcal{TQ} \,|\, y = h(q), \dot{y} = \frac{\partial h}{\partial q}(q)\dot{q} = 0 \Big\} , \tag{2.25}$$

and the dimension of the zero dynamics is of dimension $2(N_p - N_a)$.

Now, let us derived the equations of the zero dynamics as in [49, 127]. In $\mathcal{Z}$, from the implicit function theorem, there exists $N_p - N_a$ independent coordinates $q_f \in \mathcal{Q}_f \subset \mathcal{R}^{N_p - N_a}$ such that (q_c, q_f) is a set of generalized coordinates for the robot. Let $F : \mathcal{Q}_f \to \mathcal{Q}$ be the embedding from $\mathcal{Q}_f$ to $\mathcal{Q}$, that is the mapping that allows to reconstruct q from q_f

$$q = F(q_f) . \tag{2.26}$$

Mutliplying the equations of motion (2.12) by $B^\perp$, a full rank $(N_p - N_a) \times N_a$ matrix, such that $B^\perp B = 0_{(n-N_a)\times 1}$, and using (2.26) gives

$$\begin{aligned} &\Big[B^\perp D(q) \frac{\partial F}{\partial q_f}(q_f) \Big]_{|q=F(q_f)} \ddot{q}_f + B^\perp \Big[D(q) \frac{\partial}{\partial q_f} \Big(\frac{\partial F}{\partial q_f}(q_f)\dot{q}_f \Big) \dot{q}_f + H(q, \dot{q}) \Big]_{|q=F(q_f)} \\ &\quad = 0_{(n-N_a)\times 1} . \end{aligned} \tag{2.27}$$

[3] This can be easily generalized to nonholonomic virtual constraints.

[4] At least two derivations with respect to the time are required to get the acceleration $\ddot{q}$, and so the vector of inputs u in the expression.

We assume that F is chosen such that the matrix $\Big[B^{\perp}D(q)\frac{\partial F}{\partial q_f}(q_f)\Big]_{|q=F(q_f)}$ is invertible. Then, letting $z=(q_f;\dot{q}_f)$, the zero dynamics is

$$\begin{aligned}\dot{z} &= \begin{pmatrix} \dot{q}_f \\ -\Big[B^{\perp}D(q)\frac{\partial F}{\partial q_f}(q_f)\Big]^{-1}_{|q=F(q_f)} B^{\perp}\Big[D(q)\frac{\partial}{\partial q_f}\Big(\frac{\partial F}{\partial q_f}(q_f)\dot{q}_f\Big)\dot{q}_f + H(q,\dot{q})\Big]_{|q=F(q_f)} \end{pmatrix} \\ &= f_{zero}(z)\,. \end{aligned} \tag{2.28}$$

Actually, the process of derivation of the zero dynamics is close to the one used to write the equations of motion of a system subject to mechanical holonomic constraints in a set of reduced coordinates (see Section 1.3.4). Here, mechanical holonomic constraints are replaced by virtual constraints. And the input u plays the role of the generalized forces associated to the mechanical holonomic constraints λ. The input (2.51) allows to maintain the system into the zero dynamics manifold $\mathcal{Z}$.

Remark: Note that the zero dynamics depends on F, and so on the choice of the virtual constraints $y=h(q)$.

2.4.4.2 The Hybrid Zero Dynamics

The zero dynamics is a useful concept in the sense that it defines a lower dimensional dynamics that is more convenient to study. Indeed, the evolution of the complementary dynamics $\eta = 0_{r\times 1}$ is entirely known and controlled by the input u. But, additional care must be taken so that the study of the dynamics of a biped robot may be reduced to the zero dynamics. Indeed, the input (2.51) ensures that the state remains in $\mathcal{Z}$ during the continuous phase, but not during the discrete phase, i.e. when the state jumps when it crosses the impact surface $\mathcal{S}$. If the zero dynamics manifold $\mathcal{Z}$ is not preserved by the impact, it is not possible to restrict the study to it. The property of *impact invariance* is then necessary and defined as follows. The zero dynamics manifold $\mathcal{Z}$ is said to be impact invariant when

$$\Delta\Big(\mathcal{Z}\cap\mathcal{S}\Big)\subset\mathcal{Z}\,, \tag{2.29}$$

where Δ is the impact map. This means that the action of the impact map Δ is to send back a state in the hybrid zero dynamics manifold to the hybrid zero dynamics manifold.

When $\mathcal{Z}$ is both continuous and impact invariant, the nonlinear system with impulse effects can be defined

$$\begin{cases} \dot{z} = f_{zero}(z), & z\notin\mathcal{S}\subset\mathcal{Z} \\ z^{+} = \Delta_{zero}(z), & z\in\mathcal{S}\subset\mathcal{Z} \end{cases} \tag{2.30}$$

where Δ_{zero} is the restriction of the impact map Δ to the hybrid zero dynamics manifold.

(2.30) is called the *hybrid zero dynamics*, and defines an exact reduced-order model. Restricting the study of the robot to the hybrid zero dynamics reduces the complexity of the problem. For a planar five-link walker with point feet, as depicted in figure 2.4, we have $N_a = 4$ and $N_p = 5$. Then, the hybrid zero dynamics is of dimension $2(N_p - N_a) = 2$, while the full dynamics is of dimension $2N_p = 10$.

The asymptotic stability of the limit cycle in the hybrid zero dynamics is equivalent to the stability of the full system. The design of periodic orbits can be realized by restricting the study to the hybrid zero dynamics. Because the hybrid zero dynamics is made invariant by a proper choice of the input, any solution inside the hybrid zero dynamics is a solution of the overall (closed-loop) system. Then, periodic orbits for the hybrid zero dynamics correspond to periodic orbits for the full dynamics [127] (Theorem 5.5 page 134). Hence, the hybrid zero dynamics is an exact reduced-order model that facilitates analytical studies and reduces the computation cost when integrating the equations of motion of the robot. See an illustration of a limit cycle in the hybrid zero dynamics manifold depicted in figure 2.3.

The exponential stability of the hybrid zero dynamics is studied by using the so called method of Poincaré sections. See Appendix A for further details on this method. The Poincaré map is restricted to the hybrid zero dynamics to leverage the computation cost.

Achieving impact invariance. We address now the problem of how to achieve the impact invariance (2.29). For one degree underactuated robots, like RABBIT or VS, this is not a difficulty. It was proved in [127] (Theorem 5.2 page 126) that if there exists at least one point of $\mathcal{Z} \cap \mathcal{S}$ that is impact invariant, then the manifold is impact invariant, i.e. $\Delta\Big(\mathcal{Z} \cap \mathcal{S}\Big) \subset \mathcal{Z}$. This means that if a walking trajectory, i.e. a periodic orbit, exists for a given set of virtual constraints, then the corresponding zero dynamics manifold $\mathcal{Z}$ is impact invariant[5].

This property is not true anymore for several degrees underactuated robots. More precisely, no surface was found to have this property. Thus, [77] introduced a systematic way to locally modify the virtual constraints (or equivalently the outputs) after impact such that η be zero after impact, thus achieving impact invariance. More especially, for an output $y_i = h_i(x)$ of relative degree r, it consists in defining the new output

$$\tilde{y}_i = \tilde{h}_i(y_i^+, \dot{y}_i^+, ..., y_i^{(r-1)+}, x)\,, \tag{2.31}$$

where $(y_i^+, \dot{y}_i^+, ..., y_i^{(r-1)+})$ are the values of the output and its derivatives just after impact. $\tilde{y}_i$ is designed such that its value and its time derivatives just after impact $(\tilde{y}_i^+, \dot{\tilde{y}}_i^+, ..., \tilde{y}_i^{(r-1)+})$ be zero, and $y_i = \tilde{y}_i$ from a point during the step, typically from the middle of the step. This is done online by using polynomial interpolations to smoothly join $\tilde{h}_i$ to h_i. This technique is sometimes called *deadbeat hybrid extension*.

[5]Indeed, due to the existence of the periodic orbit in $\mathcal{Z}$, one point of $\mathcal{Z}$ is impact invariant.

2.4.5 Virtual Constraints Used in the HZD Method

We have just presented the general concepts of virtual constraints and hybrid zero dynamics, which are the two key concepts used in the HZD method. The choice of virtual constraints is large, but only some peculiar classes of virtual constraints are used in the HZD method. Their use is motivated by physical intuition.

2.4.5.1 Introducing a Curvilinear Abscissa: the Phase Variable

Synchronizing the Actuated Dynamics with the Nonactuated Dynamics. Usually, in robotics and more generally in the control of systems, reference trajectories are parameterized by the ordinary time t. Nonetheless, controlling an underactuated system using a time parameterization may be tricky, especially if the motion is computed offline as it is the case for the HZD method.

To be convinced of that, let us consider a one degree underactuated point feet biped robot, like the robot VS or RABBIT and depicted in figure 2.4. A set of generalized positions is $q = (q_a; \theta)$, where $q_a = (q_1; q_2; q_3; q_4)$ is the vector of the actuated coordinates. The angle θ is called the *virtual stance leg* angle, and is not actuated. It describes the orientation of the robot in the sagittal plane. Assuming that the four actuators are used to control the four actuated joints $q_a = (q_1; q_2; q_3; q_4)$, and given that the foot-ground contact is not actuated, the dynamics of $(\theta; \dot{\theta})$ are not directly controlled. They correspond to the zero dynamics and are of dimension two.

The evolution of $(\theta; \dot{\theta})$ depend on the initial conditions and on the perturbations exerted on the robot. Then, the rotating motion of the robot around its stance leg is likely to be different than the one obtained during the offline trajectory generation. So, using time-parameterized trajectories computed offline, and ignoring the information on the evolution of the zero dynamics risk to lead to a too early or too late impact of the swing leg onto the ground, and to the fall of the robot. Actually, it seems better that the actuated degrees of freedom of the robot be synchronized with the nonactuated dynamics.

Phase Variable. Due to the aforementioned reasons, the HZD method uses time-independent reference trajectories. Instead of time, the trajectories are parameterized by a generalized position of the hybrid zero dynamics, called *phase variable*, and playing the role of a curvilinear abscissa. The phase variable must evolve strictly monotonically during one step. Indeed, bijectivity with the time is necessary, otherwise the parameterization would be ambiguous: several points of the reference trajectory would correspond to a single value of the phase variable.

A common choice for the phase variable is the virtual stance leg angle θ (see figure 2.4), which was found to evolve strictly monotonically for humans and containing information on the evolution of the underactuated dynamics in the sagittal plane.

Then, holonomic virtual constraints on the actuated degrees are defined as follows

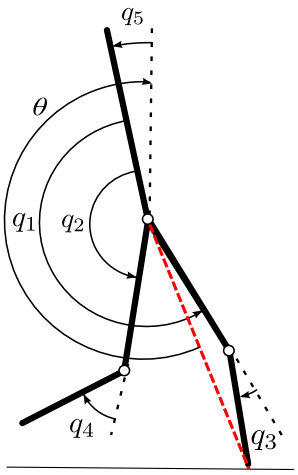


Figure 2.4: In the HZD method, the gait phase variable θ is generally chosen as the virtual stance leg orientation relative to the vertical. The virtual stance leg is the red dotted line.

$$y = q_a - q_{a,ref}(\theta) =: h(q)\,, \tag{2.32}$$

where q are the generalized positions of the robot, q_a are the actuated degrees of freedom [6], and $\theta \to q_{a,ref}(\theta)$ is the vector of the reference trajectory for q_a.

2.4.5.2 Parameterization of the Reference Trajectory for q_a

The virtual constraints (2.32) have been used for a long time in the HZD method. But, they have some limitations and it may be sometimes preferable to use others types of virtual constraints.

Holonomic virtual constraints Considering the holonomic virtual constraints (2.32) is satisfactory for planar biped robots with point feet with one single degree of underactuation, like for RABBIT or the robot VS. In this case, the virtual constraints define a unique path in the configuration space. Indeed, for a given value of the phase variable θ, the holonomic virtual constraints define a unique configuration $q = (q_{a,ref}(\theta), \theta)$.

But, if the degree of underactuation is greater than one, the path described by the virtual constraints is not unique anymore. To be convinced of that, let us consider the example of a point feet 3D biped robot as described in [22], and depicted in figure 2.5. In single support phase, the robot has two degrees of underactuation: the orientation in the sagittal plane θ (pitch) and in the frontal plane ϕ (roll). The orientation in the transverse plane (yaw) is assumed to be locked. A set of generalized positions is $q = (q_a; \theta; \phi)$. The holonomic constraints (2.32) do not define a unique configuration, since they are independent of the roll angle.

[6] The subscript a stands for "actuated".

Then, if ϕ is not as expected, the robot risks impacting prematurely the ground and falling (see figure 2.5). Hence, virtual constraints, and so the phase variable, should also depend on ϕ. A good parameterization must partition the configuration space such that the aforementioned problem does not occur anymore [16]. This is still an open problem.

To compensate for the bad parameterization offered by the virtual stance leg θ, [15] proposed a new choice of virtual constraints that couple θ and ϕ in ways that cannot be found through intuition. That new choice of outputs comes to add a dependance of the virtual constraints to the roll angle ϕ, that is considering virtual constraints under the form

$$y = q_a - q_{a,ref}(\theta, \phi)\,. \tag{2.33}$$

See Section 2.4.6.4 for further details.

Nonholonomic virtual constraints. Recently, the use of the nonholonomic virtual constraints proved to offer a wider parameterization that outperforms the holonomic virtual constraints, especially in term of the size of the basin of attraction of the limit cycle in the hybrid zero dynamics manifold [47]. Especially, the use of σ, the momenta conjugate to q_u (2.19), in the parametrization results in the nonholonomic virtual constraints

$$\begin{aligned} y &= q_a - q_{a,ref}(\theta, \sigma) \\ &=: h(q, \dot{q})\,. \end{aligned} \tag{2.34}$$

Actually, the reference for q_a should ideally depend on all the coordinates of the hybrid zero dynamics manifold $(q_u; \dot{q}_u)$

$$y = q_a - q_{a,ref}(q_u, \dot{q}_u)\,, \tag{2.35}$$

or assuming that $(q_u; \sigma)$ forms a valid set of coordinates for this manifold

$$y = q_a - q_{a,ref}(q_u, \sigma)\,. \tag{2.36}$$

This offers the widest possibilities for the actuated degrees of freedom to adapt to the current dynamic state of the robot. Differently said, it a priori offers more freedom to design an hybrid zero dynamics manifold containing a periodic orbit with a large basin of attraction. Surprisingly, to the best of our knowledge, virtual constraints (2.36) have not been considered in the HZD method.

2.4.6 Offline Design of the Virtual Constraints

Virtual constraints are designed so that the hybrid zero dynamics asymptotically converges to a periodic orbit, and the corresponding walking cycle has some desired properties.

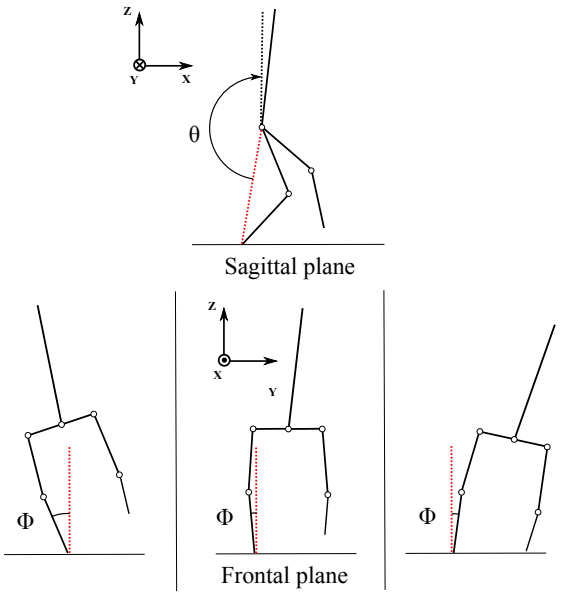


Figure 2.5: Illustration of the importance of taking into account the roll angle ϕ into the virtual constraints. For one given value of the absolute orientation in the sagittal plane θ, the roll angle angle of the robot may take any value. Not considering ϕ may lead to a too early or too late impact, even though the virtual constraints are perfectly tracked.

2.4.6.1 Designing Virtual Constraints by Solving a Parametric Optimization Problem

A set of virtual constraints is designed by solving a nonlinear optimization problem. The virtual constraints are parameterized as polynomial functions. Then, in the general case, virtual constraints (2.14) are searched under the form

$$y = h(x, \beta)\,, \tag{2.37}$$

where β is the vector of the coefficients of the polynomials

More especially, when considering the peculiar class of virtual constraints (2.32) and (2.34), we have

$$h(q) = q_a - h_d(\theta, \beta)\,, \tag{2.38}$$

or

$$h(q) = q_a - h_d(\theta, \sigma, \beta)\,, \tag{2.39}$$

where q_a are the actuated joints, and h_d is the vector of the polynomials [7].

The coefficients of the polynomials β and an initial condition in the hybrid zero dynamics manifold are taken as the parameters $P \in \mathcal{P}$ to optimize. Then,

[7] Bézier polynomials are generally used, since each coefficients has a local influence on the shape of the polynomial.

assuming that the holonomic constraints are of vector relative degree $(2, ..., 2)$, and that each polynomial has k coefficients, then kN_a polynomial coefficients and a vector of $2(N_p - N_a)$ coordinates in the hybrid zero dynamics have to be found. This number of optimization parameters can be reduced by using some of the constraints that we mention next, such the constraint of periodicity.

The desired properties of the walking cycle generated by the virtual constraints can be formulated with a set of equality $C_e(x, u) = 0$, inequality constraints $C_i(x, u) \leq 0$, and using a cost function $J(x, u)$.

Hence, designing walking trajectories comes to solve the nonlinear parametric optimization problem:

$$\begin{aligned} &\min_{P \in \mathcal{P}} J(x, u) \\ &\text{s.t. } C_e(x, u) = 0 \\ &\quad C_i(x, u) \leq 0\,. \end{aligned} \tag{2.40}$$

2.4.6.2 Constraints

The motion must satisfy a minimum set of equality $C_e(x, u) = 0$ and inequality constraints $C_i(x, u) \leq 0$ that we divide into four categories:

Constraints to have an asymptotic stable periodic orbit [8]:

- Periodicity of the trajectory.
- Stability of the trajectory: eigenvalues of the Poincaré map inside the unit circle (see Section 2.4.4.2 and Appendix A).

Constraints due to the model hypotheses (see Sections 1.3.5 and 1.4.2):

- No slippage of the stance foot.
- No take off of the stance foot.
- Swing foot strictly above the ground during a step.
- Impact model hypotheses respected.

Constraints due to the physical limits of the robot:

- Joint limits.
- Actuator bounds (velocity, torque).

User's defined constraints:

[8] From Section 2.4.4.2, it is enough to check these properties on the hybrid zero dynamics only.

- Mean walking speed.
- Step length.
- Maximum torso inclination.
- ...

2.4.6.3 Cost Function

The sum of the square torques is generally chosen as a cost function [127]

$$J(x,u) = \left(\frac{1}{L(x)} \int_0^{T(x)} ||u(t)||^2 \, dt\right), \tag{2.41}$$

where $L(x)$ is the step length, and $T(x)$ the step duration. This function approximates the consumption due to the Joule effect in the motors. Furthermore, minimizing torques leaves the possibility for the robot to use more torques without saturating actuators when subject to disturbances.

To design a motion robust to disturbances, such as an external force, or terrain height disturbances, [45] introduced in the cost function an additional term penalizing the deviations induced by the disturbances

$$J_i(x,u) = \int_0^{T(x)} \frac{\theta - \theta^+}{(\theta^- - \theta^+)^2} \Big(||\delta x(t)||^2 + ||\delta u(t)||^2\Big) \, dt \,, \tag{2.42}$$

where $\delta x(t)$ is the state deviation and $\delta u(t)$ is the input derivation relative to the reference, obtained when integrating the motion when subject to the disturbances. The weighting term $\frac{\theta-\theta^+}{(\theta^- - \theta^+)^2}$ gives more importance to deviations at the end of the step, since it is crucial for the robot to be as close as possible to the reference at the end of a step.

The cost function becomes

$$J(x,u) = \left(\frac{1}{L(x)} \int_0^{T(x)} ||u(t)||^2 \, dt\right) + \sum_{i=1}^{M} J_i(x,u) \,, \tag{2.43}$$

where M is the number of scenarios with disturbances considered. This means that at each iteration of the optimization, the dynamics of the robot is integrated $M+1$ times.

Solving the optimization problem. There exist many methods and software available to solve nonlinear programming problems. But, due to the complexity of the probleme, the design of the virtual constraints online is not feasible. Then, this design is performed offline. In the HZD method, the MATLAB function *fmincon* is generally used and offers good results [127, 16, 116]. More recently, [53] presented a methodology that allows for solving faster the nonlinear parametric optimization problem using a direct collocation framework.

Remark: The equations of motion of the robot must be integrated at each iteration of the optimization problem. As seen above, to alleviate the computation cost, only the hybrid zero dynamics may be integrated. But, it is also possible to work with the full model.

2.4.6.4 Treating the Asymptotic Stability A Posteriori: Selection of the Virtual Constraints

For the case of a one degree underactuated planar robot with point feet, like RABBIT or the robot VS, and considering outputs of relative degree two, the hybrid zero dynamics is of dimension two. Then, the Poincaré map restricted to the hybrid zero dynamics is scalar. Due to the simplicity of the Poincaré map, analytic algebraic criteria for the existence and stability of a periodic orbit in the hybrid zero dynamics can be exhibited, and directly included in the nonlinear programming problem (2.40) [127] (Theorem 5.3 page 129). More specifically, a periodic orbit is stable if and only if the angular momentum of the robot juste after impact is strictly less than the angular momentum just before impact. In practice, this is generally the case and periodic orbits are generally exponentially stable.

But things get more complex for several degree underactuated robots, for which checking the asymptotic stability of the periodic orbit in the hybrid zero dynamics may dramatically slow down the resolution of the nonlinear programming problem (2.40). In this case, the Poincaré map restricted to the hybrid zero dynamics is multidimensional and no analytic algebraic criteria could be found to check for the stability. Then, the numerical computation of the eigenvalues of the jacobian of the Poincaré map must be performed. It is time consuming (see Appendix A).

That's why it may be preferred to remove the property of asymptotic stability from the nonlinear programming problem (2.40). Thus, the resolution of the problem gives a periodic orbit $\mathcal{O}$ described by the system in the state space. But, it is not necessarily asymptotically stable. In a second step, the virtual constraints are modified to design a new hybrid zero dynamics manifold such that the system asymptotically converges to $\mathcal{O}$ when it is constrained to evolve into this manifold.

Indeed, several choices of virtual constraints can be made to control some components of one periodic orbit $\mathcal{O}$. It has been especially shown that, for a given periodic orbit, the hybrid zero dynamics depends on the choice of the virtual constraints [9] when the degree of underactuation is greater or equal than two (see equation (35) in [22]).

How to select the virtual constraints? The selection of the virtual constraints to design an hybrid zero dynamics manifold asymptotically stable to a given periodic orbit has only been done for the holonomic constraints (2.32).

The selection of the virtual constraints can be realized by intuition. Considering a new virtual constraint as the distance between the swing leg end and the center

[9] Of vector relative degree (2,..,2).

of mass along the frontal plane direction rendered periodic orbits asymptotically stable for a 3D biped robot with two degrees of underactuation [22].

In [16] a systematic selection of the outputs by solving a BMI problem is proposed. A family of virtual constraints preserving the orbit $\mathcal{O}$ is considered

$$y = H(\xi)(q - q_d(\theta)) \,, \tag{2.44}$$

where the $N_a \times N_p$ selection matrix $H(\xi)$ is parametrized by the parameters ξ, and $\theta \to q_d(\theta)$ is the reference motion for the generalized positions q.

Let ξ^* be the parameters corresponding to the nominal choice of the virtual constraints (2.32) used in the nonlinear programming problem (2.40). A Taylor expansion of the jacobian of the Poincaré map is

$$\frac{\partial P}{\partial x}(x^*, \xi) \approx A_0 + \sum_{i=1}^{n} A_i(\xi_i - \xi_i^*) \,, \tag{2.45}$$

where x^* is the nominal point of the periodic orbit on the Poincaré surface. $\Delta_{\xi_i} = \xi_i - \xi_i^*$ is then chosen by optimization such that the eigenvalues of $\frac{\partial P}{\partial x}(x^*, \xi)$ are inside the unit circle.

In practice, only the parameters ξ introducing the roll angle ϕ in the virtual constraints are varied. This is motivated by the intuition that if the virtual constraints vary with the roll angle, the robot will be able to compensate appropriately when its roll deviates from the nominal orbit (see Section 2.4.5.2). The method resulted in asymptotic stable walking trajectories which were tested experimentally with success [15].

2.4.7 Stabilization of the System into the Hybrid Zero Dynamics Manifold

The previous Section explained how the HZD method proposes to come up with the problem of finding a walking trajectory with some desired properties by designing a set of virtual constraints. The virtual constraints are designed such that they define an embedded manifold of the state space, called hybrid zero dynamics manifold, in which there exists an asymptotically stable periodic orbit. Due to this property, it is enough for the system to be stabilized into the hybrid zero dynamics manifold. This means that the virtual constraints must be tracked sufficiently accurately. This Section addresses the design of feedback laws achieving this goal.

For the sake of brevity, without loss of generality, we will consider the case of holonomic virtual constraints, that is

$$y = h(q) \,, \tag{2.46}$$

of vector relative degree (2,...,2). This case is generally considered in the HZD method. The results can be easily extended to the others types of virtual constraints.

2.4.7.1 An Input-Output Linearizing Controller

Enforcing the virtual constraints can be performed using the standard technique of input-output linearization [3]. More specifically, derivating twice the output relative to the time gives

$$\ddot{y} = L_f^2 h + (L_g L_f h) u \,, \tag{2.47}$$

where:

$$\begin{cases} L_f^2 h = -\dfrac{\partial h}{\partial q}(q) D^{-1}(q) H(q,\dot{q}) + \dfrac{\partial}{\partial q}\left[\dfrac{\partial h}{\partial q}(q)\dot{q}\right]\dot{q} \\ L_g L_f h = \dfrac{\partial h}{\partial q}(q) D^{-1}(q) B \end{cases} \tag{2.48}$$

The notation $L_f h$ corresponds to the Lie derivative of h relative to f [3].

Assuming that the *decoupling matrix* $L_g L_f h$ is invertible [10], one can define the feedback transformation

$$u(x) = (L_g L_f h(x))^{-1}(-L_f^2 h(x) + v(x)) \,, \tag{2.49}$$

where $v(x)$ is the new input. This creates a linear relationship between the inputs and the outputs

$$\ddot{y} = v \,. \tag{2.50}$$

Thus, a controller under the form (2.49) is called an *input-ouput linearizing controller*. The term

$$u^* = -(L_g L_f h(x^*))^{-1} L_f^2 h(x^*) \,, \tag{2.51}$$

where x^* is the state on the periodic orbit is called the *feedforward term*. It corresponds to the input that must be applied to the system so that the state remains in the zero dynamics manifold $\mathcal{Z}$ (see Section 2.4.4.1).

Choice of v. Several choices of v can be made to exponentially stabilize the outputs $(y, \dot{y})$ to zero. Since nothing guarantees a priori that the periodic orbit is exponentially stable in the hybrid zero dynamics when the outputs are not zero, attention must be paid to the stabilizing property of v. See Section 5.5 in [127] for further details.

[76, 127] proved that when using the PD controller

$$v(y,\dot{y}) = -\frac{1}{\epsilon} K_D \dot{y} - \frac{1}{\epsilon^2} K_P y \,, \tag{2.52}$$

where K_D and K_P are diagonal positive definite matrices, for $\epsilon > 0$ sufficiently small, an exponentially stable periodic orbit of the hybrid zero dynamics is also an exponentially stable periodic orbit of the full-order model. This means that the periodic orbit is exponentially stable if the outputs converge sufficiently fast to zero.

The resulting input-ouput linearizing controller with a PD controller is

[10] Meeting this condition is not difficult in practice.

$$u(x) = (L_g L_f h(x))^{-1}\Big(- L_f^2 h(x) - \frac{1}{\epsilon} K_D \dot{y} - \frac{1}{\epsilon^2} K_P y\Big)\,, \tag{2.53}$$

with $y = h(x)$ and $\dot{y} = L_f h(x)$.

Remark: From now on, for the sake of brevity, we denote $K_D = \frac{1}{\epsilon} K_D$ and $K_P = \frac{1}{\epsilon^2} K_P$.

2.4.7.2 Approximate Controllers Derived From the Input-Output Linearizing Controller

The input-ouput linearizing controller (2.49) is known to be sensitive to dynamics parameters uncertainties [16]. Then, approximated versions of this controller, and especially of (2.53), are used in practice and offer good tracking performances.

A PD + feedforward controller. [16, 111] and references therein proposed the following PD + feedforward controller

$$u(x) = -(L_g L_f h(x^*))^{-1} L_f^2 h(x^*) - T^{-1}(K_D \dot{y} - K_P y)\,, \tag{2.54}$$

where x^* is the state on the periodic orbit when the phase variable is $\theta = \theta(x)$.

The feedforward term $-(L_g L_f h(x^*))^{-1} L_f^2 h(x^*)$ is computed regressing the torques of the simulated model along the periodic trajectory, or regressing the experimental torques obtained when using a high gain PD controller (see the next paragraph). The constant matrix T approximates the decoupling matrix. It is not reported how it is determined in [16], but it is taken equal to the identity matrix in [111].

A High Gain PD controller. In practice, many biped robots are controlled using a high gain Proportionnal Derivative (PD) controller [127, 68, 100, 37, 132]

$$u = -K_P y - K_D \dot{y}\,. \tag{2.55}$$

This type of controller is easy to implement, easy to tune, requires little computer resources and offers good tracking performance for robots with a high transmission ratio. The simplicity of this controller may leave in doubt. We explain here why it gives good tracking results.

We remind that high transmission ratios approximately decouple the dynamics of the robot (see Section 1.6.3). The approximated dynamics of the actuated joint coordinates q_a are

$$I_{eq} \ddot{q}_a = u\,, \tag{2.56}$$

The actuated coordinates q_a are chosen to be directly controlled by the actuators. Let $y = q_a - q_{a,ref}$ be the vector of outputs, where $q_{a,ref}$ is the reference. Using an

input-output linearization, and the approximated dynamics (2.56), the dynamics of the output is

$$\ddot{y} = (I_{eq})^{-1}u - \ddot{q}_a \,. \tag{2.57}$$

where I_{eq} is the transmission inertia reflected to joint. The input-output linearizing controller with a PD feedback is [11]

$$u = -K_P y - K_D \dot{y} + \ddot{q}_a \,. \tag{2.58}$$

The PD controller

$$u = -K_P y - K_D \dot{y} \tag{2.59}$$

ensures a practical convergence of the output to zero, that is for all $\epsilon > 0$, there exists K_P and K_D such that $lim_{t\to\infty}|y(t)| \leq \epsilon$. In other words, when the higher the gains of the PD controller are, the more negligible $\ddot{q}_a$ become, and the smaller ϵ is.

But, when setting the gains too high, the torques tend to reach their bounds and to saturate. Additionally, in practice, phenomena of chattering appear due to noise measurement. Then, a compromise has to be found when setting the gains. See Section 3.4.4 for further details on the experimental performances of a high gain PD controller on the planar biped robot VS.

2.4.7.3 The Peaking Phenomenon

Attention must be paid when stabilizing the output with a high gain feedback controller. Actually, a fast stabilization of the output may cause large transient peaks that may disturb the hybrid zero dynamics. Hence, a fast convergence of the output (= a fast stabilization of the system into the hybrid zero dynamics manifold) may eject the system out of the basin of attraction of the periodic orbit. This is called the *peaking phenomenon*. To the best of our knowledge, this phenomenon has not been investigated in the HZD method. For instance, in the Theorem 5.5 in [127], the variation of the size of the basin of attraction when the feedback gains are changed is not considered. For further details on the peaking phenomenon, see Section 4.5 in [108].

To cope with this phenomenon, a solution is to stabilize all the state components of the periodic orbit, as it is done in Section 5.7.1 of this manuscript.

2.4.8 Latest Improvements and Perspectives of the HZD Method

One of the main drawback of the HZD method is to require the design of trajectories offline. But, recent works go into the direction of online trajectory design,

[11] For the sake of brevity, K_P and K_D include the constant term $(I_{eq})^{-1}$.

using walking gait primitivess to design online trajectories respecting a set of constraints [26, 82]. The use of supervised learning for deciding a transition between precomputed gaits seems to be promising [28].

Additionally, [96] exhibited a class of holonomic constraints, called symmetric virtual constraints, which when enforced by controllers ensure the existence of periodic orbits. This interesting property is a step towards the robust and versatile control of biped robots without requiring offline and online search for periodic orbits.

2.4.9 Summary

The HZD method proposes to design virtual constraints defining a submanifold in the state space which is both continuous and impact invariant, and called hybrid zero dynamics manifold. The virtual constraints are selected by solving a nonlinear programming problem such that the hybrid zero dynamics manifold asymptotically converges to a periodic orbit and the corresponding walking trajectories has some desired properties.

There is no unique choice for the class of virtual constraints used to control the robot. Actually, this choice may depend on the platform, and especially on the degrees of underactuation. As far as we understood, using the nonhlonomic virtual constraints (2.21) and the cost function (2.43) offers the best performances.

In this dissertation, we will consider only planar biped robots with one degree of underactuation. We will use holonomic virtual constraints since they proved to give satisfactory results for this class of systems.

2.5 Literature on the Control of Lower-Limb Exoskeletons for Rehabilitation

Active lower limb exoskeletons are anthropomorphic robotic devices that fit around the legs of its user. They are especially designed for performance augmentation (assistance for soldiers or workers in the industry) and for the rehabilitation of persons with physical impairments. A complete review of the state-of-the-art on lower limb exoskeletons can be found in [131, 30]. The control design of an exoskeleton strongly depends on its application and on the physical abilities of its user. We will focus here only on the control strategies that are currently used on lower limb exoskeletons like the exoskeleton of Wandercraft, i.e. aiming at recovering the ability to walk for the people who lost the control of their lower limbs such as paraplegics or myopathics.

The control of such robots is still in its fancy and little progress has been made since the rehabilitation exoskeleton of Vukobratovic et al. (beginning of the 1970s) [121]. Nonetheless, rehabilitation exoskeletons have been available on the market since 2012. The exoskeleton REX is the only exoskeleton able to stabilize impaired patients [9]. But, it is bulk and offers a static gait with a reported mean walking speed of 3 meters per minute (0.05 km/h)! The exoskeleton EKSO and ReWalk

provide a faster gait ($\approx$ 3 km/h), but they are not self stabilized. Users have to use crutches to partially support their weight and to preserve balance [8, 98].

Many researches are currently done to improve those performances [36, 81, 65, 109, 66]. To date, like EKSO and ReWalk, most of exoskeletons are only actuated in the sagittal plane and simply replay trajectories recorded on unimpaired subjects. A finite state machine is used to switch between the different walking phases. The decision of when to make the transition between states is critical. Actually, it seems preferable that the user may decide when to make the next step. Then, a remote control is used for REX and ReWalk, and foot sensors are used to detect the position of the center of pressure, allowing the user to control the exoskeleton by transferring its weight with torso movements [114]. [70] uses the hip angle measurement to estimate when the swing foot touches the ground without foot sensor.

Some more elaborated strategies were tested. The exoskeleton XoR realizes a real-time balance recovery strategy in the sagittal plane using a Spring-Loaded Flywheel model [31]. The Vanderbilt exoskeleton uses electric stimulations of some muscles so that the joints are actuated by both motors and muscles [51]. The MINDWALKER exoskeleton uses a step width adjustment algorithm based on the extrapolated center of mass to modify online the walking trajectory [124]. But, the controller is not efficient enough to self stabilize impaired people.

Then, almost everything remains to be done in that field. And even if the cutting edge techniques used for the control of biped robots are probably a good starting point, the control of exoskeletons rises new challenges. Among them are:

- Designing trajectories that are "human friendly", i.e. ensuring the comfort of the user.
- Properly dividing the control between the human and the robot to allow online interaction and a robust walk.

2.6 Discussion

Today, the community of biped robot locomotion is very active and major improvements have been made over the past few years. In spite of all those intensive years of researches, there is still work to reach the ultimate goal of designing biped robots as agile as humans and consuming little energy.

We found that the HZD method is the most documented and rigorous approach to design controllers for underactuated biped robots. Hence, to get practical experience, we decided to implement this method on a real planar biped robot. This is the object of the next chapter.

Chapter 3

Design and Implementation of a HZD Controller For a Planar Biped Robot

Résumé français Dans ce chapitre, nous présentons la synthèse et l'implémentation de trajectoires de marche pour le robot 5 liens plan de Wandercraft, appelé VS, en utilisant la méthode des contraintes virtuelles et de la dynamique des zéros hybride (méthode HZD). Ce chapitre se structure de la manière suivante. Dans la section 3.2, nous donnons une description du robot VS et expliquons comment nous l'avons modélisé. Puis, dans la section 3.3, nous présentons l'algorithme utilisé à partir de la méthode HZD, et les trajectoires de marche obtenues en simulation. Dans la section 3.4, nous expliquons comment nous avons implémenté ce contrôleur expérimentalement. Nous comparons ensuite les résultats de simulation et expérimentaux. La robustesse des trajectoires à des perturbations modérées est illustrée expérimentalement.

Ce chapitre apporte peu de contributions, puisque la méthode HZD a déjà été implémentée avec succès sur plusieurs robots plans (RABBIT, ERNIE, MABEL, NAO, AMBER-1, AMBER-2, DURUS-2D) et tridimensionnels (MARLO et DURUS). Voir [58], [57] et [5] pour des vidéos et des publications. Le robot VS est semble-t-il le dixième robot à marcher en utilisant cette méthode. Mais il semblerait qu'il soit le premier robot à marcher avec un mécanisme de stabilisation latérale dont la configuration n'est pas invrariante par translation. De plus, nous reportons, probablement pour la première fois, la possibilité pour un robot de marcher, même en présence de perturbations modérées, sans l'usage de capteur pour détecter le contact du pied avec le sol.

3.1 Introduction

In this chapter, we will present the design and the implementation of walking trajectories for the planar five-link biped robot with point feet of Wandercraft, called VS, using the method of virtual constraints and hybrid zero dynamics (HZD method).

The outline of this chapter is as follows. In section 3.2, we will introduce the robot VS and its modeling. In section 3.3 we will describe the path planning algorithm based on the HZD method and walking trajectories obtained using this algorithm. Next, in section 3.4 we will explain how we implemented the controller in practice. We will make comparisons between the simulation and the experimental results. We will try to explain as much as possible the differences observed. The experimental robustness of the walk to moderate disturbances will then be illustrated on various scenarios.

This work brings few contributions, since the method has already been implemented with success on several planar (RABBIT, ERNIE, MABEL, NAO, AMBER-1, AMBER-2, DURUS-2D) and tridimensional robots (MARLO and DURUS). See [58], [57] and [5] for videos and publications. VS is yet an other robot walking with the HZD method. More specifically, it must be the tenth robot. But, to the best of our knowledge, this is the first robot walking with a lateral stabilizing mechanism whose configuration is not translational invariant. Furthermore, we also report, probably for the first time, the possibility for a biped robot to walk, even in the presence of moderate perturbations, without using sensors to detect the contact of the feet with the ground.

3.2 The VS Robot

3.2.1 Hardware description

3.2.1.1 Experimental setup

VS robot (see figure 3.1) is a planar biped robot with point feet designed by Thibault Gayral [1] to test control algorithms for biped robots. It is an almost replica of RABBIT, the first robot to walk with the HZD method [21]. The robot has four independently actuated joints (hip and knee joints). Since the robot has no feet, the foot-ground contact is unactuated. Then, in single support phase, the robot has one degree of underactuation. The motion of the robot is planar, constrained to the sagittal plane, i.e. the robot cannot move laterally.

For reason of space, a boom system constraining the robot to follow a circular path as used for most of the planar biped robots [127, 132, 111] could not be used. Instead, a 2 bars mechanism is used and the robot walks on a treadmill. The stabilizing system only provides lateral stabilization. It does not prevent the robot from falling down, forward or backward. This mechanism has the drawback of rendering the system not translational invariant, which complicated the control design. See 3.2.2.5 for a discussion.

The robot is surrounded by a base frame on which the lateral stabilizing mechanism is fixed. The base frame is also used to prevent the robot from going out the treadmill, touching the control interface of the treadmill, and defines an area where nobody should go during the experiments.

[1] Head of the Wandercraft "robotics mechanical design" team.

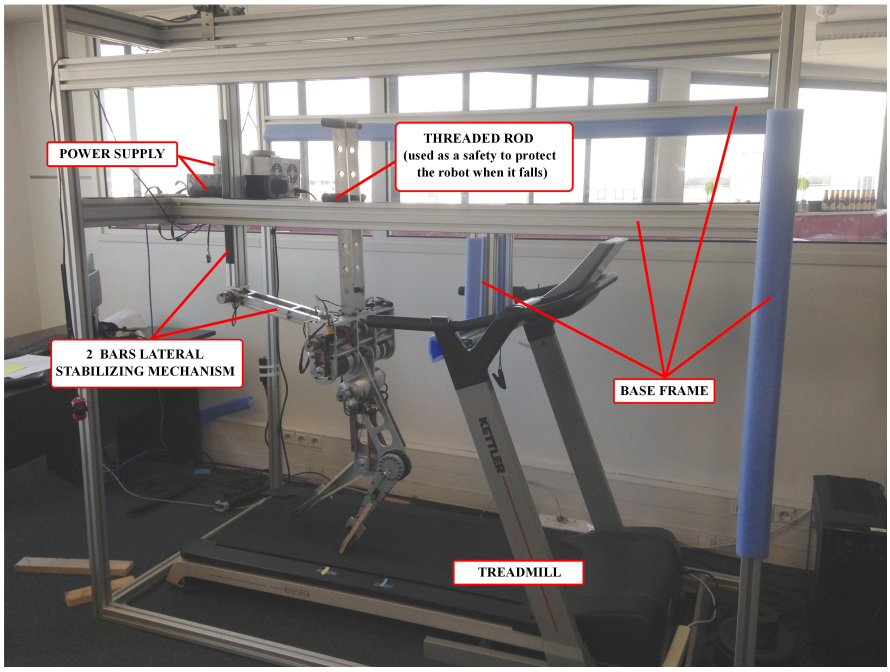


Figure 3.1: Photo of VS's experimental setup. A 2 bars mechanism constrains the robot to evolve in the sagittal plane. For safety, a threaded rod fixed on the torso touches the base frame when the robot falls. During walking, the rod is not in contact with the base frame. No mechanism stabilizes the robot in the sagittal plane.

The kinematic and dynamic parameters of the robot can be found in Appendix B.

3.2.1.2 Sensors and actuators

The four joints are actuated by brushless motors MOOG BN34-35EU-02LH. The mechanical transmission system, depicted figure 1.7, whose goal is to transmit the power from the motor to the joint, is made of a pulley belt transmission and a high gear ratio reducer. Harmonic Drive gearing HFUS 25-100-2SO are used for the knees (reduction factor equal to 101) and PLE 80-100 for the hips (reduction factor equal to 100). The transmission can deliver a joint torque[2] up to 120 Nm and a joint velocity up to 300°/s.

17 bits multi-turn absolute encoders Kübler Sendix absolu F3683 are mounted on the motor shaft. They are used to measure the position and to estimate the velocity of the motor. There is no encoder mounted on the joints. Then, joint position and velocity are deduced from the motor position and velocity assuming no mechanical backlash in the mechanical transmission system. Encoders are mounted on the joints of the 2 bars mechanism to measure the absolute orientation of the torso of the robot. Initially, Force Resistive Sensors (FSR) Tekscan flexiforce A301 were mounted on each foot to detect impacts with the ground. We ended up using them since they deteriorated fast during the experiments. Then, as explained later, we used no sensor to detect the contact of the feet with the ground.

Motor controllers are Elmo Gold Solo Twitter digital servo drives. They servo the motor current to the desired value sent by the control algorithm. They are also used to estimate the motor velocity from encoder measurements.

The computer, that is not embedded but that could, is an Axiomtek PICO 880. For real-time control, the real-time operating system QNX is used [93]. The communication protocol used to transfer data between the servo drives and the computer is EtherCAT [35]. The embedded software, developped by the Wandercraft "embedded sofware team", is coded in C++.

3.2.2 Modeling

We adopt the modeling described in chapter 1. We will consider walking trajectories with instantaneous double support phases, as it is commonly done in the HZD method for robots with point feet [127, 50]. Then, we will only focus on the dynamics of the robot in single support phase. This means that we will consider two continuous phases, and the two corresponding transient (impact) dynamics. The modeling corresponds to the diagram figure 1.6.

3.2.2.1 Notations and Generalized Coordinates

As in chapter 1, to differentiate between the two legs, one leg is called "leg 1" and the other "leg 2". When the robot is in single support and the leg 1 (resp. 2) is

[2]By taking into account the efficiency of the reducers indicated by the manufacturer.

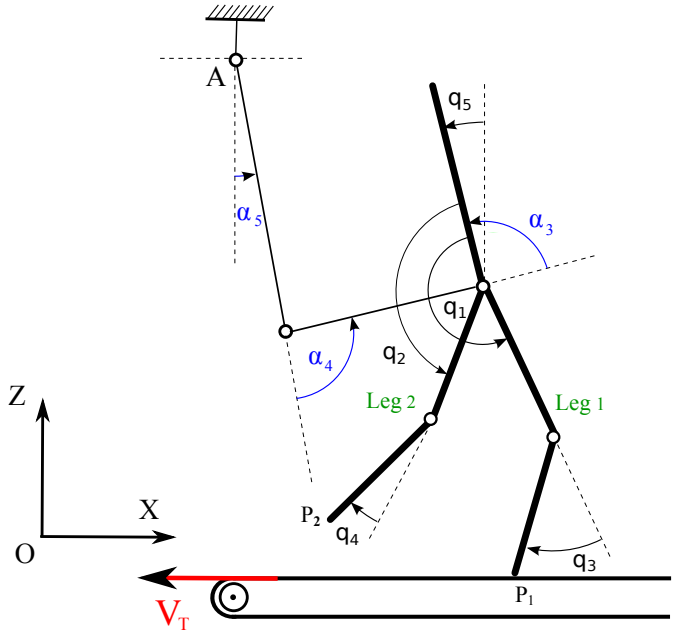


Figure 3.2: Kinematic model of the robot VS. Angles are measured in the trigonometric sense, i.e. positive is counterclockwise.

the stance leg, the phase is called "SS1" (resp. "SS2"). A kinematic model of VS in the phase SS1 is depicted figure 3.2.

The hip and knee joint angles of the leg 1 (resp. leg 2) are parameterized by q_1 and q_3 (resp q_2 and q_4). The torso orientation relative to the vertical is q_5. The treadmill velocity, assumed to be constant, is V_T. The angles of the bars are parameterized by α_4 and α_5. The point A, that corresponds to the point of attachment of the upper bar to the world, is fixed in the world frame (0,X,Z). The point P_1 (resp. P_2) corresponds to the extremity of the leg 1 (resp. 2). Its cartesian coordinates expressed in the world frame are $p_1 = (p_1^x; p_1^z)$. A set of generalized positions for the system is

$$q_e = (q_1; q_2; q_3; q_4; q_5; p_1^x; p_1^z; \alpha_4; \alpha_5) \in \mathcal{Q}_e \subset \mathbb{R}^9 \,. \tag{3.1}$$

The state is $x_e = (q_e; \dot{q}_e) \in \mathcal{TQ}_e$. The dynamical parameters can be found in figure B.1.

3.2.2.2 Dynamics in Single Support Phase

In single support phase, only one leg of the robot is in contact with the ground. Let i be the subscript of the stance leg ($i = 1$ or 2). The extremity of the stance leg is assumed to be pinned on the ground. This means that the foot-ground contact is modeled as an ideal pivot joint. The foot cannot translate but can only rotate without friction relative to the treadmill. All these hypotheses are commonly used in the HZD method [127]. The system is subject to four holonomic constraints, due to the foot-ground contact and due to the non mobility of the point A

$$\eta_i(q_e,t) = \begin{pmatrix} p_i^x \\ p_i^z \\ p_A^x \\ p_A^z \end{pmatrix} - \begin{pmatrix} p_i^{x,0} - V_T t \\ 0 \\ p_A^{x*} \\ p_A^{z*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{3.2}$$

where p_A^{x*}, p_A^{z*} are real constant, and represent the cartesian coordinates of A in the world frame (0,X,Z). $p_i^{x,0}$ is the horizontal position of the extremity of the stance leg i in the world frame at $t = 0$, the instant when the leg i hit the ground.

Note that since P_1 is moving with the treadmill, one holonomic constraint depends on the time.

The equations of motion are derived using the method of Lagrange (see 1.3.3)

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e,\dot{q}_e)\dot{q}_e + G(q_e) = B_e(q_e)u + \left(\dfrac{\partial \eta_i}{\partial q_e}(q_e,t)\right)^T \lambda \\ \eta_i(q_e,t) = 0 \end{cases} \tag{3.3}$$

with $u \in \mathbb{R}^4$ is the vector of inputs (joint torques) and λ are the efforts related to the holonomic constraints. From the principle of virtual works (or d'Alembert's principle), the time dependance of the holonomic constraints does not modify the expression of the generalized force $\left(\frac{\partial \eta_i}{\partial q_e}(q_e,t)\right)^T \lambda$ (see [39] page 52).

The calculation of λ can be performed differentiating twice the holonomic constraints relative to the time. Since the treadmill velocity is assumed to be constant, then $\frac{\partial \eta_i}{\partial t}(q_e,t)$ is a constant, and the expression of λ is the same as (1.31).

The model is valid if and only if the swing leg j is above the ground, the stance leg i does not take off and slip. We assume that the friction coefficient is equal to 0.6. Then, the domain of admissibility of the model is

$$D_{\eta_i} = \left\{ \begin{pmatrix} q_e \\ \dot{q}_e \\ u \end{pmatrix} \in \mathcal{TQ}_e^{\eta_i} \times \mathbb{R}^5 \,|\, p_j^z(q_e) > 0,\ F_i^z(q_e,\dot{q}_e,u) > 0,\ \left|\frac{F_i^x}{F_i^z}(q_e,\dot{q}_e,u)\right| < 0.6 \right\}, \tag{3.4}$$

where F_i^x (resp. F_i^z) is the horizontal (resp. vertical) component of the ground reaction force acting on the stance leg i.

If the stance leg slips or takes off, the model is not valid anymore, and the simulation results are not considered. After the swing leg j hits the ground, that is

$$x_e = (q_e;\dot{q}_e) \in \mathcal{S}_{\eta_i \to \eta_j} = \left\{ x_e = (q_e;\dot{q}_e) \,|\, p_j^z(q_e) = 0,\ \frac{\partial p_j^z}{\partial q_e}(q_e)\dot{q}_e < 0 \right\}, \tag{3.5}$$

the model with the leg j is the stance leg is considered (SSj). This model is initialized using the impact map $\Delta_{i\to j}$.

3.2.2.3 Impact Map

When the swing leg of the robot hits the ground, an impact occurs. Without loss of generality, we assume that before impact the robot is in the phase SSi (subject to the holonomic constraints $\eta_i(q_e) = 0$), and after impact in the phase SSj (subject to the holonomic constraints $\eta_j(q_e) = 0$). The impact map $\Delta_{i\to j}$ maps a point of $\mathcal{S}_{\eta_i\to\eta_j}$ to $\mathcal{TQ}_e^{\eta_j}$.

As we saw, the impact dynamics is assumed instantaneous and is modeled as a jump on the generalized velocities $\dot{q}_e$, whereas q_e is unchanged. But, contrary to the common case, VS is moving on a treadmill. Consequently, the expression (1.64) is not valid anymore. Indeed, due to the treadmill, VS is subject to a time-dependent honolomic constraint (rheonomic constraint). Then, an additional term appears in the expression of the time derivative of the holonomic constraints after impact $\eta_j(q_e, t) = 0$

$$\frac{d}{dt}\Big(\eta_j(q_e,t)\Big) = \frac{\partial\eta_j}{\partial q_e}(q_e,t)\dot{q}_e + \frac{\partial\eta_j}{\partial t}(q_e,t) = 0\,. \tag{3.6}$$

From (3.2) and given that the treadmill velocity is constant, then $\frac{\partial\eta_j}{\partial t}(q_e,t) = \frac{\partial\eta_j}{\partial t}(q_e)$ (no time dependance of the jacobian).

Using the same approach as in 1.4, the impact map becomes

$$\begin{cases} q_e = q_e^- = q_e^+ \\ \dot{q}_e^+ = \Delta_{\dot{q}_e}(q_e,\dot{q}_e^-) = \dot{q}_e{}^- - D_e^{-1}(q_e)\Big(\dfrac{\partial\eta_j}{\partial q_e}(q_e)\Big)^T\left(\dfrac{\partial\eta_j}{\partial q_e}(q_e)D_e^{-1}(q_e)\Big(\dfrac{\partial\eta_j}{\partial q_e}(q_e)\Big)^T\right)^{-1} \\ \qquad\times\left[\dfrac{\partial\eta_j}{\partial t}(q_e) + \dfrac{\partial\eta_j}{\partial q_e}(q_e)\dot{q}_e{}^-\right] \\ I = -\left[\left(\Big(\dfrac{\partial\eta_j}{\partial q_e}(q_e)\Big)D_e^{-1}(q_e)\Big(\dfrac{\partial\eta_j}{\partial q_e}(q_e)\Big)^T\right)^{-1}\left[\dfrac{\partial\eta_j}{\partial t}(q_e) + \dfrac{\partial\eta_j}{\partial q_e}(q_e)\dot{q}_e^-\right]\right. \end{cases} \tag{3.7}$$

where $x_e^+ = (q_e^+;\dot{q}_e^+)$ ($x_e^- = (q_e^-;\dot{q}_e^-)$) is the state after impact (resp. before), and I is the ground reaction impulse.

Due to the time-dependent holonomic constraints, equations are affine but not linear anymore in the velocity before impact $\dot{q}_e^-$, as it was in (1.64). Not surprisingly, when setting $\frac{\partial\eta_j}{\partial t}(q_e^+) = 0$ (no dependance in time), one gets (1.64).

The impact map is valid if and only if the former stance leg, i.e. the leg i, lifts off the ground without interaction, and the leg that impacts the ground (leg j) does not take off and slip "during" the impact

$$\begin{cases} \dfrac{\partial p_i^z}{\partial q_e}(q_e)\dot{q}_e^+ > 0 \\ I_x > 0 \\ \left|\dfrac{I_x}{I_z}\right| < 0.6 \end{cases} \tag{3.8}$$

I_x (resp. I_z) is the horizontal (resp. vertical) component of the impulse. If one of these hypotheses is not verified, the model is not valid and the simulation is stopped.

3.2.2.4 Taking Into Account the Transmission Inertia

Since the transmission ratio is high (~ 100), the inertia of the transmission mechanical system is taken into account as explained in 1.6.

The reflected transmission inertias to the joint are added to the diagonal terms of the mass matrix. This comes to add the following term to the mass matrix $D_e(q_e)$

$$\left(\begin{array}{c|c} I_{eq} & 0_{5\times 4} \\ \hline 0_{4\times 5} & 0_{4\times 4} \end{array}\right), \tag{3.9}$$

with

$$I_{eq} = \begin{pmatrix} I_{eq,hip} & 0 & 0 & 0 & 0 \\ 0 & I_{eq,hip} & 0 & 0 & 0 \\ 0 & 0 & I_{eq,knee} & 0 & 0 \\ 0 & 0 & 0 & I_{eq,knee} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{3.10}$$

where $I_{eq,hip}$ (resp. $I_{eq,knee}$) is the reflected inertia transmission to the hip joint (resp. knee joint). From now, we will denote $D_e(q_e)$ as the mass matrix considering the transmission inertia.

3.2.2.5 Additional Remarks

Influence of the 2 bars mechanism. The configuration of the 2 bars mechanism depends on the absolute position of the stance foot of the robot on the treadmill. Then, the problem is not translationnal invariant. More specifically, we noticed that the 2 bars mechanism tends to pull back the robot when the robot is located in front of the treadmill (on the right in figure 3.2). This is probably due to the moment exerted by the gravity on the upper bar that increases when the robot moves forward. By comparing a model with and without the mechanism in simulation, we noted that the influence of the mechanism cannot be neglected .

Possibility of using a reduced set of coordinates. Due to the holonomic constraints, the generalized positions q_e are not independent each other. The reduced set of independent coordinates $(q; \dot{q})$, with $q = (q_1; q_2; q_3; q_4; q_5) \in \mathcal{Q}_e^{\eta_i}$, may be used. But, the 2 bars mechanism complicates the task of finding the embedding from $\mathcal{Q}_e^{\eta_i}$ to $\mathcal{Q}_e$. We remind that the embedding maps the reduced set of coordinates to the full set of coordinates (see 1.3.4). Actually, after some calculations, the expression of α_4 and α_5 as a function of q can be found. But, several cases must be considered, and even though it is possible, we found that implementing the model in the reduced coordinates $(q; \dot{q})$ is not easy. That's why we finally preferred working with the full coordinates $(q_e; \dot{q}_e)$. The drawback when using the full coordinates is that one cannot reduce the study to the hybrid zero dynamics, and use some of its properties for the design of walking trajectories. Hopefully, restricting the study to the hybrid zero dynamics is not compulsory in the HZD method. In practice, it is mainly a way to speed up calculations. We did not encounter difficulties to find walking trajectories with the full model.

3.3 Designing Walking Trajectories For VS

We present here how we designed walking trajectories for the planar five-link walker VS using the HZD method. See 2.4 for a presentation of this method. We remind that the method consists in finding virtual constraints to be imposed by the control and such that the constrained dynamics (the hybrid zero dynamics) admits an asymptotically stable periodic orbit associated with a satisfactory periodic walking. The design of the virtual constraints is achieved by solving offline a nonlinear parametric optimization problem under constraints. Since the robot has only one degree of underactuation, there is no need to realize a selection of the virtual constraints a posteriori to asymptotically stabilize the obtained periodic orbit, as described in 2.4.6.4.

Walking hypotheses As explained before, the walking gait is assumed to be composed of a single support phase, and an instantaneous double support phase (during the impact). The gait is assumed to be symmetric, i.e. the left and the right leg simply swap their respective motion when a change of stance leg occurs. Then, the planning is realized for one step. The motion of the second step is deduced by symmetry.

Gait phasing variable The gait phasing variable, which plays the role of the curvilinear abscissa of the cycle, is chosen as the virtual stance leg angle relative to the vertical. See figure 3.3. This is a common choice [127, 50]. Since the shin and thigh have the same length, the gait phasing variable can be expressed as a linear combination of the generalized positions. When leg 1 is the stance leg, the gait phasing variable is

$$\theta = -q_1 - \frac{q_3}{2} - q_5 \,, \tag{3.11}$$

and when leg 2 is the stance leg

$$\theta = -q_2 - \frac{q_4}{2} - q_5 \,. \tag{3.12}$$

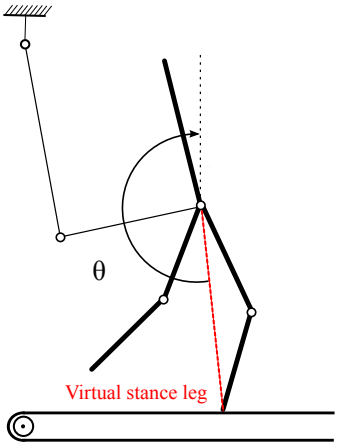


Figure 3.3: The gait phasing variable θ is chosen as the virtual stance leg orientation relative to the vertical. The virtual stance leg is the red dotted line.

Motion planning algorithm We used the virtual holonomic constraints (2.32) as it is generally done in the HZD method. They were designed using the motion planning algorithm described in Section 2.4.6.1.

Since the robot has four independent actuators, we define four virtual constraints for the actuated coordinates

$$y_i = q_i - h_{i,d}(\theta) \,, i \in \{1, ..., 4\} \,. \tag{3.13}$$

The functions $\theta \to h_{i,d}(\theta)$ are parameterized by a 6 degree Bézier polynomial [3]

$$\begin{aligned} h_{i,d}(s) = a_{0i}(1-s)^6 + 6a_{1i}s(1-s)^5 + 15a_{2i}s^2(1-s)^4 + 20a_{3i}s^3(1-s)^3 \\ + 15a_{4i}s^4(1-s)^2 + 6a_{5i}s^5(1-s) + a_{6i}s^6 \,, \end{aligned} \tag{3.14}$$

where

$$s = \frac{\theta - \theta^+}{\theta^- - \theta^+} \,, \tag{3.15}$$

is the normalized gait phasing variable. θ^+ (resp. θ^-) is the value of θ at the beginning (= right after impact) and at the end (= right before impact) of a step. Bézier

[3] The degree of the polynomial was chosen sufficiently "high" to have a large space of search. A degree of 6 proved to be sufficient.

polynomials are often used in the HZD method, since they allow local modifications of the curve when changing one coefficient. This is convenient for hand tuning.

The optimization parameters are the $4 \times 7 = 28$ Bézier coefficients, the treadmill velocity V_T, and the phase variable velocity $\dot{\theta}^-$ at the end of a step. For the optimization, the stance foot absolute position at the beginning of a step is kept constant at a value corresponding to the middle of the treadmill. But, it could have also been included in the optimization parameters. That corresponds to a set of 30 parameters. This set is reduced to $30 - 8 = 22$ parameters (removal of a_{0i} and a_{1i}) using the constraint of periodicity on the state. More specifically, the constraint of periodicity in position, and the assumption on symmetry give

$$\begin{cases} q_2^+ = q_1^- \Leftrightarrow a_{0,2} = a_{6,1} \,, \\ q_1^+ = q_2^- \Leftrightarrow a_{0,1} = a_{6,2} \,, \\ q_4^+ = q_3^- \Leftrightarrow a_{0,4} = a_{6,3} \,, \\ q_3^+ = q_4^- \Leftrightarrow a_{0,3} = a_{6,4} \,. \end{cases} \tag{3.16}$$

After completing a step, the velocities after impact are computed using the rigid impact model (3.7) : $\dot{q}_e^+ = \Delta_{\dot{q}_e}(q_e^-, \dot{q}_e^-)$. Periodicity in velocity and the assumption on symmetry impose that the image of a swing joint (knee or hip) velocity at the end of a step by the impact map be equal to the stance joint velocity at the beginning of a step, and vice versa. These four constraints give [4]

$$\begin{cases} a_{1,1} = a_{0,1} + \dfrac{(\theta^- - \theta^+)}{6\dot{\theta}^+} \dot{q}_2^+ \,, \\ a_{1,2} = a_{0,2} + \dfrac{(\theta^- - \theta^+)}{6\dot{\theta}^+} \dot{q}_1^+ \,, \\ a_{1,3} = a_{0,3} + \dfrac{(\theta^- - \theta^+)}{6\dot{\theta}^+} \dot{q}_4^+ \,, \\ a_{1,4} = a_{0,4} + \dfrac{(\theta^- - \theta^+)}{6\dot{\theta}^+} \dot{q}_3^+ \,. \end{cases} \tag{3.17}$$

Note that these relations only guarantee the periodicity for the actuated states, but not for the unactuated state components q_5 and $\dot{q}_5$ (or equivalently θ and $\dot{\theta}$).

The optimization parameters are found by solving the nonlinear parametric optimization problem:

$$\boxed{\begin{array}{c} \text{Given a vector of unknown } \mathcal{P} = (a_{2,1}, ..., a_{6,4}, V_T, \dot{\theta}^-) \in \mathbb{R}^{22} \,, \\ \text{find } \min_{\mathcal{P}} J(x,u) \,, \text{subject to:} \\ C_e(x) = 0 \\ C_i(x) \leq 0 \end{array}} \tag{3.18}$$

[4] Knowing a_{5i}, a_{6i} and $\dot{\theta}^-$ gives access to q^-, $\dot{q}^-$, and so to $\dot{q}^+$ by using the impact map.

The cost function is the sum of the square torques divided by the step length over a step

$$J(x,u) = \frac{1}{L(x)} \int_0^{T(x)} ||u(t)||^2 \, dt \,. \tag{3.19}$$

We enumerate now the constraints $C_e(x) = 0$, and $C_i(x) \leq 0$ used for the parametric optimization and some of their typical corresponding thresholds. Note that depending on the kind of desired trajectory, some threshold values were changed.

Constraints to have a periodic orbit:

- Periodicity of the trajectory : θ and $\dot{\theta}$ periodic.

Constraints due to the model hypotheses:

- No take off of the stance foot: $F^z > 150N$.
- No slippage of the stance foot: $|\frac{F^x}{F^z}| < 0.5$.
- Swing foot strictly above the ground during a step: the swing foot altitude profile must be above a parabola with a height of 9 cm. ($\star$)
- Vertical velocity of the swing leg strictly positive at the beginning of the step.
- No take off of the leg that impacts the ground: $I^z > 0$.
- No slipping of the leg that impacts the ground: $|\frac{I^x}{I^z}| < 0.5$.

Constraints due to the physical limits of the robot:

- Joint limits. q_1, $q_2 \in [\frac{\pi}{2}, \frac{3\pi}{2}]$ and q_3, $q_4 \in [-\frac{3\pi}{5}, 0]$.
- Maximum joint torque: 60 Nm. ($\star\star$)
- Maximum joint velocity: 2 rad/s.

User's defined constraints:

- Mean walking speed between 0.3 m/s and 0.7 m/s.
- Step length between 0.25 m and 0.6 m.
- Torso inclination q_5 inferior to 0.15 rad.
- Torso movement inferior to 0.2 rad.
- Robustness to perturbations: Minimum kinetic energy level during a step superior to 2 J. ($\star\star\star$)

($\star$) It is important that the swing foot be not close to the ground during a step, otherwise the robot may impact the ground too early (foot scuffing) if the trajectory tracking is not perfect or in case of irregular terrain. For that, we impose that the swing foot altitude profile be always above a parabola. We remarked that imposing such a behavior increases the torque consumption.

($\star\star$) Having torque reserve for quickly converging to the desired motion, especially when the robot is subject to disturbances is important. Otherwise, when the torque reaches its bounds, it is saturated for safety reasons. Consequently, the tracking of the virtual constraints is deteriorated and may be not good enough: the motion of the robot gets jerky and the robot may fall. We especially remind that joint friction is not modeled. Then, experimental joint torque is necessarily higher than the torque computed in simulation. That's why the maximum torque threshold taken in the optimization is inferior to the theoretical [5] maximum torque (120 Nm). We did the same for the velocity.

($\star\star\star$) The walking trajectory must be robust to perturbations, i.e. the size of the basin of attraction of the periodic orbit of the hybrid zero dynamics $(\theta; \dot{\theta})$ must be "large".

The risk is that if the robot loses too much kinetic energy, then it falls backward, i.e. $\dot{\theta}$ zeros and changes sign. More precisely, during the first phase of a step, the altitude of the center of mass of the robot increases, i.e. its kinetic energy is converted into potential energy. If the reserve of kinetic energy is enough, the robot crosses the potential energy barrier, then its altitude decreases and its kinetic energy increases until impacting the ground, and so on.

Actually, the hybrid zero dynamics of a planar one degree underactuated robot is energy conservative and so similar to those of an inverted pendulum model subject to the gravity (page 160 of [127]). Inspired by this [6], we define a minimum level of kinetic energy that the robot may never cross during a nominal step. Choosing a sufficiently high level guarantees that the robot at the beginning of one step has a kinetic energy reserve high enough to continue to walk without falling backward, and even in the presence of moderate perturbations, i.e. the basin of attraction of the orbit is "large". We remarked that increasing the mininum kinetic energy level tends to increase torque consumption. The threshold is increased until a good compromise between robustness and torque consumption is found.

The stability of the walking trajectory is checked a posteriori by modifying manually the state of the robot at the beginning of a step (= right before impact),

[5] This theoretical torque is computed using the average yield given by the manufacturers of the reducers. We do not know in what extend this data represents reality. It is then preferable to be far from this bound

[6] [127] (Theorem 5.3) defines algebraic criteria for the existence and the stability of a periodic orbit in the hybrid zero dynamics. These criteria also quantify the quantity of energy that may be removed to the robot without it falls. Given that we work with the full system (see 3.2.2.5), and we do not restrict the study to the hybrid zero dynamics, we cannot directly apply these results.

especially the value of $\dot{\theta}$, and integrating the dynamical equations to see if the robot converges to the walking cycle. This is roughly a Poincaré analysis.

Actually, as explained in Section 2.4.7, the stability of the hybrid zero dynamics is not difficult to obtain when the degree of underactuation is equal to one.

Solver used and performances. The optimization problem is solved using *fmincon()* of MATLAB. Integration of the full dynamics is performed using *ode45*. The initialization of *fmincon()* is done using a randomly chosen anthropomorphic trajectory designed by hand, or using trajectories found for a planar five-link walker without the 2 bars mechanism, since we worked with such a model previously. After some iterations, i.e. taking as a new initial condition the trajectory returned by *fmincon()*, and adding, modifying the constraints, stable walking trajectories respecting all the constraints (if it is possible to meet them) are found. All this process roughly takes between 5 and 30 minutes using a laptop with an 1.3 GHz Intel core i5. The same optimization performed on the planar five-link walker without the stabilizing mechanism, and integrating only the hybrid zero dynamics, is roughly 3 times faster.

Results A walking trajectory solution of (F.1) is depicted in Figures 3.4, 3.5, 3.6, 3.7, 3.8 and 3.9. Bézier coefficients and some of the properties of the walking trajectory are gathered in table 3.1. An input-output linearizing controller (see 2.4.7) is used to enforce the virtual constraints. The treadmill velocity is set to 0.61 m/s. Actually, the robot converges to a periodic motion for a treadmill velocity comprised between 0.4 and 0.8 m/s. When exceeding these bounds, the robot goes out the treadmill. Note in figure 3.5 the jump on the swing knee velocity due to the impact on the ground (for example, $t = 21.51$ s for the left sagittal knee). For the others joints, the jump is smaller.

	a_{0i}	a_{1i}	a_{2i}	a_{3i}	a_{4i}	a_{5i}	a_{6i}
Stance hip	3.6104	3.5858	3.5847	3.2598	3.1704	3.1411	3.1386
Stance knee	-0.6784	-0.6600	-0.6324	-0.4932	-0.5060	-0.5799	-0.6519
Swing hip	3.1386	3.1519	3.2439	3.3928	3.6124	3.6158	3.6104
Swing knee	-0.6519	-0.7326	-0.8219	-1.1873	-0.8422	-0.7758	-0.6784

θ^- (rad)	θ^+ (rad)	$\dot{\theta}^+$ (rad/s)	Mean walking Speed (m/s)	Step length (m)	Cost (Nm^2s/m)
-2.906	-3.359	0.954	0.61	0.34	5 503

Table 3.1: Bézier coefficients and some properties of the walking trajectory.

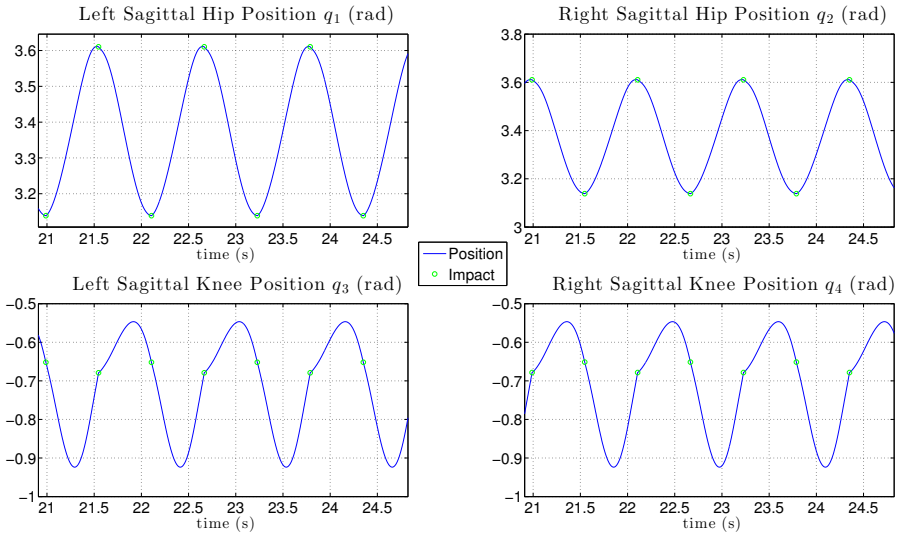


Figure 3.4: Simulated joint position evolutions for a walking trajectory obtained with the HZD method. The simulation uses a rigid ground model.

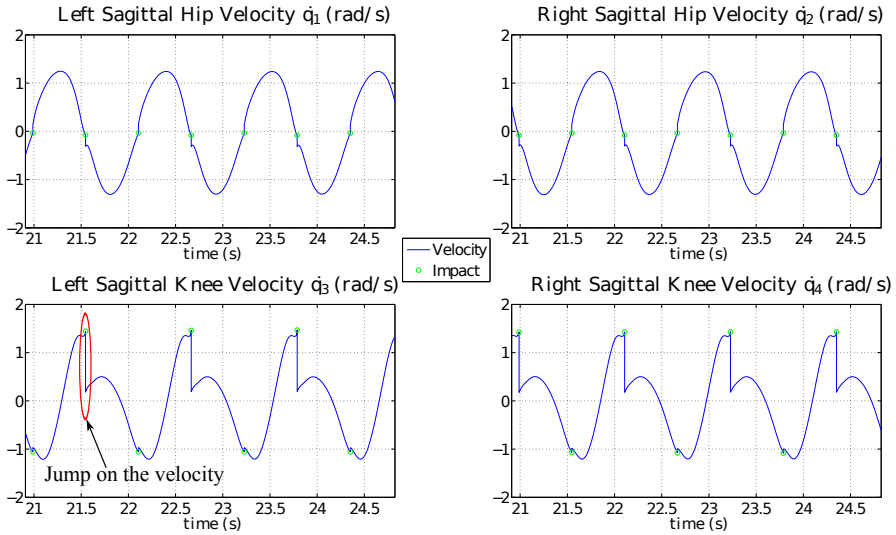


Figure 3.5: Simulated joint velocity evolutions for a walking trajectory obtained with the HZD method. Note the big jump on the knee joint velocity. Not surprisingly, it happens when the corresponding shin hits the ground. The jumps on the others joints is smaller.

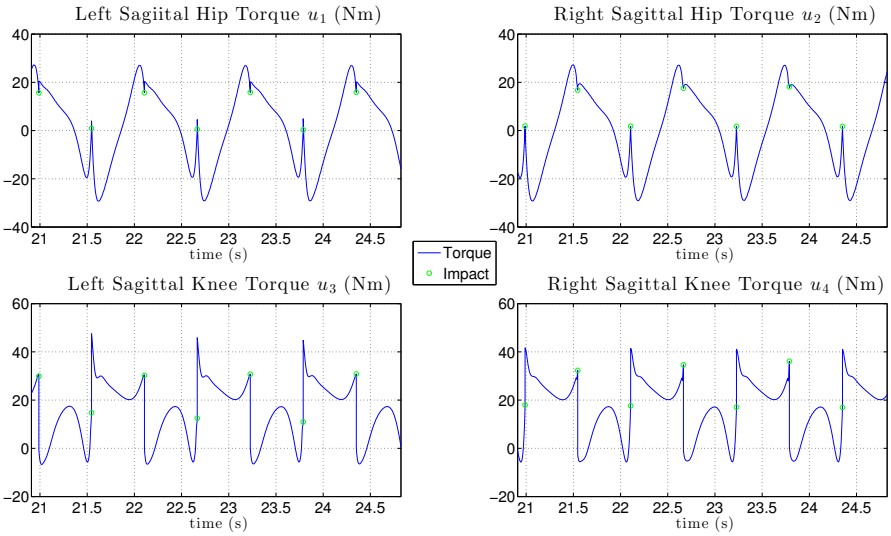


Figure 3.6: Simulated joint torque evolutions using input-output linearization for a walking trajectory obtained with the HZD method. Note that the maximum absolute value of the joint torques is smaller than 60 Nm, a bound used in the motion planning algorithm.

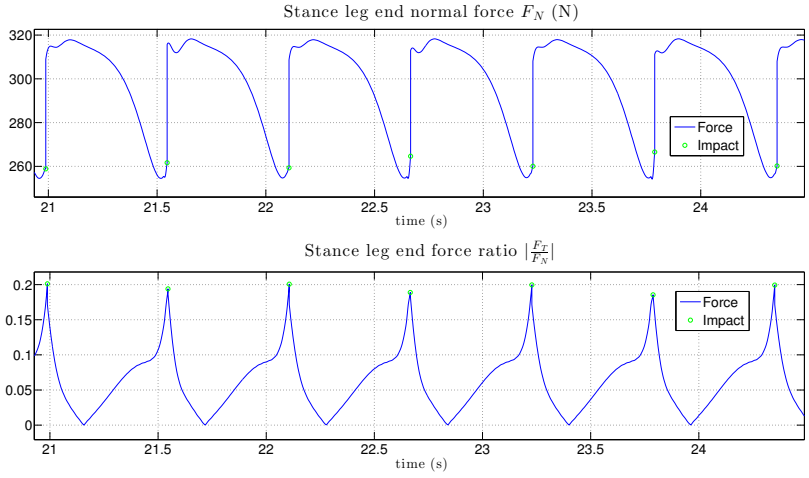


Figure 3.7: Simulated ground reaction forces evolutions for a walking trajectory obtained with the HZD method. Note that the normal ground reaction force F_N is greater than 150 Nm, and that $|\frac{F_T}{F_N}| \leq 0.5$, as imposed in the motion planning algorithm.

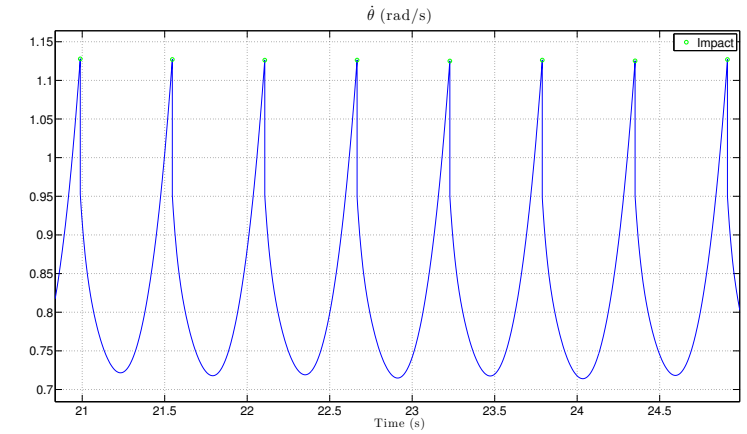


Figure 3.8: Simulated evolution of $\dot{\theta}$ for a walking trajectory obtained with the HZD method.

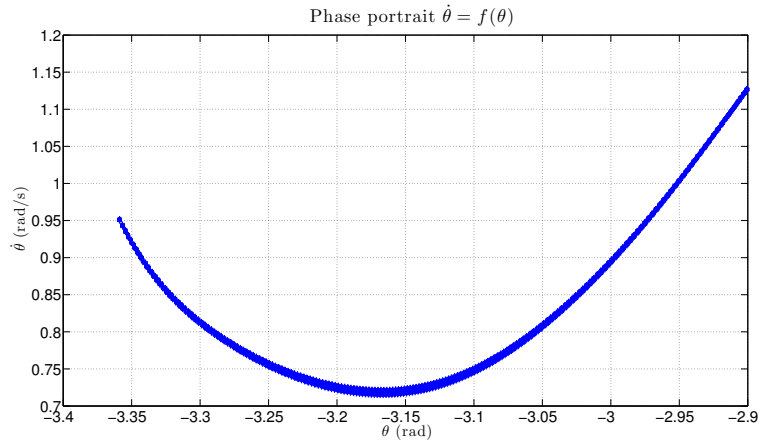


Figure 3.9: Simulated phase portrait of the hybrid zero dynamics $(\theta, \dot{\theta})$ for a walking trajectory obtained with the HZD method.

3.4 Experimental Implementation

We report here the method, the observations, and the experimental choices that we used to implement the walking trajectories designed in simulation on the real robot VS.

3.4.1 Methodology

We tried to be as methodic and rigorous as possible to achieve a stable walk as predicted by the theory. For that, we proceeded step by step to facilitate the identification of the problems that we might have encountered (software problems, hardware problems, errors of modeling,...). We proceeded as follows:

- 1. Robot on air: Tracking time-parameterized trajectories for one and several joints.
- 2. Robot on air: Tracking virtual constraints for one and several joints.
- 3. Robot on the ground: Tracking virtual constraints to achieve one step with the treadmill off.
- 4. Robot on the ground: Tracking virtual constraints to achieve several steps and walking.

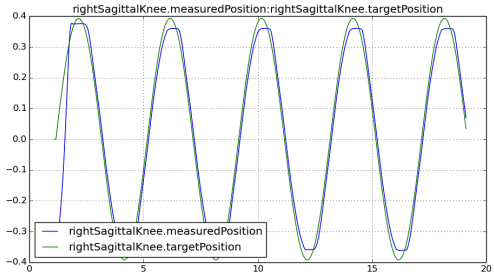
For each steps, we compared the simulation and experimental results. We endeavored to explain the differences that we noticed. We always tried to implement the simplest solutions to maximize the robustness and facilitate debugging.

3.4.2 Effects of Joint Friction

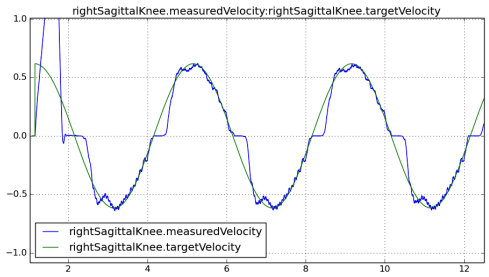
The first experiments with the robot on air aimed at getting familiar with the testbed and to have insights into the tracking performances achievable on the robot VS. As depicted in Figure 3.10 (upper plots), when setting the gains of a classical PD controller on the robot, we noticed a phase shift between the measure and the reference of the joint position, clipping on both the measured position and velocity, and jittering on the measured velocity. We remarked that the phase shift and clipping almost disappeared when using sufficiently high gains for the PD controller.

Since we did not obtain such a behavior in simulation, we added the joint friction model (1.104). We obtained the bottom plots of figure 3.10. The friction parameters were tuned by hand to obtain a behavior similar to the experiments. We noticed that the phase shift is due to the viscous term, and that clipping is due to the static term that brakes the joint at low velocity. Hence, these two phenomena are due to joint friction. When adding a discrete control (the control law is updated at 500 Hz) in the simulation, jittering appears. Then, this phenomenon is due to the use of a discrete control in presence of friction.

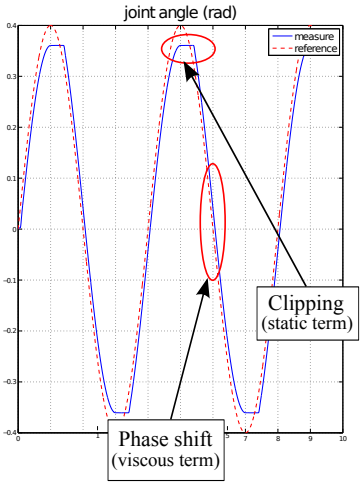
Since the effects of friction could be strongly attenuated by increasing the gains of the PD controller (see Section 3.4.4), we did not feel the necessity of identifying and compensating for friction in the experiments.



(a) Joint position in reality (above) and in simulation (bottom).



(b) Joint velocity in reality (above) and in simulation (bottom).



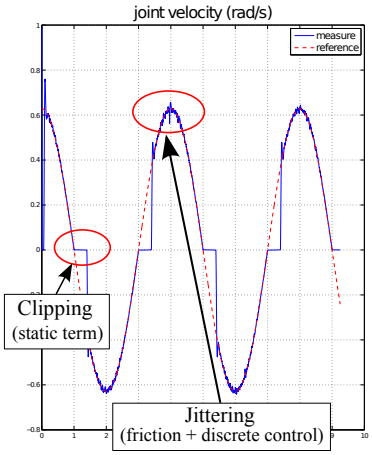


Figure 3.10: Illustration of the effects of joint friction.

3.4.3 Estimation of the Phase Variable θ and of its Velocity $\dot{\theta}$

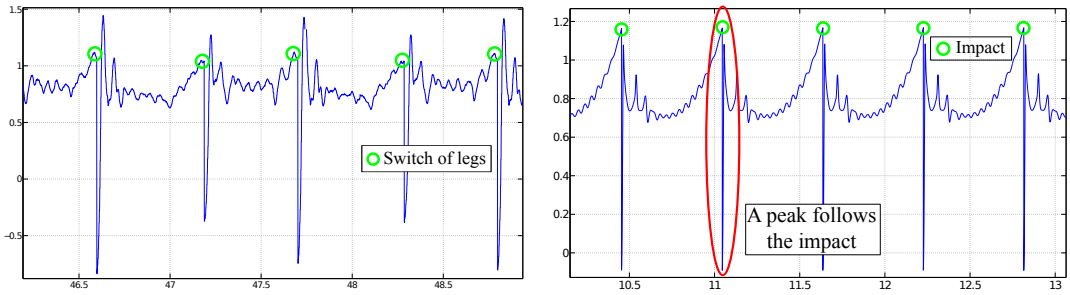
A good estimation of the phase variable θ and of its time derivative $\dot{\theta}$ are important, since they are used to evaluate the virtual constraints, and so to generate the motion of reference for the joints. The phase variable is the virtual stance leg. It is defined as a linear combination of three generalized positions (see (3.11) and (3.12)). θ and $\dot{\theta}$ are calculated using the estimation of q and $\dot{q}$. See Appendix C for further details on how the state of the robot is estimated.

3.4.3.1 Estimation of $\dot{\theta}$ and Effects of the Compliance of the Ground

The experimental estimation of $\dot{\theta}$, noted $\hat{\dot{\theta}}$, for a few walking steps is depicted in figure 3.11. Even though the global shape is the same, there are some differences with the evolution in simulation for a rigid ground model (see figure 3.8). More specifically, the shape is oscillatory and a big peak follows the impact. $\hat{\dot{\theta}}$ even sometimes takes negative values.

Actually, a similar behavior can be obtained in simulation using the compliant ground model described in 1.7.1. See the bottom plot of figure 3.11. In the simulation, peaks are smaller, and the maximum value of $\dot{\theta}$ is higher, even when changing the stiffness and the damping of the model, but the shape is the same. When ignoring oscillations, the minimum and maximum values reached by $\dot{\theta}$ are similar in simulation and in reality.

We realized the experiments using $\hat{\dot{\theta}}$. But, it would be probably better to use an estimator that suppress the big peaks and diminish the oscillations. Indeed, the peak after impact tends to cause a peak on the torques. A reduced-order Luenberger observer as used in [47] may give interesting results .



(a) Experimental evolution of the estimation of $\dot{\theta}$.

(b) Simulated evolution of $\dot{\theta}$ using the compliant ground model described in 1.7.1.

Figure 3.11: Comparison of $\dot{\theta}$ during five steps between experiments and simulations using a compliant ground. The shape is the same. Peaks just after impact and oscillations are due to the compliant nature of the foot-ground contact.

3.4.3.2 Effect of the Mechanical Backlash

It is worth noting that during the first experiments, the motion of the robot was a bit jerky when the left leg was in stance phase. It appeared that there was a mechanical backlash in the left hip. When blocking the motor shaft, the joint angle could be slightly moved by hand. Once the mechanical backlash removed, oscillations almost disappeared. The mechanical backlash probably disturbed the estimation of the phase variable and of its velocity, since no encoders are mounted on the joints.

3.4.4 Performances of a High Gain PD Controller

The high gain PD controller (2.55) was tested on simulation for walking trajectories, using a rigid and a compliant ground model (see 1.7.1). It demonstrated good results.

The experimental tracking performances are very similar. Experimental joint tracking position errors are always inferior or equal to 0.04 rad during the continuous phase. They are inferior to 0.01 rad at impact (see figure 3.15). Figure 3.16 depicts the velocity tracking performances. Joint velocity tracking is less accurate than position tracking, and especially for the knees (tracking errors are up to 0.4 rad/s for the hips and to 2 rad/s for the knees). Note the peaks on the velocity of reference of the knee after impact (see for example the left knee velocity figure 3.16 at $t = 47.51$ s and $t = 48.52$ s). They are due to the peaks of the estimation of $\dot{\theta}$. This corresponds to the times when the knee switches from the stance phase to the swing phase.

The gains of the controller K_P and K_D were tuned by trial and errors both in simulation and in reality.

3.4.5 Choosing When to Switch the Legs

When the swing leg hits the ground, that is at impact, the role of the legs must switch, and the definition of the phase variable changes (see (3.11) and (3.12)). Then, the embedded controller needs to know when this event occurs.

Force Sensitive Resistors (FSR) sensors were first used to detect the contact of the swing leg with the ground. These sensors allowed to correctly detect the impacts. But, we had some troubles when using them. The impact detection thresholds drifted during the experiments (after a few dozens minutes) and sensors progressively deteriorated due to the repetitive impacts (see figure 3.12). We observed that when the impact detection threshold was not set correctly, impact detection might occur too late. This caused the robot to have an asymmetric walk. In the worst cases, this lead to a fall after a few steps.

Due to the aforementioned hardware problems, we tried not to use FSR, and not to detect an impact with the ground. For that, we tested a kinematic switch condition that simply consists in switching the role of the legs when the gait phasing variable exceeds a given threshold. Surprisingly, it works well on flat ground and in the presence of obstacles. Actually, the use of a gait phasing variable

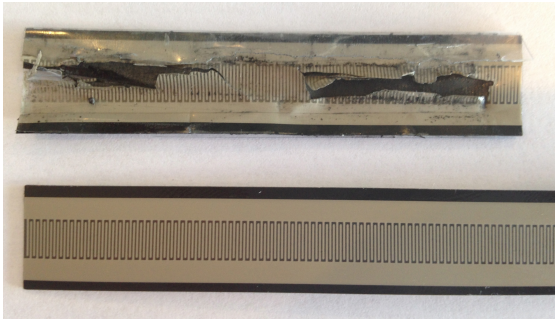


Figure 3.12: Comparison between a FSR sensor deteriorated after some experiments (above), and a new FSR (bottom).

to switch between gait phases for humans is highlighted in [119, 79]. To the best of our knowledge, this is the first time that an actuated biped robot is reported to walk without sensors detecting the contact with the ground. This highlights once again the interest of using virtual constraints in robotic bipedal locomotion, since they synchronize the motion of the robot. Then, if the tracking of the virtual constraints is good enough, the impact always occurs roughly at the same value of the virtual stance leg angle θ.

It is important to correctly set the threshold values. But, contrary to FSR, we noticed that these thresholds do not drift during the experiments. We also remarked that it is not necessary to modify them when changing the walking trajectory. We found better not to choose the same thresholds for the two legs. The thresholds values of the normalized gait phasing variable s were set equal to 0.92 for the left leg, and 0.98 for the right leg (in theory, the impact should occur when $s = 1$). This is maybe due to an asymmetry between the 2 legs.

All the reported walking experiments in this chapter were realized using no sensor to detect the contact with the ground.

3.4.6 Defining a Correct Motion of Reference

Virtual constraints are defined only for the nominal evolution of the phase variable θ, i.e. for $\theta \in [\theta^+, \theta^-]$, or equivalently for $s \in [0, 1]$. The theoretical bounds of evolution of θ were used for the experiments. But, experimentally, due to tracking errors or modeling errors, the phase variable may be out of its theoretical bounds, i.e. s may be inferior to 0 (at the beginning of a step, just after legs swapping) or greater than 1 (at the end of a step).

When $s > 1$, we saturate s to 1, i.e $h_{i,d}(s) = h_{i,d}(1)$ as in [127].

When $s < 0$, the saturation of s to 0, i.e $h_{i,d}(s) = h_{i,d}(0)$ is realized in [127]. But, this does not solve all the problems. First, if the saturation of s goes too long, the robot loses its momentum and is then more likely to fall backward. Furthermore, at the beginning of a new step, the definition of the phase variable changes (see (3.11) and (3.12)). Consequently, if the tracking before impact is not good, or if the robot encounters an obstacle, due to the change of definition of the phase variable, tracking errors may suddently be important after impact, causing

high peak torques.

To avoid this, we use modify online locally the virtual constraints so that the output and its time derivative be zero at the beginning of a step. This is called a deadbeat hybrid extension [77]. The formulas that we used to modify the virtual constraints can be found in Appendix D. We saw in 2.4.4.2 that deadbeat hybrid extension was introduced to design a hybrid zero dynamics. It is also used in practice to avoid the aforementioned problems [111, 22]. Since the estimation of $\dot{\theta}$ quickly changes just after impact (see figure 3.11), it is difficult to define the time derivative of the output: $y'(\theta^+)\dot{\theta}^+$. Therefore, we recompute virtual constraints using only position measurements, i.e. the new virtual constraints zero the output y^+ after impact, but not its time derivative $\dot{y}^+$. In spite of this simplification, experimental results are satisfactory.

Another possibility is to proceed as in [47]. They design motions robust to uneven ground, and so define virtual constraints offline for a phase variable beyond its bounds.

3.4.7 Some Additional Details on the Controller

Note that no experimental identification of the dynamic parameters of the robot was performed. The values used in the simulations were taken from the CAD model of the robot. We do not know in what extend these data describe reality.

The control loop runs at a sampling rate of 500 Hz. Data are recorded at 1kHz. For safety reasons, motor torques are strutted. This causes the joint torques to be saturated at 110. Motors are stopped when tracking errors for at least one joint exceed 0.3 radians.

3.5 Experimental Results

We report here the performances of the walking trajectories that we implemented. We first discuss the experimental results obtained for the walking trajectory described in table 3.4. Then, we present others walking trajectories.

3.5.1 Starting the Robot

For starting the motion, a person gives a push to the robot, and switches on the treadmill. This allows to place the robot in the basin of attraction of the limit cycle of the hybrid zero dynamics. A "slight" push is enough. Its magnitude does not need to be accurately controlled. The velocity of the treadmill is set until the mean absolute position of the robot stays constant. If the velocity is too low, the following behavior is observed. First, the robot goes forward. Secondly, due to the 2 bars mechanism that exerts a force to the back, the robot slows down. Thirdly, either the robot has no kinetic energy enough and reaches the rear of the treadmill, or the robot continues to walk and speeds up to reach again the front of the treadmill, and so on. If the velocity of the treadmill is too high, the robot

reaches the rear of the treadmill. When the robot reaches the rear of the treadmill, the 2 bars mechanism strikes the base frame of the platform. This prevents the robot to go out the treadmill. When the speed of the treadmill is correctly set, it takes the robot only a few steps to converge to a periodic walking.

3.5.2 Walking on Flat Ground

The motion is smooth and almost symmetric. Figure 3.14 depicts a snapshot of the walk. Experimental curves are depicted figure 3.15, 3.16, 3.17, 3.18 and 3.13. The robot converges to a periodic walking when the treadmill velocity is set between 0.4 and 0.64 m/s. For larger velocities, tracking errors at impact become too important (> 0.04 rad) causing an asymmetric walk. Many experiments were realized. The robot could walk until battery depletion (≈ 1 hour).

Joint tracking position errors are always inferior or equal to 0.04 rad during the continuous phase and to 0.01 rad at impact when legs switch their roles. We noticed that it is important to have low tracking error at impact, otherwise the walk is asymmetric and may be unstable. We achieved this by a proper tuning of the high gain PD controller. Joint velocity tracking is less accurate, and especially for the knees (up to 0.4 rad/s for the hips and to 2 rad/s for the knees). The phase variable saturates at the beginning of each step (see figure 3.18), but this is not a problem, especially since we use a deadbeat hybrid extension.

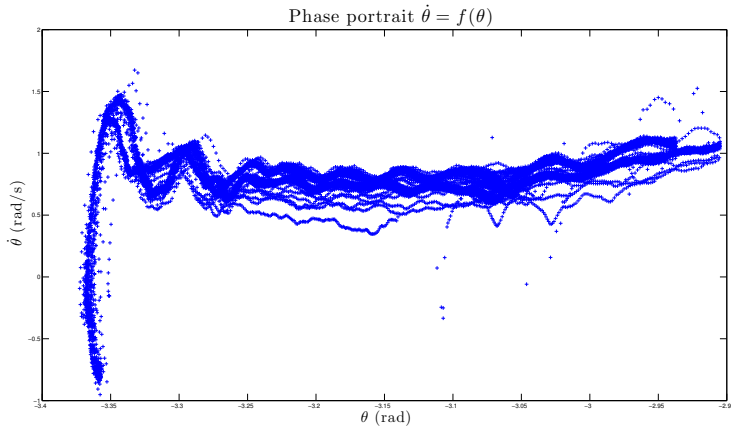


Figure 3.13: Experimental phase portrait for the walk on flat ground. It corresponds to the limit cycle of the hybrid zero dynamics. The spike and the oscillatory behavior is due to the compliance of the foot-ground contact interaction. The experimental behavior is similar to the behavior obtained with simulations on a compliant ground (see 3.4.3.1).

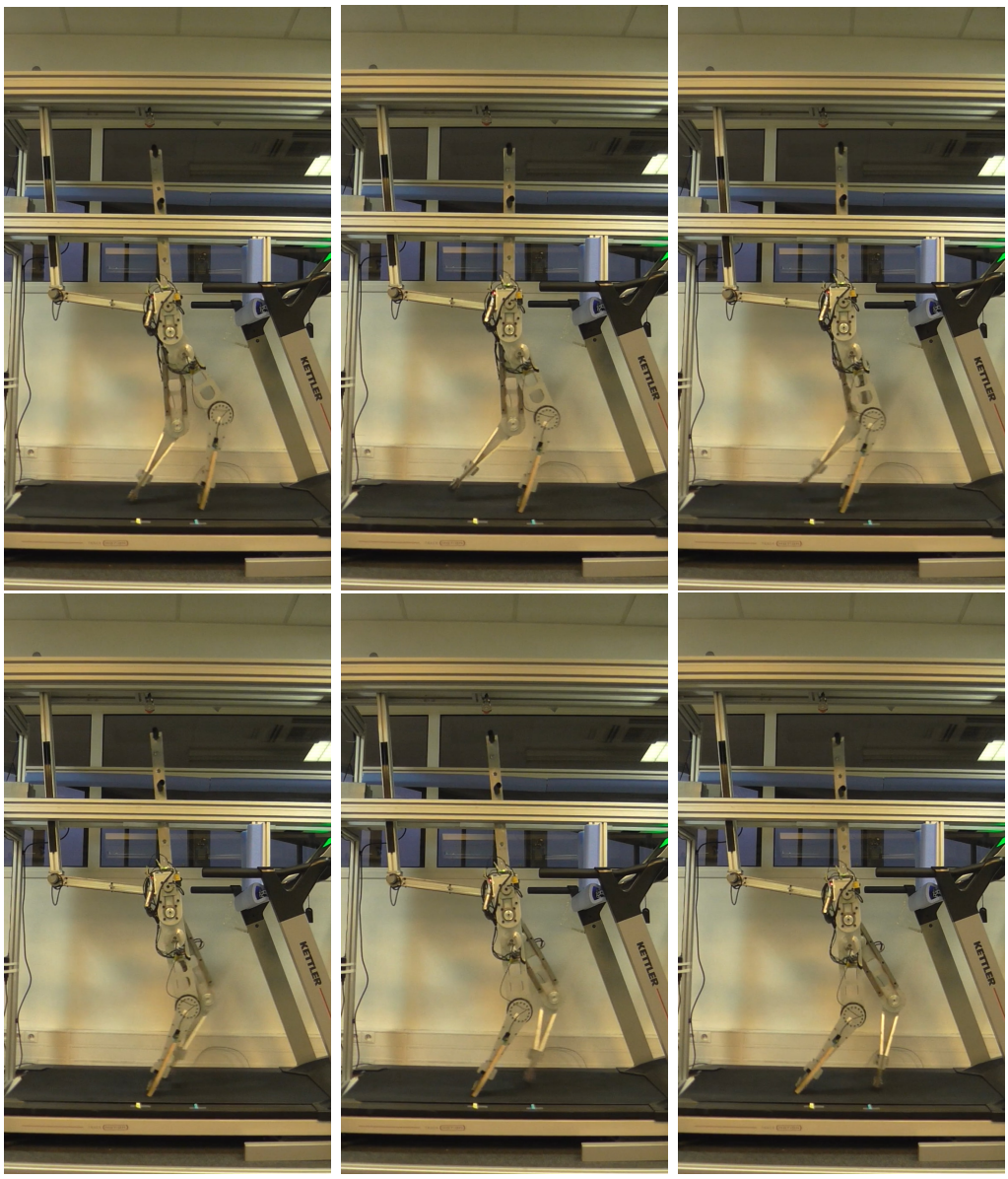
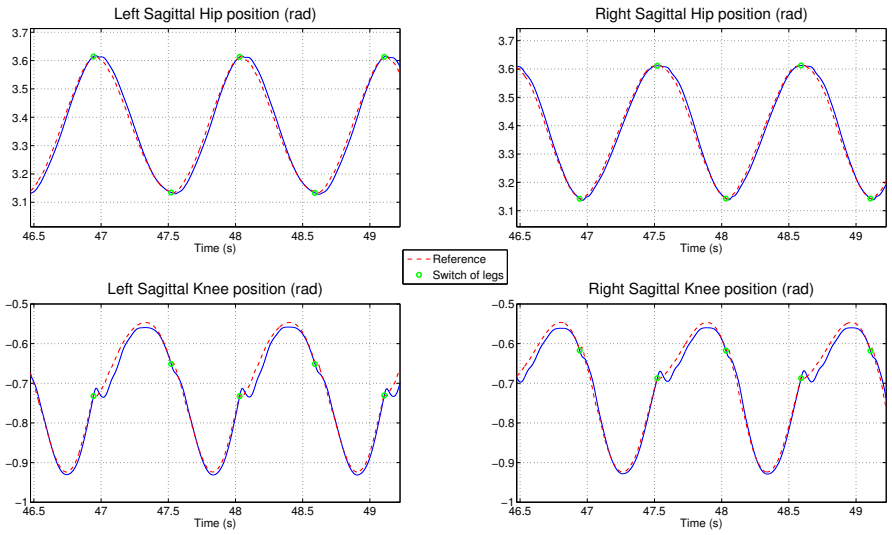
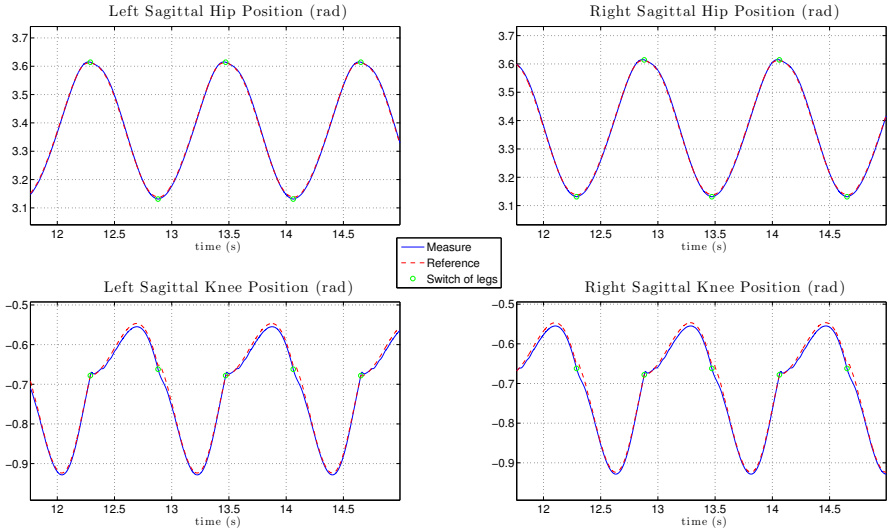


Figure 3.14: Snapshot of a step realized by VS for the walking trajectory described in table 3.1.

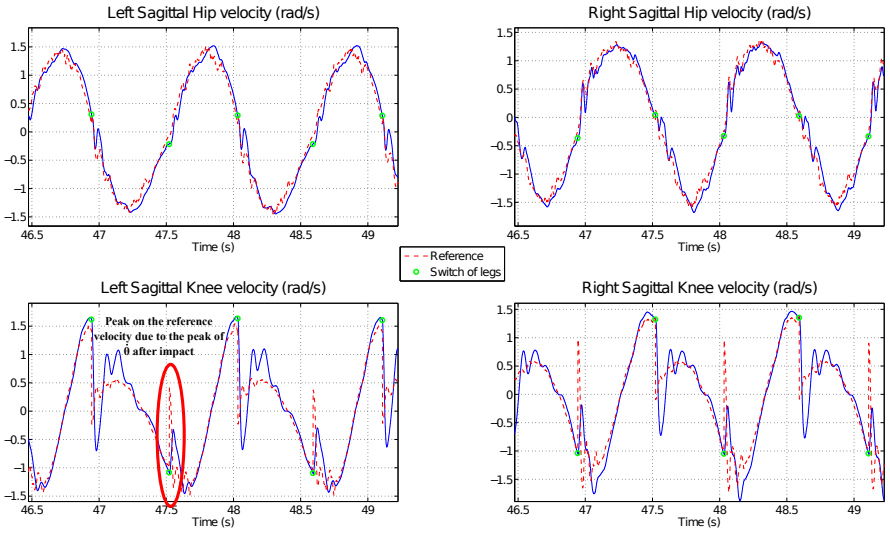


(a) Experimental joint position evolutions.

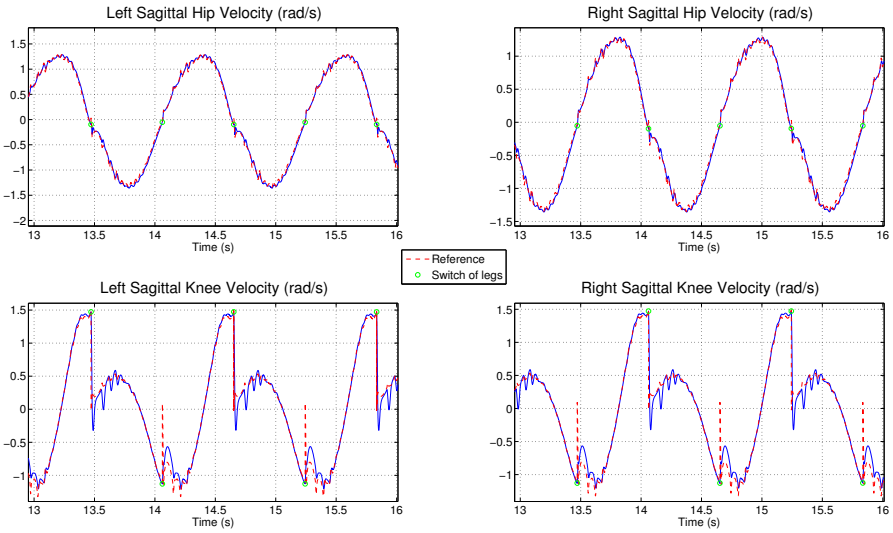


(b) Joint position evolution in simulation with a compliant ground model.

Figure 3.15: Comparison of the joint position tracking performances between simulation and reality for the walk on flat ground. A high gain PD controller is used. Experimental tracking is similar but not as good as in simulations. Joint friction is probably the cause of this difference.



(a) Experimental joint velocity evolutions.



(b) Joint velocity evolution in simulation with a compliant ground model.

Figure 3.16: Comparison of the joint velocity tracking performances between simulation and reality for the walk on flat ground. A high gain PD controller is used. Both evolutions are similar.

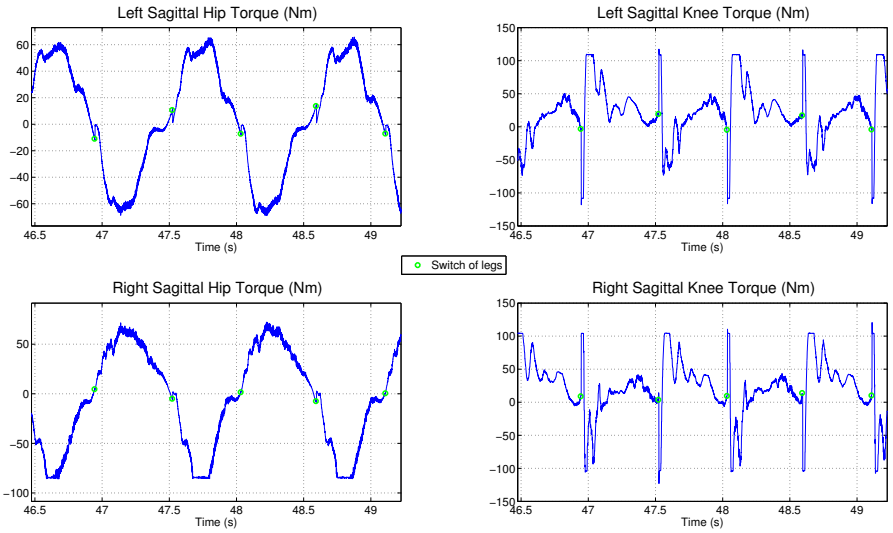


Figure 3.17: Experimental joint torques for the walk on flat ground. Note that the behavior for the left and right hip is not exactly symmetric. This is maybe due to the fact that the motor is not the same for each leg. The left motor broke down and was replaced by an other model.

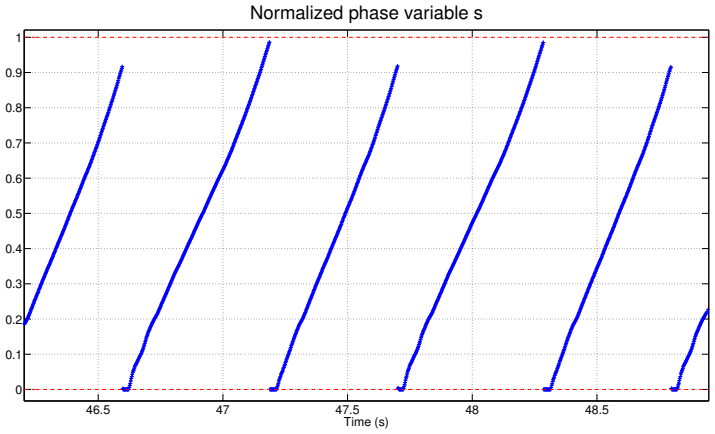


Figure 3.18: Experimental normalized gait phasing variable s for the walk on flat ground. Note that s saturates at the beginning of a step. This issue is addressed by modifying locally the virtual constraints right after impact (see 3.4.6). The role of the legs is switched ("impact detection") when the normalized gait phasing variable reaches a given threshold (see 3.4.5). We found that using a different threshold for each leg gave better results.

3.5.3 Walking in Presence of Disturbances

Some experiments were realized by exerting perturbations on the robot to test its robustness. The HZD method is known to be robust to perturbations such as moderate terrain variations and to dynamical parameters uncertainties [127]. For the next experiments, the controller is exactly the same as on flat ground.

3.5.3.1 Obstacles

The robot is able to negotiate obstacles such as notepads, pieces of wood thick of 25 mm, and even ... a French baguette. After encountering the obstacle, it takes 2 steps for the robot to converge again to the walking cycle. See figure 3.19. We remarked that if the stance leg slips on the obstacle, the robot falls.

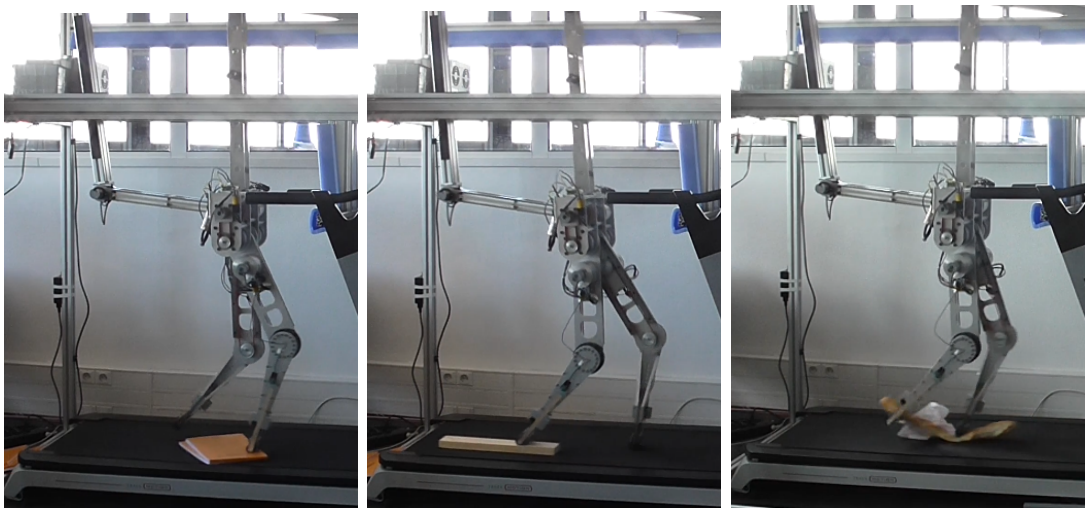


Figure 3.19: VS negociating several types of obstacles with success.

3.5.3.2 Rising slope

The robot is able to walk continuously on a 5 % slope. It can negotiate a slope up to 8 %, but the walk is asymmetric and the robot falls during some experiments.

Additionnal masses on the torso A mass of 5 kg was added at the end of the torso. This additional mass represents roughly 30 % of the mass of the torso. The robot walks continuously, but at a lower speed (0.5 m/s). The walk is a bit asymmetric. When adding supplementary masses, the robot makes a few steps and falls backward. Due to the additional masses, the robot has not enough kinetic energy to complete the next step.

3.5.4 Others Walking Trajectories

Others walking trajectories were successfully implemented. A trajectory with the knees more extended is depicted figure 3.20 and a trajectory with smaller steps figure 3.21.

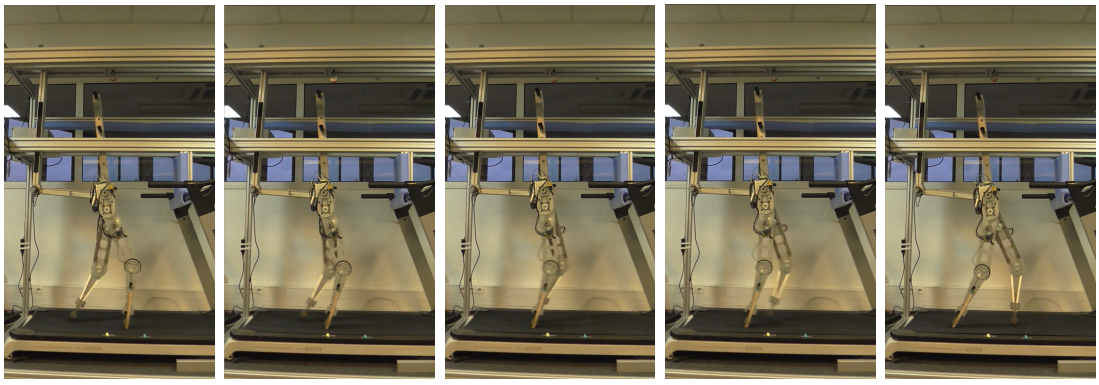


Figure 3.20: Snapshot of a second walking trajectory (knees are more extended). The treadmill velocity is 0.64 m/s.

Figure 3.21: Snapshot of a third walking trajectory (smaller steps). The treadmill velocity is 0.33 m/s.

3.6 Conclusion

This chapter presented the design and the implementation of walking trajectories for a planar five-link walker, called VS, using the HZD method. The robot is able to walk continuously and is robust to moderate perturbations. Important features are: a) VS walks with a lateral stabilizing mechanism which is not translational invariant and has a huge influence on the zero dynamics of the robot. Due to this mechanism, we found that the derivation of the hybrid zero dynamics was not easy. Then, we preferred to find a periodic orbit for the full model. This is not exactly in the spirit of the HZD method which proposes to find a periodic orbit for the hybrid zero dynamics only. b) walking is obtained without using any sensors to detect the contact with the ground, even in the presence of moderate perturbations. Instead of keeping using the fragile contact sensors, we preferred using the measure of the phase variable to decide when the robot should switch the role of the legs.

The experimental behavior is close to the simulations, and the differences between the theory and the practice were explained. More specifically, we highlighted the effects of joint friction on the tracking of the virtual constraints, and the effects of the compliance of the ground on the estimation of $\dot{\theta}$.

Even though the results are fully satisfactory, we see three improvements that may be realized. First, it would be interesting to improve the estimation of $\dot{\theta}$, i.e. removing the peaks right after impact and the oscillations, to allow the use of the velocity measurement to locally modify the virtual constraints right after impact (see Apendix D). Secondly, a deeper investigation of the difference of behavior obtained without and with the use of sensors to detect the contact with the ground is to be done. This requires finding sensors more robust than FSR or/and to find a better design of the feet, which is the third improvement. Especially, more rigid feet may render the foot-ground contact interaction closer to the rigid model resulting in less oscillations of $\dot{\theta}$.

Chapter 4

A Study on the Relative Kinetic Energy Dissipation at Impact for Biped Robots

Résumé français Les trajectoires expérimentales du robot VS (voir chapitre 3) sont bruyantes. L'impact du pied de vol sur le sol cause des vibrations dans la structure du robot, et le tapis roulant est tombé en panne plusieurs fois à cause des impacts répétés. En réalité, d'après les simulations, il s'avère que plus de la moitié de l'énergie cinétique du robot est dissipée à l'impact. Il vient alors la question de la possibilté de faire la syntèse de trajectoires qui dissipent moins d'énergie à l'impact, et de manière plus générale de l'utilité de contrôler la dissipation d'énergie. Ce chapitre s'intéresse à cette problématique.

Nous ne sommes pas les premiers à nous intéresser à ce problème. Miossec et Aoustin [74] ont fait la synthèse de trajectoires de marche qui ne dissipent aucune énergie cinétique à l'impact. Pour cela, ils imposent le pied de vol du robot à atterrir à une vitesse nulle au sol. Nous généralisons ici cette approche en proposant de contrôler, à n'importe quelle valeur, la dissipation relative d'énergie cinétique. Nous proposons deux applications pratiques : la génération de trajectoires avec une faible consommation de couple et une faible dissipation d'énergie à l'impact; la génération de trajectoires dissipant la majorité de l'énergie du robot pour l'arrêter rapidement.

Dans la section 4.2, nous calculons la dissipation relative d'énergie à l'impact en utilisant le modèle d'impact standard rigide introduit par Hurmuzlu (voir section 1.4.1). Nous montrons que l'étude revient à résoudre un problème aux valeurs propres généralisé sous contraintes, les contraintes étant les hypothèses du modèle d'impact. Nous étudions de manière analytique le problème aux valeurs propres généralisé (sans les contraintes), et mettons en évidence des conditions suffisantes sur le contact pied-sol avant et après impact telles qu'il existe des états juste avant impact qui engendrent une dissipation nulle ou totale de l'énergie cinétique du robot à l'impact. Nous vérifions dans un second temps si les contraintes sont respectées sur un modèle numérique.

Dans la section 4.3, nous considérons un robot bipède avec au plus un degré de sous actionnement, et expliquons comment faire la synthèse de mouvements

qui amènent le robot à l'impact avec une dissipation relative d'énergie désirée en modifiant légèrement l'algorithme d'optimisation de la méthode HZD (see Section 2.4.6.1).

La section 4.4 est une application des sections précédentes. Nous comparons les résultats théoriques de la section 4.2 avec les résultats numériques obtenus avec un modèle 5 liens plan avec pieds ponctuels. Nous faisons la synthèse de trajectoires qui dissipent peu d'énergie à l'impact, et qui dissipent près de 80% de l'énergie pour arrêter le robot "en un pas". Enfin, nous mettons en évidence la robustesse du contrôle de l'énergie via une étude en simulation utilisant un modèle (plus réaliste) de sol compliant [88], et introduisant des incertitudes sur les paramètres du robot (masse, inertie, position du centre de masse des membres...), et en considérant des erreurs de suivi de trajectoire.

4.1 Introduction

The experimental walking trajectories of the robot VS (see Chapter 3) are noisy. The impact of the swing leg onto the ground causes vibrations in the base frame, and the treadmill broke down several times. As a matter of fact, from simulations, it was found that the experimental walking trajectories dissipate more than a half of the kinetic energy of the robot at impact.

Hence, the trajectories have the drawback to be noisy, to deteriorate the hardware and do not seem to be suited for the comfort of an exoskeleton user. Then comes the question on the possibility of designing walking trajectories that dissipate less energy at impact, and more generally on the utility of controlling the energy dissipation. This is the core topic of this chapter.

We are not the first to address this problem. Miossec and Aoustin [74] designed walking trajectories with no dissipation of energy at impact for a planar five-link walker. They imposed the swing foot to land on the ground at zero velocity. We generalize this approach by proposing to control any value of the relative kinetic energy dissipation at impact, and propose practical applications.

From now on, for the sake of brevity, the term “energy” will refer to the kinetic energy. First, in Section 4.2, considering a standard hybrid model of biped robot described in Chapter 1, we will investigate the relative energy dissipation at impact using the standard rigid impact model introduced by Hurmuzlu (see Section 1.4.1). Actually, the theoretical study and the numerical examples in this chapter will motivate the study of the relative energy variation instead of the energy variation for control purposes. We will show that investigating the relative energy dissipation comes to solve a generalized eigenvalue problem under constraints, the constraints being the hypotheses of the impact model. We will analytically study the corresponding generalized eigenvalue problem without the constraints and exhibit sufficient conditions on the foot-ground contact before and after impact so that there exist states right before impact causing no dissipation of energy (relative energy dissipation equal to 0), or a full dissipation of the energy (relative energy dissipation equal to -1). The verification of the constraints will be left as a second step on a numerical particular example. Next, in Section 4.3,

we will consider a biped robot with maximally one degree of underactuation and explain how to design motions bringing the robot at impact with a desired relative energy dissipation by using virtual constraints and solving a slightly modified version of the nonlinear programming problem used in the HZD method (see Section 2.4.6.1).

Section 4.4 is an application of the previous sections. We will compare the theoretical results of Section 4.2 with the numerical results obtained using a planar five-link walker with point feet, especially by checking if the impact model hypotheses are verified. We will also design motions bringing the robot with a desired energy dissipation at impact. Two practical applications of the control of the relative energy dissipation are proposed: the design of walking trajectories that dissipate little energy at impact, and the design of motions dissipating more than 80% of the energy to stop the robot "in one step". Finally, we will highlight the robustness of the control of the relative energy dissipation by using a standard (more realistic) compliant ground model [88], introducing a mismatch in the dynamic parameters (mass, inertia, position of the center of mass of the links...), and considering tracking inaccuracy.

4.2 Theoretical Study on the Relative Energy Variation at Impact

We will consider the impact model described in Section 1.4.1. We remind that we consider that the robot is subject to p^- holonomic constraints $\eta^-(q_e) = 0$ before impact. Then, the configuration space is a $(N - p^-)$-dimensional embedded manifold of $\mathcal{Q}_e$

$$\mathcal{Q}_e^{\eta^-} = \{q_e \in \mathcal{Q}_e \,|\, \eta^-(q_e) = 0\}\,. \tag{4.1}$$

The state evolves in the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^-}$ and the dynamical equations are

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e) + G_e(q_e) = B_e u + \left(\dfrac{\partial \eta^-}{\partial q_e}(q_e)\right)^T \lambda \\ \eta^-(q_e) = 0 \end{cases} \tag{4.2}$$

The phase after impact is defined by p^+ holonomic constraints $\eta^+(q_e) = 0$. The configuration space is a $(N - p^+)$-dimensional embedded manifold of $\mathcal{Q}_e$

$$\mathcal{Q}_e^{\eta^+} = \{q_e \in \mathcal{Q}_e \,|\, \eta^+(q_e) = 0\}\,. \tag{4.3}$$

The state evolves in the tangent bundle $\mathcal{T}\mathcal{Q}_e^{\eta^+}$ and the dynamical equations are

$$\begin{cases} D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e) + G_e(q_e) = B_e u + \left(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\right)^T \lambda \\ \eta^+(q_e) = 0 \end{cases} \tag{4.4}$$

The impact describes the jump of the state from $\mathcal{TQ}_e^{\eta^-}$ and $\mathcal{TQ}_e^{\eta^+}$.

4.2.1 Kinetic Energy Variation at Impact

We remind that the kinetic energy variation at impact in the extended coordinates $(q_e; \dot{q}_e) \in \mathcal{TQ}_e^{\eta^-} \subset \mathbb{R}^{2N}$ is

$$\begin{aligned}&\mathrm{VarK}_e(q_e, \dot{q}_e) = \\ &\quad -\frac{1}{2}\dot{q}_e^T \left[\left(\frac{\partial \eta^+}{\partial q_e}(q_e) \right)^T \left(\frac{\partial \eta^+}{\partial q_e}(q_e) D_e(q_e)^{-1} \left(\frac{\partial \eta^+}{\partial q_e}(q_e) \right)^T \right)^{-1} \frac{\partial \eta^+}{\partial q_e}(q_e) \right] \dot{q}_e \,.\end{aligned} \tag{4.5}$$

In the reduced coordinates $(q; \dot{q}) \in \mathcal{TQ}_f^{\eta^-} \subset \mathbb{R}^{2(N-p^-)}$

$$\mathrm{VarK}(q, \dot{q}) = \frac{1}{2}\dot{q}^T A(q) \dot{q} \,, \tag{4.6}$$

where

$$\begin{aligned}A(q) = -\left(\frac{\partial E^{\eta^-}}{\partial q}(q) \right)^T \Bigg[\left(\frac{\partial \eta^+}{\partial q_e}(q_e) \right)^T \left(\frac{\partial \eta^+}{\partial q_e}(q_e) D_e^{-1}(q_e) \left(\frac{\partial \eta^+}{\partial q_e}(q_e) \right)^T \right)^{-1} \\ \times \left(\frac{\partial \eta^+}{\partial q_e}(q_e) \right) \Bigg] \left(\frac{\partial E^{\eta^-}}{\partial q}(q) \right) .\end{aligned} \tag{4.7}$$

with $q_e = E^{\eta^-}(q)$, and $E^{\eta^-}(q)$ is the embedding from $Q_f^{\eta^-}$ to $Q_e^{\eta^-}$.

4.2.2 Investigating the Relative Energy Dissipation

4.2.2.1 The Relative Energy Dissipation is a Suitable Indicator

In the study of the mapping $(q_e, \dot{q}_e) \to \mathrm{VarK}_e(q_e, \dot{q}_e)$ on $\mathcal{TQ}_e^{\eta^-}$, i.e. of the influence of the state before impact on the dissipation of energy, only the geometry is of interest, since it is quadratic relative to the velocity $\dot{q}_e$. Yet, it is more interesting to know if there exist some non trivial values of the state, such that the kinetic energy of the robot is totally dissipated. This comes to study the ratio of energy dissipated at impact, and so to study the relative energy dissipation

$$\frac{\mathrm{VarK}_e(q_e, \dot{q}_e)}{\mathrm{K}_e(q_e, \dot{q}_e)} \,. \tag{4.8}$$

Note that since the impact dissipates energy (1.75), and since we study a relative variation, necessarily

$$-1 \leq \frac{\mathrm{VarK}_e(q_e, \dot{q}_e)}{\mathrm{K}_e(q_e, \dot{q}_e)} \leq 0 \,. \tag{4.9}$$

4.2.2.2 Parameters Dependency of the Relative Energy Dissipation

From (4.5) and (4.8), the relative energy dissipation depends on three parameters

- The nature of the foot-ground contact after impact $\eta^+(q_e) = 0$. It defines in particular how the ground reaction acts on the robot during the impact.
- The configuration of the robot before impact q_e.
- The direction of the velocity of the robot before impact $\dot{q}_e$.

The nature of the foot-ground contact before impact $\eta^-(q_e) = 0$ constrains the state before impact to evolve in an embedded manifold $\mathcal{TQ}_e^{\eta^-}$ of $\mathcal{TQ}_e$. Then, the robot cannot reach any position and velocity just before impact, restricting the possibility of dissipating energy.

4.2.2.3 A Generalized Eigenvalue Problem Under Constraints

We are interested in the values of the relative energy variation, and in highlighting the influence of the parameters presented above.

For that purpose, we propose to consider a given foot-ground contact before impact, corresponding to p^- holonomic constraints $\eta^-(q_e) = 0$. This allows to define a reduced set of coordinates for the robot before impact $(q; \dot{q}) \in \mathcal{TQ}_f^{\eta^-}$ as explained above. In addition, the two first parameters of the previous section, that is the foot-ground contact after impact and the configuration right before impact impact, are chosen a priori.

Proceeding this way facilitates the study. Indeed, this comes to investigate the values of the relative energy variation on the tangent space $\mathcal{T}_q\mathcal{Q}_f^{\eta^-}$, that is of the mapping

$$\begin{aligned} \mathcal{T}_q\mathcal{Q}_f^{\eta^-} &\rightarrow [0,1] \\ \dot{q} &\rightarrow \frac{\mathrm{VarK}(q,\dot{q})}{\mathrm{K}(q,\dot{q})} = \frac{\dot{q}^T A(q)\dot{q}}{\dot{q}^T D(q)\dot{q}}\,, \end{aligned} \tag{4.10}$$

where $D(q)$ is positive definite and $A(q)$ is symmetric negative.

We are especially interested in finding the extremal values of the relative energy variation

$$\min_{\dot{q}\in\mathcal{T}_q\mathcal{Q}_f^{\eta^-}} \frac{\dot{q}^T A(q)\dot{q}}{\dot{q}^T D(q)\dot{q}}\,. \tag{4.11}$$

We will see in the next section that this is equivalent to solve a *generalized eigenvalue problem*.

Since (4.8) was derived using the rigid impact model described in 1.4.1, the impact model hypotheses (1.68) must be verified. Then, finding the admissible

values of the relative energy variation comes to solve the generalized eigenvalue problem under constraints

$$\begin{aligned} &\min_{\dot{q}\in\mathcal{T}_q\mathcal{Q}_f^{\eta^-}} \frac{\dot{q}^T A(q)\dot{q}}{\dot{q}^T D(q)\dot{q}}, \\ &\text{s.t. } C_\Delta^{\eta^+}\Big(E^{\eta^-}(q), \frac{\partial E^{\eta^-}}{\partial q}(q)\dot{q}\Big) \geq 0\,, \end{aligned} \tag{4.12}$$

where $q_e = E^{\eta^-}(q)$.

Several approaches can be used to solve (4.12). We decided to consider the following approach.

Step 1: Solving (4.11) analytically. In a first step, we chose not to consider the constraints (1.68). Indeed, these constraints are model dependent, that is depend on the values of the dynamic parameters (mass, length, inertias...), which complicates an analytical resolution. Hence, we analytically investigate (4.10) and (4.11). This problem is addressed in Sections 4.2.3, 4.2.4 and 4.2.5.

Step 2: Checking numerically the impact model hypotheses (1.68). In a second step, to properly use the obtained results for a particular model of robot, one numerically verifies if the constraints (1.68) are verified. This is done in 4.4.2 for the model of the planar five link walker.

4.2.3 Relative Energy Variation and Generalized Eigenvalue Problem

We show here that the study of the mapping (4.10) and of (4.11) are equivalent to the resolution of a generalized eigenvalue problem.

4.2.3.1 Generalized eigenvalue problem

A generalized eigenvalue problem consists in finding the roots of the characteristic polynomial

$$\det[P - \lambda Q]\,, \tag{4.13}$$

where P and Q are square matrices of same dimension, and $\lambda \in \mathbb{R}$ is the variable of the polynomial. A value of λ zeroing the polynomial is called *generalized eigenvalue* of P and Q. A vector v of the kernel of $P - \lambda Q$, that is verifying $Pv = \lambda Qv$, is called a *generalized eigenvector* associated to λ. Note that the standard eigenvalue problem corresponds to the case where Q is the identity matrix.

It can be proved that the solutions of the generalized eigenvalue problem are the stationary points of the function $x \to \frac{x^T Px}{x^T Qx}$, if Q is positive definite. $x^T Px - \lambda x^T Qx$ is sometimes called a *pencil of quadratic forms*. For further details, see for example the chapter 15 in [86] and [42].

4.2.3.2 Relative energy variation and generalized eigenvalue problem.

Values of the mapping (4.10). The resolution of a generalized eigenvalue problem gives information on the values of the mapping (4.10) and the solutions of (4.11). More especially, if λ is a generalized eigenvalue of $A(q)$ and $D(q)$, that is a root of

$$\det[A(q) - \lambda D(q)]\,, \tag{4.14}$$

then it is a stationary value of the relative energy dissipation (4.10).

Additionally, from [42] (theorem 8 page 310), the characteristic polynomial (4.14) has $N - p^-$ real roots

$$\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_{N-p^-}\,. \tag{4.15}$$

And it follows from [42] (equation (70) page 318) that we have

$$\lambda_1 \leq \frac{\dot{q}^T A(q)\dot{q}}{\dot{q}^T D(q)\dot{q}} \leq \lambda_{N-p^-} \text{ on } \mathcal{T}_q\mathcal{Q}_f^{\eta^-}\,. \tag{4.16}$$

The corresponding generalized eigenvectors, $v_{i,i\in\{1,N-p^-\}}$ ($Av_i = \lambda_i D v_i$) can be chosen such that they are orthonormal with the scalar product defined by the mass matrix $D(q)$, that is

$$v_i^T D(q) v_j = \delta_i^j\,, \tag{4.17}$$

where

$$\delta_i^j = 1 \text{ if } i = j, \text{ else } 0. \tag{4.18}$$

Decomposition of the tangent space $\mathcal{T}_q\mathcal{Q}_f^{\eta^-}$. From (4.17), the tangent space $\mathcal{T}_q\mathcal{Q}_f^{\eta^-}$ can be decomposed as the direct sum of the subspaces generated by the generalized eigenvectors

$$\mathcal{T}_q\mathcal{Q}_f^{\eta^-} = \text{Vect}(v_1) \oplus \text{Vect}(v_2)... \oplus \text{Vect}(v_{N-p^-})\,, \tag{4.19}$$

where $\text{Vect}(v_i)$ is the subspace generated by v_i.

Hence, any velocity $\dot{q} \in \mathcal{T}_q\mathcal{Q}_f^{\eta^-}$ can be written as a linear combination of the generalized eigenvectors v_i. Additionally, from [42] (Theorem 11 page 319), for $1 \leq d \leq N - p^-$, any velocity belonging to the subspace

$$\text{Vect}(v_d) \oplus \text{Vect}(v_{d+1})... \oplus \text{Vect}(v_{N-p^-})\,, \tag{4.20}$$

corresponds to a relative energy dissipation comprised between λ_d and λ_{N-p^-}.

As a result, studying the relative energy dissipation comes to compute, for each position $q \in \mathcal{Q}_f^{\eta^-}$, generalized eigenspaces in $\mathcal{T}_q\mathcal{Q}_f^{\eta^-}$.

4.2.3.3 A Deeper Investigation of λ_1 and λ_{N-p^-}

We have just seen that the relative energy dissipation, without considering the constraints imposed by the impact model, is comprised between λ_1 and λ_{N-p^-}. From (4.9), we have

$$-1 \leq \lambda_1 \leq \lambda_{N-p^-} \leq 0\,. \tag{4.21}$$

We will investigate now in the two next sections, under which conditions on the foot-ground contact, and on the configuration at impact, $\lambda_{N-p^-} = 0$ (no dissipation) and $\lambda_1 = -1$ (total dissipation of the energy), that is under which conditions the relative energy dissipation ratio at impact can take any value. We will also determine the corresponding states before impact.

As a first step, in 4.2.4, we will investigate the problem on $\mathcal{TQ}_e$, that is in the case where the robot is subject to no constraints before impact (flight phase). Next, in 4.2.5, we will consider the more restrictive cases where the robot is subject to holonomic constraints before impact due to the foot-ground contact, constraining the reachable states before impact and the values of the relative energy dissipation.

During this study, we will assume that $\frac{\partial \eta^-}{\partial q_e}(q_e)$ and $\frac{\partial \eta^+}{\partial q_e}(q_e)$ are full rank. Hence

$$\begin{cases} rank\Big(\dfrac{\partial \eta^-}{\partial q_e}(q_e)\Big) = p^- \\ rank\Big(\dfrac{\partial \eta^+}{\partial q_e}(q_e)\Big) = p^+ \\ rank\Big(\dfrac{\partial E^{\eta^-}}{\partial q}(q)\Big) = N - p^- \end{cases} \tag{4.22}$$

4.2.4 The Particular Case of Flight Phase Before Impact

We consider here that the robot is in flight phase before impact, i.e. the robot is not in contact with the ground ($p^- = 0$). Such a phase is met in the case of running. Then, any state is theoretically reachable before impact, and the study is performed on the entire manifold $\mathcal{TQ}_e$, i.e. $q = q_e$ and E^{η^-} is the identity. After impact, it is assumed that the robot is subject to p^+ holonomic constraints $\eta^+(q_e) = 0$.

We prove here that the generalized eigenvalues of (4.14) are necessarily -1 or 0. Then, a robot in flight phase, with an appropriate choice of state, can theoretically[1] land on the ground by dissipating between 0 and 100 % of its kinetic energy at impact. This particular case will be useful to treat the case of the robot in contact with the ground before impact.

[1] Actually, the actuator limits and the validity of the impact model have to be also considered.

4.2.4.1 No Dissipation of Energy

Let $V_0^{\eta^+}$ be the generalized eigenspace associated to the generalized eigenvalue 0. The relative energy dissipation is zero if and only if the energy dissipation is zero. Then, from 1.4.3.2, we have

$$V_0^{\eta^+} = Ker\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)\,, \tag{4.23}$$

where $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is a $p^+ \times N$ matrix and is full rank by hypothesis. We deduce from the nullity rank theorem that

$$dim(V_0^{\eta^+}) = N - p^+. \tag{4.24}$$

4.2.4.2 Total Dissipation of Energy

Let $V_{-1}^{\eta^+}$ be the eigenspace associated to the generalized eigenvalue -1 (100% of the energy is dissipated). Let $q_e \in \mathcal{Q}_e$, we look for $v \in \mathcal{T}_{q_e}\mathcal{Q}_e$ such that

$$\left[\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\left(\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\right)^{-1}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)\right]v = D_e(q_e)v\,, \tag{4.25}$$

or equivalently

$$D_e^{-1}(q_e)\left[\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\left(\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\right)^{-1}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)\right]v = v\,. \tag{4.26}$$

Let v be obtained as

$$v = D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T w\,, \tag{4.27}$$

with

$$w \in \mathbb{R}^{p^+}\,. \tag{4.28}$$

Then, we get

$$\begin{aligned} &D_e^{-1}(q_e)\left[\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\left(\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T\right)^{-1}\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)\right]v \\ &\quad = D_e^{-1}(q_e)\Big(\frac{\partial \eta^+}{\partial q_e}(q_e)\Big)^T w\,. \end{aligned} \tag{4.29}$$

So any such v is solution. This shows that

$$Im\left(D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right) \subset V_{-1}^{\eta^+}\,. \tag{4.30}$$

Additionnally, since $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank by hypothesis , and $D_e^{-1}(q_e)$ is a positive definite matrix, then

$$dim\left(Im\left(D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right)\right) = p^+. \tag{4.31}$$

As a result, from (4.19), we have

$$p^+ \leq dim(V_{-1}^{\eta^+}) \leq N - dim(V_0^{\eta^+})\,. \tag{4.32}$$

Then, using (4.24) gives

$$dim(V_{-1}^{\eta^+}) = p^+. \tag{4.33}$$

This proves that the subspace of the velocities resulting in a total dissipation of energy is

$$V_{-1}^{\eta^+} = Im\left(D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right). \tag{4.34}$$

This is a vector space of dimension p^+.

4.2.4.3 Summary for the Case of Flight Phase Before Impact

We proved the following result:

Let a model of biped robot in flight phase be as described in Chapter 1. The state space $\mathcal{TQ}_e$ is of dimension $2N$. The foot-ground contact during and after impact is modeled with p^+ holonomic constraints $\eta^+(q_e) = 0$. For any configuration $q_e \in \mathcal{Q}_e$, if $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank, and the hypotheses of the impact model are respected (see Section 1.4.2), then there exist velocities $\dot{q}_e \in \mathcal{T}_{q_e}\mathcal{Q}_e$ such that the energy dissipation at impact is zero, and velocities such that the dissipation of energy is total. Specifically:

- $V_0^{\eta^+} = Ker\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)$ is the set of velocities $\dot{q}_e \in \mathcal{TQ}_e$ such that the dissipation of energy is zero. It is a vector space of dimension $N - p^+$.
- $V_{-1}^{\eta^+} = Im\left(D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right)$ is the set of velocities $\dot{q}_e \in \mathcal{TQ}_e$ such that the dissipation of energy is total. It is a vector space of dimension p^+.
- The tangent space can be partitionned as:
  $\mathcal{T}_{q_e}\mathcal{Q}_e = Ker\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right) \oplus Im\left(D_e^{-1}(q_e)\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right)$.

4.2.4.4 Remarks

Remark 1. Only the kinematics of the robot plays a role in the definition of $V_0^{\eta^+}$. On the contrary, both the kinematics and the dynamics appear in $V_{-1}^{\eta^+}$. The dynamics contribution is only due to the presence of the mass matrix $D_e(q_e)$.

The physical meaning of $V_0^{\eta^+}$ is quite straightforward. See Section 1.4.3.2.

The physical meaning of $V_{-1}^{\eta^+}$ is less clear. We do not know how to interpret the term $D_e(q_e)^{-1}$, and could not relate this set with a particular behavior of the swing foot such as the orientation of the velocity along a given direction. But, we can remark that its dimension is equal to the rank of $\eta^+(q_e)$, i.e. to the number of degrees of freedom locked by the feet-ground contact during and after impact. Actually, the ground reaction impulse works only along these locked degrees of freedom, see (1.73).

Remark 2. Actually, we have that the only generalized eigenvalues are 0 and 1. This is consistent with the fact that

$$P = D_e(q_e)^{-1}\left[\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\left(\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)D_e(q_e)^{-1}\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)^T\right)^{-1}\left(\frac{\partial \eta^+}{\partial q_e}(q_e)\right)\right] \tag{4.35}$$

is an idempotent matrix, i.e. $P^2 = P$.

4.2.5 A More General Case: Robot Subject to Holonomic Constraints Before Impact

We consider now that the robot is in contact with the ground before impact. The foot-ground contact is described by p^- holonomic constraints $\eta^-(q_e) = 0$ and constrains the configuration of the robot to evolve in a $N - p^-$ dimensional submanifold $\mathcal{Q}_e^{\eta^-}$ of $\mathcal{Q}_e$. The state is then constrained to evolve in a $2(N - p^-)$ dimensional embedded manifold $\mathcal{TQ}_e^{\eta^-}$ of $\mathcal{TQ}_e$.

The key point is that now only the velocities in the image of $\frac{\partial E^{\eta^-}}{\partial q}(q)$ (see (1.76)), that is under the form

$$\dot{q}_e = \frac{\partial E^{\eta^-}}{\partial q}(q)\dot{q}\,, \tag{4.36}$$

where $q \in \mathcal{Q}_f^{\eta^-} \subset \mathbb{R}^{N-p^-}$, and $\dot{q} \in \mathcal{T}_q\mathcal{Q}_f^{\eta^-}$, are reachable. We figure out here under which conditions there exist such velocities resulting in no dissipation, or a total dissipation of the energy, that is belonging respectively to $V_0^{\eta^+}$ and $V_{-1}^{\eta^+}$.

4.2.5.1 No Dissipation of Energy

From (4.23), for a given configuration $q \in \mathcal{Q}_f^{\eta^-}$, we know that $\dot{q} \in \mathcal{T}_q\mathcal{Q}_f^{\eta^-}$ dissipates no energy if and only if

$$\dot{q}_e = \frac{\partial E^{\eta^-}}{\partial q}(q)\dot{q} \in Ker\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\right). \tag{4.37}$$

Hence

$$\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\right)\dot{q} = 0\,. \tag{4.38}$$

Therefore, the nullspace of $\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)$ is the set of the velocities that dissipate no energy at impact. Let us investigate now the dimension of this space.

From classical results of linear algebra gathered in Appendix E, we have

$$\begin{cases} rank\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\right) \leq min\left(rank\Big(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\Big), rank\Big(\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)\right). \\ rank\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\right) + rank\left(\frac{\partial E^{\eta^-}}{\partial q_e}(q)\right) - N \leq rank\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\right). \end{cases} \tag{4.39}$$

From (4.22), we get

$$p^+ - p^- \leq rank\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\right) \leq min\Big(p^+, N - p^-\Big). \tag{4.40}$$

Then, from the rank-nullity theorem

$$N - p^- - min(p^+, N - p^-) \leq dim\left(Ker\Big(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)\right) \leq N - p^+. \tag{4.41}$$

There are two possibilities, depending on the sign of the left-hand side of this last inequality.

Case 1. If $N-p^- - min(p^+, N-p^-) > 0$, this is equivalent to $min(p^+, N-p^-) = p^+$, and so to $N - p^- > p^+$. Then

$$0 < N - p^- - p^+ \leq dim\left(Ker\Big(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\Big)\right) \leq N - p^+\,. \tag{4.42}$$

There exist velocities resulting in no energy dissipation at impact. This case is encountered for most of biped robots. As a matter of fact, the inequality $N - p^- > p^+$ implies that it is possible for the robot to dissipate no energy at

impact if it has enough degrees of freedom. For that, N must be "large", or p^- "small". This is the case for the planar five-link robot in single support phase before impact and for more complex models. But, this is not the case for the compass walker.

Case 2. If $N - p^- - min(p^+, N - p^-) \leq 0$, one cannot conclude, except when[2] $p^+ = N$. In this latter case, there exist no velocity that result in no dissipation of energy at impact.

4.2.5.2 Total Dissipation of Energy

Let a configuration $q_e \in \mathcal{Q}_e^{\eta^-}$. Due to the holonomic constraints, the velocity $\dot{q}_e \in \mathcal{T}_{q_e}\mathcal{Q}_e^{\eta^-}$ verifies

$$\dot{q}_e \in Ker(\frac{\partial \eta^-}{\partial q_e}(q_e))\,. \tag{4.43}$$

And from (4.34), we know that $\dot{q}_e$ results in a total dissipation of the energy if and only if there exists $v \in \mathbb{R}^{p^+}$ such that

$$\dot{q}_e = \Big(D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)v\,. \tag{4.44}$$

This implies

$$\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)v = 0\,. \tag{4.45}$$

The converse is trivially true: (4.45) $\Rightarrow$ { (4.44) + (4.43) }.

Hence, the problem lies on computing the null space of $\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)$, a matrix of dimension $p^- \times p^+$.

Using the nullity-rank theorem, we have

$$dim\left(Ker\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)\right) = p^+ - rank\left(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\right). \tag{4.46}$$

And using classical results of linear algebra in Appendix E gives

$$\begin{cases} rank\left(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\right) \leq min\left(rank\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)\Big), rank\Big(D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)\right) \\ rank\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)\Big) + rank\Big(D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big) - N \leq rank\left(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\right) \end{cases} \tag{4.47}$$

[2]Necesssarlily, $p^+ \leq N$. Indeed p^+ is the rank of $\frac{\partial \eta^+}{\partial q_e}(q_e)$, and this matrix has N rows.

Given that $D_e(q_e)^{-1}$ is positive definite

$$rank\Big(D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big) = rank\Big(\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big). \tag{4.48}$$

Thus, (4.47) gives

$$p^- + p^+ - N \leq rank\left(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\right) \leq min\Big(p^+, p^-\Big). \tag{4.49}$$

And from (4.46)

$$p^+ - min(p^+, p^-) \leq dim\left(Ker\Big(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big)\right) \leq N - p^-. \tag{4.50}$$

There are two possibilities.

Case 1. If $p^+ - min(p^+, p^-) > 0$. This is equivalent to $p^+ > p^-$. Then, the dimension of the null space is greater or equal to $p^+ - p^-$. Said differently, if more degrees of freedom between the feet and the ground are locked after impact than before, then there exist velocities that lead to a total dissipation of the energy.

Case 2. If $p^+ - min(p^+, p^-) \leq 0$, or equivalently $p^+ \leq p^-$, (4.50) does not allow to conclude[3].

4.2.6 Main result

In this section we proved the following result:

Let a model of biped robot be such as described in Chapter 1 and be subject to p^- holonomic constraints $\eta^-(q_e) = 0$. Let $q_e \in \mathcal{Q}_e^{\eta^-} \subset \mathbb{R}^N$ be a set of generalized positions, and $\dot{q}_e \in \mathcal{T}_{q_e}\mathcal{Q}_e^{\eta^-}$ be the generalized velocities. The foot-ground contact during and after the impact is modeled with p^+ holonomic constraints $\eta^+(q_e) = 0$. For any configuration $q_e \in \mathcal{Q}_e$, if $\frac{\partial \eta^+}{\partial q_e}(q_e)$ and $\frac{\partial \eta^-}{\partial q_e}(q_e)$ are full rank, and the hypotheses of the impact model are respected (see 1.4.2), then:

- $V^{\eta^+}_{0,\eta^-} = Ker(\frac{\partial \eta^+}{\partial q_e}(q_e)) \cap Ker(\frac{\partial \eta^-}{\partial q_e}(q_e))$ is the set of velocities $\dot{q}_e$ that dissipate no energy at impact.
- $V^{\eta^+}_{-1,\eta^-} = Im\Big(D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\Big) \cap Ker(\frac{\partial \eta^-}{\partial q_e}(q_e))$, is the set of velocities $\dot{q}_e$ that dissipate all the energy at impact.

[3]The case where $N - p^- = 0$ has no interest, since it means that all the degree of freedom of the robot are locked before impact.

Furthermore

$$N - p^- - min(p^+, N - p^-) \leq dim(V^{\eta^+}_{0,\eta^-}) \leq N - p^+ . \tag{4.51}$$

$$p^+ - min(p^+, p^-) \leq dim(V^{\eta^+}_{-1,\eta^-}) \leq N - p^- . \tag{4.52}$$

Hence

- $N - p^- - p^+ > 0$ is a sufficient condition for the existence of velocities that dissipate no energy at impact.
- $p^+ > p^-$, that is more degrees of freedom are locked by the foot-ground contact after impact than before, is a sufficient condition for the existence of velocities that dissipate all the energy at impact.

Answering the question of the existence of such velocities for the other cases is still an open problem.

Note that Miossec and Aoustin [74] designed walking trajectories with no dissipation of energy at impact for a planar five-link walker. For that purpose, they imposed the swing foot to land on the ground at zero velocity. This is exactly the same as imposing the velocity $\dot{q}_e$ to be in $\mathrm{V}^{\eta^+}_{0,\eta^-}$ at impact.

4.3 Using the Relative Energy Dissipation for Motion Planning

In this section, we consider the problem of designing motions bringing the robot at impact with a given relative energy dissipation at impact. We see two practical interests of designing such motions. A walking cycle with a small relative energy dissipation is likely to result in less noise and vibration at impact in the structure. A motion with a high relative energy dissipation can be used to stop the robot in "one step". The theoretical framework introduced above will be useful to facilitate the design of such motions.

We use virtual constraints for parameterizing the motion (see Section 2.4.3). Indeed, the use of virtual constraints has some interests for the problem that we consider. As seen in Section 2.4.5.2, virtual constraints define a unique path in the configuration space for biped robots with maximally one degree of underactuation. In such a case, we will see that the relative energy dissipation at impact is independent of the travel speed of this path. This facilitates both the motion planning algorithm, and the control of the energy dissipation at impact in practice.

4.3.1 Virtual Constraints Define a Unique Relative Energy Dissipation

Let us consider the case of biped robots whose joints are independently actuated (no passive joints), and such that the foot-ground contact has maximally one

degree of freedom. In others words, the degree of underactuation does not exceed one. A biped robot in single support phase with the stance foot flat on the ground (no underactuation), or a planar biped robot with point feet in single support phase (one degree of underactuation) are some examples.

Let θ be a phase variable[4], that is a geometrical quantity used to parameterize the motion. Let the virtual constraints for the actuated coordinates q_a be

$$y = q_a - h_d(\theta)\,, \tag{4.53}$$

where $\theta \to h_d(\theta)$ is the motion of reference of the actuated coordinates. Assuming that the output y is zero, and since the degree of underactuation does not exceed one, then the generalized positions q are only a function of θ

$$q = f(\theta)\,, \tag{4.54}$$

and

$$\dot{q} = f^{'}(\theta)\dot{\theta}\,. \tag{4.55}$$

From (4.8), the relative energy dissipation is

$$\frac{f^{'}(\theta)^T A\Big(f(\theta)\Big)f^{'}(\theta)}{f^{'}(\theta)^T D\Big(f(\theta)\Big)f^{'}(\theta)}\,. \tag{4.56}$$

This quantity is independent of the velocity $\dot{\theta}$. Therefore, the energy dissipation depends only on the position along the path in the configuration space parameterized by θ. This proves that correctly following the virtual constraints ensures a given relative energy dissipation at impact, if the impact occurs for the same value of $\theta = \theta^-$, as for the case of a walk on flat ground. Note that this is not true anymore for several degree underactuated robots, since the path defined by the virtual constraints is not unique.

4.3.2 Imposing a Relative Energy Dissipation in Virtual Constraints

Using (4.56), one can easily shape the virtual constraints to have a desired relative energy dissipation at impact if the robot has maximally one degree of underactuation. Let us consider that the virtual constraints bring the robot at a given configuration at impact q^-. The phase variable is equal to $\theta = \theta^-$. Let r be the desired relative energy dissipation at impact. From (4.16), r must verify $\lambda_1 \leq r \leq \lambda_{N-p^-}$. From (4.56), imposing r is equivalent to impose $f'(\theta^-)$.

4.3.2.1 Computing $f'(\theta^-)$ to Get the Desired Relative Energy Dissipation at Impact r

We propose to proceed the following way by considering 3 cases.

[4]If the robot is fully actuated, the ordinary time t can also be used. The real interest of using a phase variable is when the robot is underactuated.

Case 1. If $r = \lambda_1$ or $r = \lambda_{N-p^-}$, then $f'(\theta^-)$ is taken as a linear combination of the corresponding generalized eigenvector(s) of A and D.

Case 2. If $\lambda_1 < r < \lambda_{N-p^-}$, and $r \neq \lambda_i, i \in \{1, N-p^-\}$, the velocity $f'(\theta^-)$ is chosen in the vector subspace

$$\text{Vect}(v_d) \oplus \text{Vect}(v_{d+1}) ... \oplus \text{Vect}(v_{N-p^-}) \,, \tag{4.57}$$

where $\lambda_d < r < \lambda_{d+1}$. Hence

$$f'(\theta^-) = \alpha_d v_d + \alpha_{d+1} v_{d+1} + ... + \alpha_{N-p^-} v_{N-p^-} \,, \alpha_i \in \mathbb{R},\, i \in \{d, N-p^-\} \,. \tag{4.58}$$

Since the eigenvectors are orthogonal to each other (see (4.17)), the relative energy dissipation is

$$\frac{f'(\theta^-)^T A\Big(f(\theta^-)\Big) f'(\theta^-)}{f'(\theta^-)^T D\Big(f(\theta^-)\Big) f'(\theta^-)} = \frac{\sum\limits_{i=d}^{N-p^-} \alpha_i^2 \lambda_i}{\sum\limits_{i=d}^{N-p^-} \alpha_i^2} \,. \tag{4.59}$$

Leaving free $\alpha_{i, i\in\{d+1, N-p^-\}}$, so that the relative energy dissipation be equal to r, α_d must verify

$$\alpha_d^2 = \frac{\sum\limits_{i=d+1}^{N-p^-} \alpha_i^2 \Big(\lambda_i - r\Big)}{r - \lambda_d} \,. \tag{4.60}$$

This equation has necessarily one solution, since the right-hand side of the equation is positive by construction.

Case 3. If $\lambda_1 < r < \lambda_{N-p^-}$, and there exits $d \in \{2, N-p^-1\}$ such that $r = \lambda_d$, the velocity $f'(\theta^-)$ is chosen as a combination of the corresponding eigenvectors, and of the eigenvectors associated to the eigenvalue 0 (if they exist).

Remarks: The orientation of $f'(\theta^-)$ does not modify the relative energy dissipation. This sign is chosen to be in accordance with the conditions under which an impact occurs. For instance, the vertical velocity of the swing leg must be downward if it is assumed that the impact occurs when the swing leg hits the ground.

There exist other ways to construct $f'(\theta^-)$, but proceeding like this was enough for the numerical example of the five-link waker given in the next section.

4.3.2.2 Shaping the Virtual Constraints such that $\dot{q}^- = f'(\theta^-)\dot{\theta}^-$

The motion of reference $h_d(\theta)$ must verify

$$(h_d^{'}(\theta^-); 1) = f^{'}(\theta^-)\,, \tag{4.61}$$

for one degree underactuated robots or

$$h_d^{'}(\theta^-) = f^{'}(\theta^-)\,, \tag{4.62}$$

for fully actuated robots.

4.3.3 Motion Planning Algorithm

We use the HZD method to design virtual constraints bringing the robot at impact with a desired relative energy dissipation and respecting some constraints (friction cone constants, torque limits,...). We solve the corresponding parametric nonlinear optimization problem (see Section 2.4.6.1) that we adapt for our problem, especially by using the results of Section 4.3.2.

With this approach, the motion of reference $h_d(\theta)$ is parameterized by polynomials. Then, from (4.61) or (4.62), one can derive explicit relations which must be verified by the coefficients of the polynomials, reducing the size of the search space of the optimization. Hence, the relative energy dissipation can be directly imposed by a proper choice of the polynomial coefficients.

This method is applied in the next section for the particular case of a planar five-link walker. Further details can be found in Appendix F.

4.4 Application to a Planar Five-Link Walker

As an illustration, we apply the previous study realized in Section 4.2 and 4.3 to a planar five-link walker with point feet. First, we present this model. Then, we perform a numerical exploration on the configuration space of the robot. For each position explored, the generalized eigenvalue problem is solved, and the results are compared to the theoretical result of Section 4.2.6. We especially investigate in what extend the hypotheses of the impact model are verified, as it was assumed in these theorems. Next, using Section 4.3, we generate motions bringing the robot with a low or high relative energy dissipation at impact, and discuss the practical interest of such motions.

4.4.1 Model of the Planar Five-Link Walker

The planar five-link walker with point feet has seven degrees of freedom. It is made of two articulated legs and of one torso. A description of the model can be found in [127].

The robot has four independently actuated joint angles $q_b = (q_1; q_2; q_3; q_4)$. Its absolute position is parameterized by the Cartesian position of the foot of one leg, called leg 1. This position is denoted $p_1 = (p_1^x; p_1^z)$. The Cartesian position

of the other leg, called leg 2, is $p_2 = (p_2^x; p_2^z)$. The absolute orientation of the robot is parameterized by q_5. Then, a set of (extended) coordinates for the robot $q_e \in \mathcal{Q}_e \subset \mathbb{R}^7$ is

$$q_e = (q_1; q_2; q_3; q_4; q_5; p_1^x; p_1^z)\,. \tag{4.63}$$

The robot in single support phase is depicted figure 4.1. In single support phase, the stance leg is assumed to be pinned onto the ground. The foot-ground contact is unactuated and is modeled as an ideal revolute joint. Then, the robot is subject to the two-dimensional holonomic constraints

$$\eta_i(q_e) = \begin{pmatrix} p_i^x(q_e) - p_i^{x*} \\ p_i^z(q_e) - p_i^{z*} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}\,, \tag{4.64}$$

where $i = 1$, if the stance foot is the leg 1, or $i = 2$, if the stance foot is the leg 2, p_i^{x*} and p_i^{z*} are real constants.

Hence, the absolute orientation of the robot q_5 is unactuated and the robot has one degree of underactuation. A set of independent coordinates (reduced coordinates) is

$$q = (q_1; q_2; q_3; q_4; q_5) \in \mathcal{Q}_f^{\eta_i} \subset \mathbb{R}^5\,. \tag{4.65}$$

The equations of motion were generated using the method of Lagrange. The Lagrange's equations were derived using the MATLAB's symbolic toolbox. The dynamic parameters of the model are given in figure 4.2. Note that they are the same as for the robot VS, except that, for simplicity, we do not consider the 2 bars stabilizing mechanism.

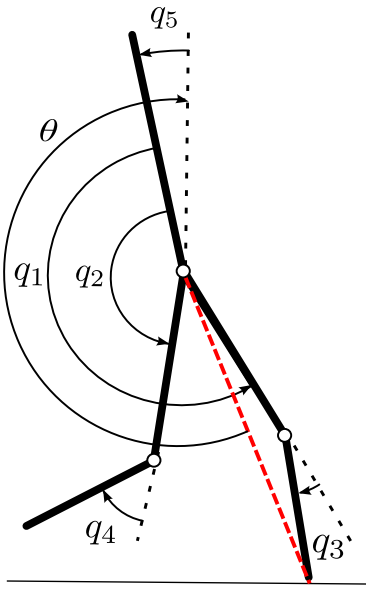
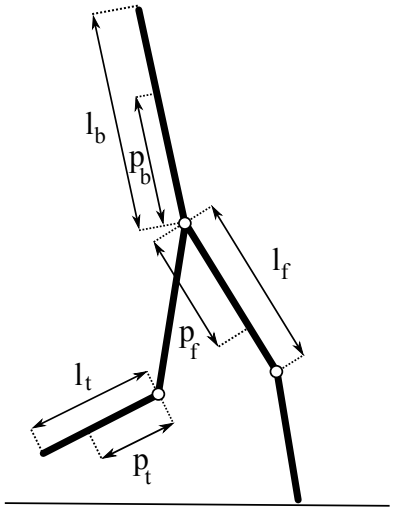


Figure 4.1: The planar five-link walker with measurements conventions. The red dotted line is called the virtual stance leg.



(a) The planar five-link walker.

Torso	l_b	820 mm
	p_b	95 mm
	M_b	15.43 kg
	I_b^{com}	0.8567 kg.m^2
Thigh	l_f	400 mm
	p_f	201 mm
	M_f	5.42 kg
	I_f^{com}	0.1230 kg.m^2
Shin	l_t	400 mm
	p_t	215 mm
	M_t	2.25 kg
	I_t^{com}	0.029 kg.m^2
Transmission ratio	r_{knee}	101
	r_{hip}	100
Transmission inertia	I_{knee}	1.89 kg.m^2
	I_{hip}	1.46 kg.m^2

(b) Dynamic parameters of the planar five-link walker. p_i gives the position of the center of mass of the link i. I_i^{com} is the inertia of the link i expressed at the center of mass of the link i.

Figure 4.2: Dynamic parameters and notations used for the planar five-link walker.

4.4.2 Validity of the Impact Model

We consider that before the impact, the robot is either in flight phase or in single support phase. The impact occurs when at least one leg hits the ground. When and after impacting the ground, the robot can be either in single support phase or in double support phase. The determination of the phase after impact is made by checking if the impulsive contact wrench I and the behaviour of the former stance leg are compatible with the new phase. See 1.4.2 for further details.

We denote I_i^x (resp. I_i^z) the horizontal (resp. vertical) ground reaction impulse exerted on the leg i, and μ the Coulomb static friction. From [55, 78, 127], the hypotheses to verify for the validity of the impact model are the following ones.

4.4.2.1 Single Support Phase After Impact

The robot is in single support phase after impact if the leg that impacts the ground does not take off (a), does not slip (b), and the former stance leg lifts the ground without interaction (c). Without loss of generality, we assume that the leg impacting the ground is leg 2. Then, we have

$$\begin{cases} I_2^z > 0 & (a) \\ \left|\dfrac{I_2^x}{I_2^z}\right| < \mu & (b) \\ \dfrac{\partial p_1^z}{\partial q_e}(q_e)\dot{q}_e^+ \geq 0 & (c) \end{cases} \tag{4.66}$$

4.4.2.2 Double Support Phase After Impact

The robot is in double support phase after impact if both legs do not take off (d), (f), and do not slide (e), (g)

$$\begin{cases} I_1^z > 0 & (d) \\ \left|\dfrac{I_1^x}{I_1^z}\right| < \mu & (e) \\ I_2^z > 0 & (f) \\ \left|\dfrac{I_2^x}{I_2^z}\right| < \mu & (g) \end{cases} \tag{4.67}$$

4.4.2.3 Checking the Hypotheses of the Impact Model for the Generalized Eigenvalues

Since I is linear in the velocity $\dot{q}_e$ (see (1.61)), then the no slipping conditions (b), (e), and (g) do not depend on the norm of $\dot{q}_e$, but only of the direction of $\dot{q}_e$. Likewise, the no take off conditions (a), (d), and (f) depend only on the orientation of the velocity $\dot{q}_e$. Then, it is sufficient to check the hypotheses on the normalized eigenvector obtained from v_d, and whose orientation is chosen such that the swing

leg velocity has a decreasing altitude at impact. If this vector verifies the impact model hypotheses, then it is the case for kv_d, $k \in \mathbb{R}^{+*}$.

We remind that in the particular case where $\frac{\partial \eta^+}{\partial q_e}(q_e)\dot{q}_e^- = 0$, as it is the case for the generalized eigenvectors associated to the eigenvalue 0, there is no impact. Then, there is no need to check the impact model hypotheses.

4.4.3 Numerical exploration in the Configuration Space at Impact

To check numerically the theoretical results of Section 4.2.6, we perform here a numerical exploration in the set of the configurations of the robot at impact, and assuming a flat ground. As we saw, before impact, the robot may be either in flight phase, or in single support phase. After impact, it may be either in single support phase or in double support phase. Then, we consider four cases:

- Transition from a flight phase to a single support phase.
- Transition from a flight phase to a double support phase.
- Transition from a single support phase to a single support phase.
- Transition from a single support phase to a double support phase.

For each of these cases, we discretize the set of the configurations at impact. See the algorithm 1.For each point of the discretized space, the generalized eigenvalue problem (4.14) is solved. In particular, we investigate if the hypotheses of the impact model (4.66) and (4.67) are not violated for the generalized eigenvectors. Indeed, as explained before, the validity of them were not taken into account in our analytical study, since they are model-dependent and cannot be guaranteed a priori to be true. As explained in 4.4.2.3, it is enough to check the hypotheses on the normalized eigenvectors. The orientation of the eigenvectors is selected such that the leg(s) that impact the ground have a decreasing altitude (otherwise the leg(s) do(es) not impact the ground).

We ignore some configurations that we consider as not "anthropomorphic" (absolute torso angle q_5 is greater than $\frac{\pi}{4}$ in absolute value, knee under the ground, step length inferior to 0.1 m) and such that the swing leg is behind the stance leg at impact[5]. Note that the problem is translational invariant, i.e. independent of the coordinate p_1^x.

4.4.3.1 Flight Phase Before Impact

We assume here that prior to the impact, the robot is in flight phase, i.e is not in contact with the ground ($p^- = 0$). Then, the configuration space of the model before impact $\mathcal{Q}_e^{\eta^-} = \mathcal{Q}_e$ is a 7-dimensional manifold. The resolution of the generalized eigenvalue problem gives seven generalized eigenvectors

$$-1 \leq \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \lambda_4 \leq \lambda_5 \leq \lambda_6 \leq \lambda_7 \leq 0\,. \tag{4.68}$$

[5]The robot is assumed to move from the left to the right

for each configuration q of the grid **do**
  if the configuration is "anthropomorphic" **then**
    Compute the generalized eigenspaces of $A(q)$ and $D(q)$
    for each generalized eigenvector v_d **do**
      $v_d \leftarrow -v_d$ if the corresponding swing foot altitude is increasing.
      if (4.66) is true and (4.67) is false **then**
        The robot is in single support phase after impact when $x^- = (q; v_d)$.
        The corresponding relative energy dissipation is the generalized eigenvalue λ associated to v_d.
      end if
    end for
    Store the results
  end if
end for

Algorithm 1: Numerical exploration in the configuration space at impact of the relative energy dissipation assuming a transition from flight phase to single support phase.

Single support phase after impact We consider here that the robot is in single support phase after impact. We have

$$\begin{cases} N = 7 \\ p^- = 0 \\ p^+ = 2 \end{cases} \tag{4.69}$$

Without loss of generality, we assume that an impact occurs when leg 2 hits the ground. We remind (see Section 4.4.1) that the Cartesian position of leg 2 is denoted $p_2 = (p_2^x; p_2^z)$. Then, the configurations at impact are

$$\mathcal{D}_A = \left\{ q_e \in \mathcal{Q}_e \,|\, p_2^z(q_e) = 0 \right\}. \tag{4.70}$$

$\mathcal{D}_A$ is discretized, considering the translational invariance of the problem: 10790 "anthropomorphic" configurations are investigated.

After impact, the system is subject to two holonomic constraints

$$\eta^+(q_e) = \begin{pmatrix} p_2^x(q_e) - p_2^{x*} \\ p_2^z(q_e) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{4.71}$$

where p_2^{x*} is a real constant. $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank[6] on $\mathcal{Q}_e$.

According to the theoretical result of Section 4.2.6, the set of velocities that dissipate no energy at impact (generalized eigenvalue equal to 0) is of dimension

[6] We have $p_2(q_e) = p_1 + f_{p_2}(q)$. Then, $\frac{\partial \eta^+}{\partial q_e}(q_e) = \begin{pmatrix} \frac{\partial f_2}{\partial q}(q) & I_{2\times 2} \end{pmatrix}$. Hence, the rank of the $(2 \times N)$ jacobian matrix is two.

$N - p^+ = 7 - 2 = 5$, and the set of velocities that dissipate all the energy (generalized eigenvalue equal to -1) is of dimension $p^+ = 2$. This is in agreement with the obtained numerical results: for each configuration q explored, we numerically found five generalized eigenvalues equal to 0 and two equal to -1.

Over the 26914 configurations investigated, 4656 configurations have at least one eigenvector associated to the eigenvalue -1 and respecting the hypotheses (4.66) of the impact model. Some of these configurations are depicted figure 4.3.

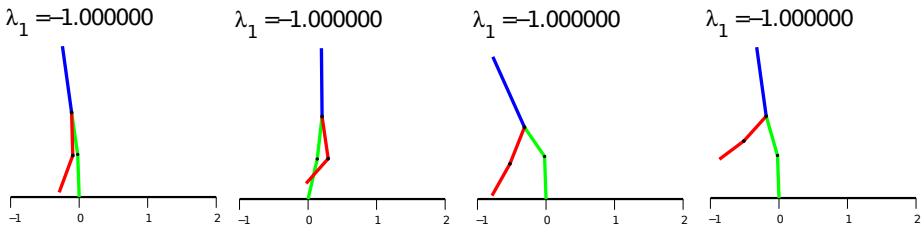


Figure 4.3: Transition from a flight phase to a single support phase. Some configurations with at least one eigenvector associated to the eigenvalue -1 (total dissipation of the energy at impact) that respect the hypotheses of the impact model.

Double support phase after impact The robot is assumed to be in double support phase after impact. We have

$$\begin{cases} N = 7 \\ p^- = 0 \\ p^+ = 4 \end{cases} \tag{4.72}$$

The set of configurations at impact is

$$\mathcal{D}_B = \left\{ q_e \in \mathcal{Q}_e \,|\, p_1^z(q_e) = 0,\, p_2^z(q_e) = 0 \right\}. \tag{4.73}$$

$\mathcal{D}_B$ is discretized, considering the translational invariance of the problem: 33393 "anthropomorphic" configurations are explored.

After impact, the system is subject to four holonomic constraints

$$\eta^+(q_e) = \begin{pmatrix} p_1^x(q_e) - p_1^{x*} \\ p_1^z(q_e) \\ p_2^x(q_e) - p_2^{x*} \\ p_2^z(q_e) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{4.74}$$

where p_1^{x*} and p_2^{x*} are real constant values. $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank[7], except when

[7] Since $p_2(q_e) = p_1 + f_{p_2}(q)$, then $\frac{\partial \eta^+}{\partial q_e}(q_e) = \begin{pmatrix} 0_{2\times 5} & I_{2\times 2} \\ \frac{\partial f_2}{\partial q}(q) & I_{2\times 2} \end{pmatrix}$. Hence, $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank if and only if the rank of $\frac{\partial f_2}{\partial q}(q)$ is two. It was checked that this is the case on $\mathcal{Q}_f^{\eta^-}$, except when $q_1 = q_2 = q_3 = q_4 = q_5 = 0$.

$q_1 = q_2 = q_3 = q_4 = q_5 = 0$.

The theoretical result of Section 4.2.6 says that the set of velocities that dissipate no energy at impact ($\lambda = 0$) is of dimension $N - p^+ = 7 - 4 = 3$, and that the set of velocities that dissipate all the energy ($\lambda = 1$) is of dimension $p^+ = 4$. This is in agreement with the obtained numerical results: for each configuration q explored, we numerically found three generalized eigenvalues equal to 0 and four equal to -1.

Over the 33393 configurations, 4123 configurations have at least one eigenvector associated to the eigenvalue -1, and respecting the hypotheses of the impact model. Some of these configurations are depicted figure 4.4.

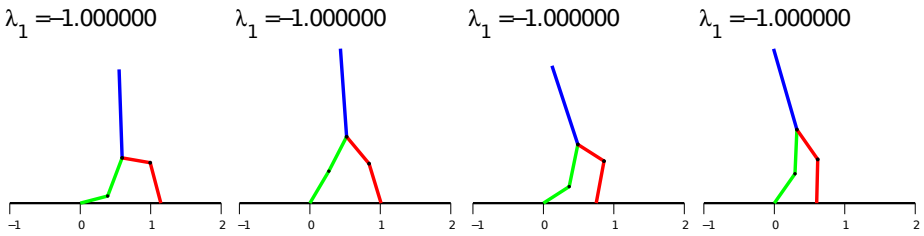


Figure 4.4: Transition from a flight phase to a double support phase. Some configurations with at least one eigenvector associated to the eigenvalue -1 (total dissipation of the energy at impact) that respect the hypotheses of the impact model (4.67).

4.4.3.2 Single Support Phase Before Impact

Before impact, the robot is now assumed to be in single support phase. Without loss of generality, we assume that leg 1 is the stance leg. Then, the robot before impact is subject to two holonomic constraints

$$\eta^-(q_e) = \begin{pmatrix} p_1^x - p_1^{x*} \\ p_1^z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} , \tag{4.75}$$

where p_1^{x*} is a real constant.

The configuration manifold of the model before impact is $\mathcal{Q}_e^{\eta^-}$, a five-dimensional manifold. We use the reduced coordinates $q = (q_1; q_2; q_3; q_4; q_5) \in \mathcal{Q}_f^{\eta^-} \subset \mathbb{R}^5$.

To define the mapping between the extended coordinates and the reduced coordinates, an embedding is defined (see 1.3.4). The embedding E^{η^-}, mapping an element $q \in \mathcal{Q}_f^{\eta^-}$ to an element $q_e \in \mathcal{Q}_e^{\eta^-}$, is trivial, and defined as follows

$$q_e = E^{\eta^-}(q) = \begin{pmatrix} q \\ p_1^{x*} \\ 0 \end{pmatrix} . \tag{4.76}$$

The resolution of the generalized eigenvalue problem (4.14) gives five generalized eigenvectors

$$-1 \leq \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \lambda_4 \leq \lambda_5 \leq 0\,. \tag{4.77}$$

An impact occurs when leg 2 hits the ground. Then, the set of configurations at impact is[8]

$$\mathcal{D}_C = \left\{ q \in \mathcal{Q}_f^{\eta^-} \mid p_1^z\Big(E^{\eta^-}(q)\Big) = 0,\, p_2^z\Big(E^{\eta^-}(q)\Big) = 0 \right\}. \tag{4.78}$$

The number of "anthropomorphic" configurations explored on this discretized domain, and considering the translational invariance of the problem, is equal to 33 393.

Single support phase after impact After impact, the robot is assumed to be in single support phase with leg 2 pinned on the ground. Thus, after impact, the system is subject to the two holonomic constraints

$$\eta^+(q_e) = \begin{pmatrix} p_2^x(q_e) - p_2^{x*} \\ p_2^z(q_e) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{4.79}$$

where p_2^{x*} is a real constant. For the same reason as before, $\eta^+(q_e)$ is full rank on $\mathcal{Q}_e$.

We have

$$\begin{cases} N = 7 \\ p^- = 2 \\ p^+ = 2 \end{cases} \tag{4.80}$$

Hence, from the theoretical result of 4.2.6, we know that for a given configuration $q \in \mathcal{Q}_f^{\eta^-}$, the subspace of $\mathcal{T}_q\mathcal{Q}_f^{\eta^-}$ that corresponds to no dissipation of energy at impact is a vector space whose dimension is comprised between $N - p^- - p^+ = 7 - 2 - 2 = 3$ and $N - p^+ = 7 - 2 = 5$. Numerically, we found three eigenvalues equal to zero for each configuration of the grid, i.e. $\lambda_3 = \lambda_4 = \lambda_5 = 0$. We checked that the associated eigenvectors respectively belong to $Ker\left(\frac{\partial \eta^+}{\partial q_e}\Big(E^{\eta^-}(q)\Big)\frac{\partial E^{\eta^-}}{\partial q}(q)\right)$ and $Ker\left(\frac{\partial \eta^-}{\partial q_e}(q_e)D_e^{-1}(q_e)\frac{\partial \eta^{+T}}{\partial q_e}(q_e)\right)$.

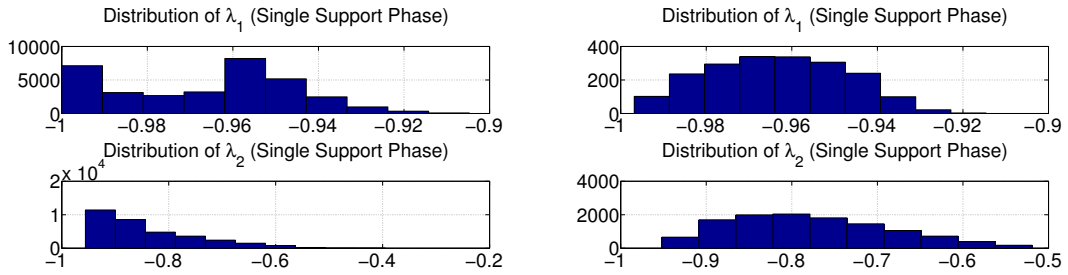
Let us focus now on the two others eigenvalues λ_1 and λ_2. Since $p^+ = p^-$, the theoretical result of Section 4.2.6 says nothing on the existence of velocities that dissipate all the energy at impact (eignevalue equal to -1). From our numerical study, see figure 4.5, we see that such velocities do not exist, even though the lowest eigenvalue λ_1 is sometimes very close to -1. Figure 4.6 depicts some configurations whose eigenvectors associated to λ_1 respect the hypotheses of the impact model.

Figure 4.7 depicts the relationship between the step length and λ_1, for the configurations whose eigenvectors associated to λ_1 respect impact hypothesis. We remark that the configurations offering the lowest relative energy dissipation, i.e.

[8] Actually, $\mathcal{D}_B$ and $\mathcal{D}_C$ are the same set. But, since we use two different sets of coordinates to describe them, we give them a different name.

the closest to -1, correspond to large step lengths.

Finally, it is important to remark that among the 33 393 configurations explored, only 1982 have the eigenvector associated to the lowest eigenvalue λ_1 that respects the hypotheses of the impact model. And 12 013 configurations have the eigenvector associated to the second lowest eigenvalue λ_2 that respect them (see figure 4.5). This illustrates the necessity of checking these hypotheses, especially when the relative energy dissipation is close to -1.



(a) Considering all the configurations.

(b) Considering only the configurations that respect the hypotheses of the impact model.

Figure 4.5: Transition from a single support phase to a single support phase. Distribution of the lowest eigenvalue λ_1 and λ_2 assuming the robot is in single support phase after impact.

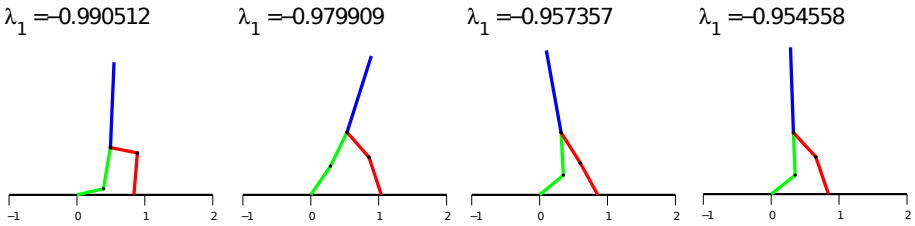


Figure 4.6: Transition from a single support phase to a single support phase. Some configurations such that the eigenvector associated to the lowest eignevalue λ_1 respect the hypotheses of the impact model (4.66).

Double support phase after impact The robot is assumed to be in double support phase after impact. Thus, after impact, the system is subject to four holonomic constraints

$$\eta^+(q_e) = \begin{pmatrix} p_1^x(q_e) - p_1^{x*} \\ p_1^z(q_e) \\ p_2^x(q_e) - p_2^{x*} \\ p_2^z(q_e) \end{pmatrix}, \tag{4.81}$$

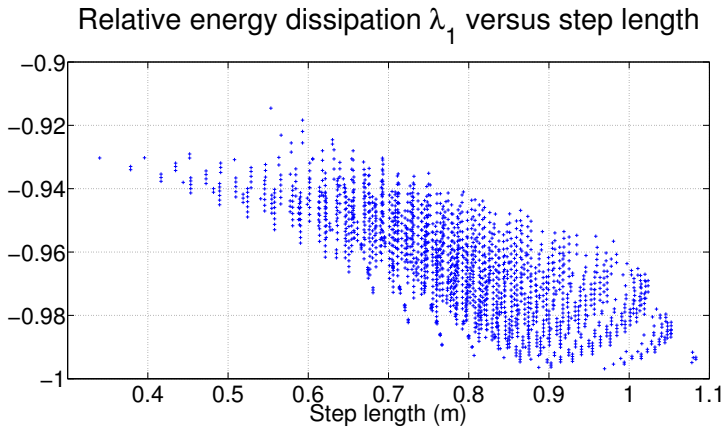


Figure 4.7: Transition from a single support phase to a single support phase. Relationship between the step length and the relative energy dissipation λ_1. Only the configurations whose eigenvectors associated to λ_1, and respecting the hypotheses of the impact model (4.66) are considered.

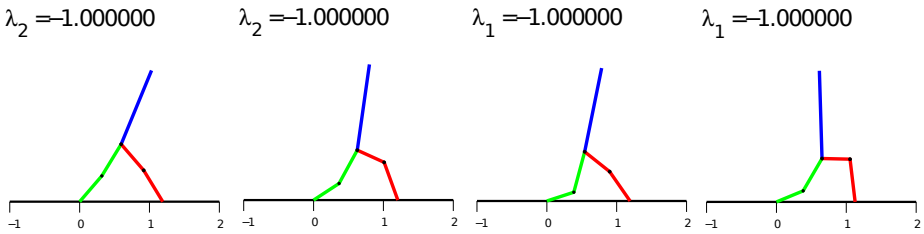


Figure 4.8: Transition from a single support phase to a double support phase. Some configurations such that the eigenvector associated to the eigenvalue -1 (total dissipation of energy at impact) respect the hypotheses of the impact model (4.67). All of them correspond to large step lengths (between 1.1 and 1.4 m).

where p_2^{x*} are p_2^{x*} are real constants.

For the same reasons as before, $\frac{\partial \eta^+}{\partial q_e}(q_e)$ is full rank, except when $q_1 = q_2 = q_3 = q_4 = q_5$. We have

$$\begin{cases} N = 7 \\ p^- = 2 \\ p^+ = 4 \end{cases} \tag{4.82}$$

From the theoretical result of Section 4.2.6, we know that, for a given configuration $q \in \mathcal{Q}_f^{\eta^-}$, the subspace of $\mathcal{T}_q \mathcal{Q}_f^{\eta^-}$ that corresponds to no dissipation of energy at impact is a vector space whose dimension is comprised between $N - p^- - p^+ = 7 - 2 - 4 = 1$ and $N - p^+ = 7 - 4 = 3$. Numerically, we find three eigenvalues equal to zero for each configuration of the grid, i.e. $\lambda_3 = \lambda_4 = \lambda_5 = 0$.

Since $p^+ > p^-$, i.e the number of degrees of freedom locked by the foot-ground contact is greater for the model after impact than for the model before impact,

then there exist velocities that dissipate all the energy at impact. The vector space of such velocities has a dimension comprised between $p^+ - p^- = 4 - 2 = 2$, and $N - p^- = 7 - 2 = 5$. We numerically observed that it is of dimension two for any configuration at impact, i.e. $\lambda_1 = \lambda_2 = -1$. On the 33 393 configurations explored, only 93 have at least one eigenvector associated to the eigenvalue -1 that respects the hypotheses of the impact model. Some of them are depicted figure 4.8. All of them correspond to large step lengths (between 1.1 and 1.4 m). As previously, this illustrates the difficulty of not slipping and not taking off when the dissipation of energy is high.

4.4.3.3 Discussion

The numerical study is in accordance with the theoretical result of Section 4.2.6. Since these theorems do not say nothing about the fact that the constraints imposed by the foot-ground contact are satisfied, an extra check of the hypotheses of the impact model has to be down.

The study especially showed that many states with a relative energy dissipation close to -1 do not respect some of these hypotheses. As a matter of fact, it seems intuitive that the higher the energy relative dissipation is, the more likely it is that the foot impacting the ground slips or takes off.

The configurations at impact with a relative energy dissipation close to -1 and respecting the hypotheses of the impact model generally correspond to a large step.

4.4.4 Controlling the Energy Dissipation at Impact

We design here motions for the planar five-link walker with a desired relative energy dissipation at impact. For that purpose, we design virtual constraints using the motion planning algorithm described in Section 4.3.3. The standard phase variable θ corresponding to the virtual stance leg angle, and depicted in figure 4.1, is used to parameterize the motions.

First, we generate walking trajectories with a small relative energy dissipation at impact. As explained in the introduction of this chapter, such trajectories minimize a priori vibrations in the structure of the robot and the noise. This is an important point, especially for preserving the hardware or for the comfort of an exoskeleton user. Next, we design motions with a high relative energy dissipation at impact. We show that they can be used to stop the robot in "one step".

4.4.4.1 Designing Walking Trajectories with Little Energy Loss at Impact

We address here the problem of designing walking trajectories with little energy loss at impact. We are not the first to address this problem. As explained previously, Miossec and Aoustin [74] designed walking trajectories with no dissipation of energy at impact for a planar five-link walker by imposing the swing foot to land on the ground at zero velocity. This is exactly the same as imposing the velocity $\dot{q}_e$

to be in $Ker(\frac{\partial \eta^+}{\partial q_e}(q_e))$ at impact (see the theoretical result of Section 4.2.6). The trajectory was composed of both a single and a non instantaneous double support phase.

Instead, we seek here motions with an instantaneous double support phase that dissipate little energy at impact, and not necessarily zero energy. Indeed, generating a motion with a non instantaneous double support phase require more efforts in simulation and in implementation, since one supplementary phase (the double support phase) has to be considered.

Walking trajectories with an instantaneous double support phase and no energy dissipation. We designed walking trajectories with an instantaneous double support phase and no energy dissipation at impact. Since the robot is in single support phase before impact, we saw that, for any configuration at impact, there are 3 eigenvectors λ_2, λ_3, λ_4 associated to the eigenvalue 0. In the motion planning algorithm, the velocity at impact is then chosen in $\text{Vect}(v_3, v_4, v_5)$.

The walking trajectories appeared to be weakly robust to joint tracking errors and to terrain variations. Especially, a high gain PD controller (see (2.55)) does not offer sufficient tracking accuracy to get a stable walk. Additionally, they require significative amount of torque. One such trajectory is depicted in figure 4.9 using an input-output linearizing controller (see Section 2.4.7) to track the virtual constraints. The properties of the trajectory are gathered in figure 4.10 (see trajectory 7). Note that due to numerical errors, the relative energy dissipation is not zero, but of the order of 1e-7.

[74] explained the high torque consumption by the fact that braking the swing foot to get a null velocity at impact requires power. They also showed the crucial role of the double support phase on the stability of walking trajectories with no energy dissipation. Hence, trajectories with an instantaneous double support phase and no energy dissipation do not seem to be feasible in practice. Then, we focus now on the design of trajectories with little, but non zero, energy dissipation at impact.

Walking trajectories with little energy dissipation at impact. We design here walking trajectories with little relative energy dissipation at impact. We noticed that, combined with constraints on the maximal joint velocities of the robot, imposing a relative energy dissipation close to zero results in trajectories with little energy dissipation.

The desired relative energy dissipation r is imposed as explained in 4.3.2. More specifically, we select the velocity before impact in the four-dimensional vector subspace generated by the eigenvectors associated to the four largest eigenvalues: $\text{Vect}(v_2, v_3, v_4, v_5)$. Indeed, we saw that three eigenvalues are 0, i.e. $\lambda_3 = \lambda_4 = \lambda_5 = 0$. Then, (4.60) gets

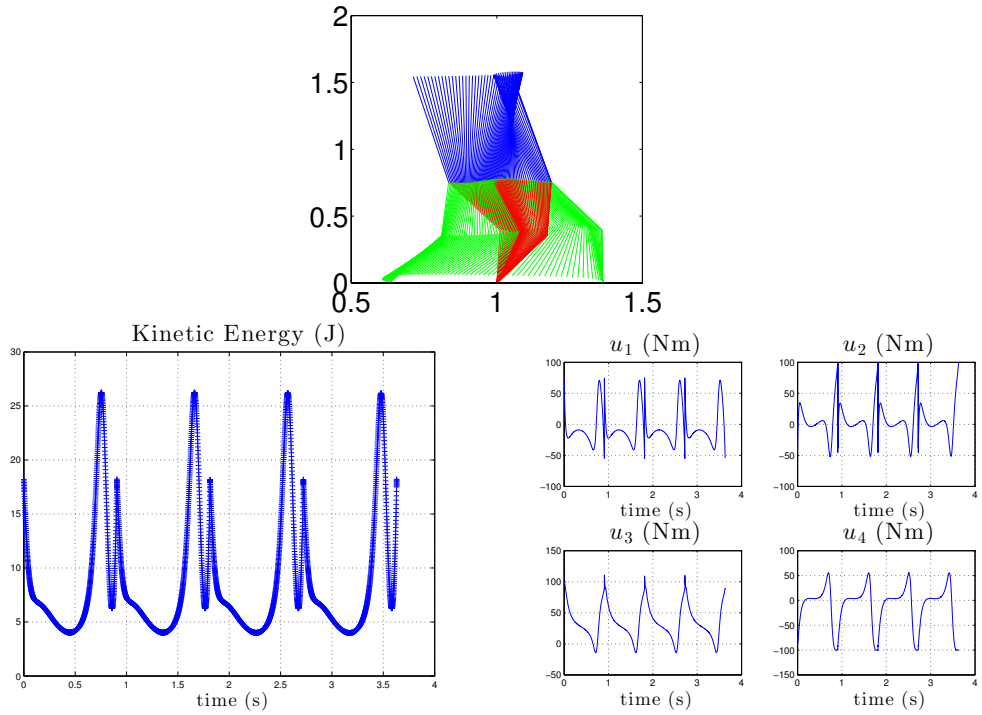


Figure 4.9: Stick animation and evolution of the kinetic energy and joint torques for a walking trajectory with an instantaneous double support phase and no energy dissipation at impact (trajectory 7 in figure 4.10) .

$$\alpha_2^2 = \frac{r \sum_{i=3}^{5} \alpha_i^2}{\lambda_2 - r}, \tag{4.83}$$

which always have solutions if $\lambda_2 < r$. Since we work with values of relative energy dissipation r close to zero, and that generally λ_2 is generally found to be smaller than -0.3, this condition is met.

The configuration of the robot at impact is fixed during the optimization. Indeed, we noticed that leaving it free results in poorer results. Several configurations at impact were tested. They were randomly chosen from anthropomorphic configurations. Further details on the motion planning algorithm can be found in F.1.

Let us compare now these trajectories to walking trajectories obtained using the standard HZD method, i.e. without constraining the energy losses at impact. See figure 4.10. The walking trajectories were obtained using different constraint thresholds on the step length and on the mean walking speed. First, we notice that the walking trajectories obtained using the standard HZD method have a relative energy dissipation far from 0. It is smaller than -0.4 for the most optimal[9]

[9]We recall that the sum of the square torques is the cost function of the optimization algorithm.

	Trajectories by not imposing the relative energy dissipation			
Trajectory label	0	1	2	3
Cost (Nm^2s/m)	14 768	3590	1818	1013
Relative Energy dissipation	-0.27	-0.62	-0.45	-0.53
Dissipated Energy (J)	6.6	29.6	10.4	17.3
Mean walking speed (m/s)	0.6	1	0.73	0.75
Step length (m)	0.22	0.52	0.41	0.48
Euclidian norm of swing foot velocity	0.87	2.73	1.17	1.94

(a) Walking trajectories obtained by not imposing a relative energy dissipation (standard HZD method).

	Trajectories by imposing the relative energy dissipation			
Trajectory label	4	5	6	7
Cost (Nm^2s/m)	20 270	7112	13 885	13 439
Relative Energy dissipation	-0.1	-0.15	-0.08	4e-7
Dissipated Energy (J)	3	3	1.7	1e-6
Mean walking speed (m/s)	1.12	0.73	0.79	0.40
Step length (m)	0.41	0.35	0.27	0.36
Euclidian norm of swing foot velocity	0.48	0.45	0.34	1.4e-3

(b) Walking trajectories obtained by imposing a relative energy dissipation close to zero.

Figure 4.10: Walking trajectories obtained by not imposing and by imposing a relative energy dissipation close to zero.

trajectories, meaning that about half of the kinetic energy of the robot is dissipated at impact. For these trajectories, impact losses increase with the mean walking speed.

On the contrary, trajectories designed by imposing a relative energy dissipation close to zero result in an important diminution of impact losses. See figure 4.11 that depicts comparative plots between trajectory 2 and trajectory 5. The counterpart is that the maximum torque is larger. It was noticed that the closer to zero r is, the higher is the maximum torque. Thus, low torque consumption and low impact losses seem to be contradictory, and a trade off has to be found. Not surprisingly, one remarks that impact losses are related to the magnitude of the swing foot velocity at impact. The higher the velocity is, the higher the losses are.

We checked that all these trajectories are stable in simulation using a high gain PD controller tracking the virtual constraints.

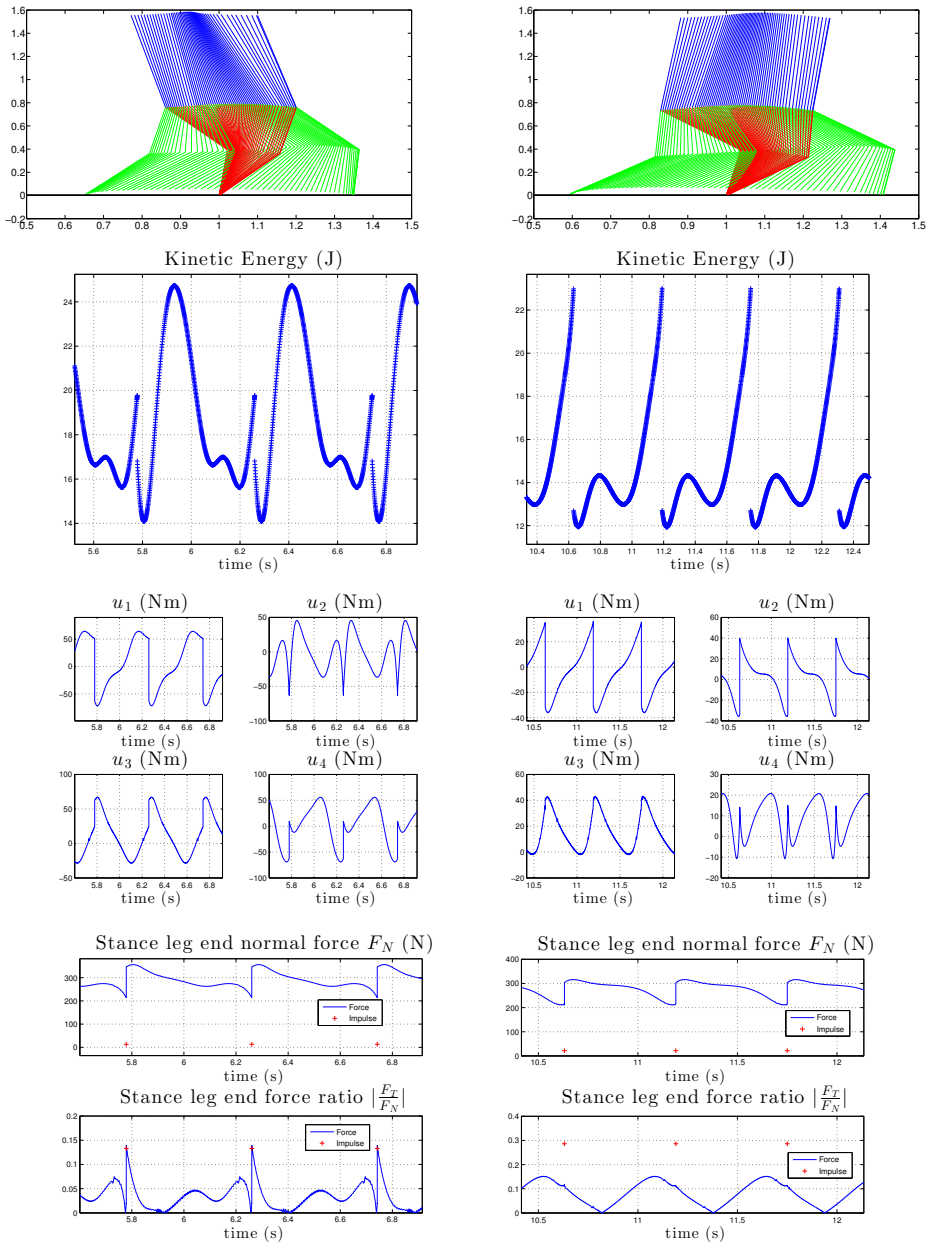


Figure 4.11: Stick animation and evolution of the kinetic energy, joint torques and ground reaction forces for two walk trajectories with the same mean walking speed.
Left : Trajectory 5 (obtained by imposing a relative energy dissipation equal to -0.1. Cost = 7812 Nm^2s/m).
Right : Trajectory 2 (obtained without constraints relative energy dissipation. Cost = 1818 Nm^2s/m).

4.4.4.2 Stopping the Robot in "One Step" by Dissipating its Kinetic Energy at Impact

We are now interested in the design of motions with a relative energy dissipation at impact close to -1. We show that such motions exist and can be used to stop the robot in "one step"[10]. This can be especially useful when the robot must stop in emergency.

As an illustration, we design a motion that starts from a walking cycle and bring the robot at impact with a high relative energy dissipation. To achieve a complete stop, the robot is then servoed to its standstill position (configuration at impact). The resulting back and forth rocking motion dissipates the remaining kinetic energy. We investigate a transition from a single support phase to a single support phase. Actually, the best would be to bring the robot in a state at impact that dissipates all its kinetic energy, i.e. with a relative energy dissipation equal to -1. But, reaching such states appeared to be too demanding in joint torques and velocities. Then, we scaled down our ambitions, and found that it was simpler to bring the robot at impact with a relative energy dissipation roughly equal to -0.8, and considering a single support phase after impact.

Important Features of the Motion Planning Algorithm. Details on the motion planning algorithm that we used can be found in Appendix F.2. Important features are that, contrary to the design of walking trajectories with little energy dissipation, the configuration at impact is varied during the optimization. We restrict the velocity before impact to be in the subvector space generated by the second lowest generalized eigenvalue, that is

$$\dot{q}^- \in \mathrm{Vect}(v_2). \tag{4.84}$$

Indeed, we found that when restricting the velocity to the subvector space generated by the lowest eigenvalue λ_1, i.e. for which the relative energy dissipation is the most important, it was difficult to find motions respecting torque constraints and the condition of no slipping. Then, we preferred working in $\mathrm{Vect}(v_2)$. This choice gave satisfying results, even though working in the plane $\mathrm{Vect}(v_1, v_2)$ may lead to richer results.

In addition to the usual constraints (friction cone constraints, torque limits...), we especially used the following equality constraints:

- The robot reaches the desired configuration at impact. ($\diamond$)
- The relative energy dissipation is equal to the desired one r. ($\diamond\diamond$)

And the following inequality constraint is used:

- $K^- \leq K^-_{Max}$, where K^- is the kinetic energy right before impact, and K^-_{Max} is the maximum kinetic energy allowed right before impact. ($\diamond\diamond\diamond$)

[10]We use quotation marks since the robot may slightly rock back and forth, or slip after impact.

($\diamond$) We found useful to add this constraint, since the robot tends to impact the ground before reaching the desired configuration at impact to diminish the torque consumption.

($\diamond\diamond$) Even though the value of λ_2 was found to be generally inferior to -0.5, it may take any value, depending on the configuration at impact. Then, we impose the algorithm to find a configuration such that $\lambda_2 = r$, where r is the desired relative energy dissipation. Since we want that a high amount of kinetic energy be dissipated, motions were designed with values of r comprised between -0.99 and -0.8.

($\diamond\diamond\diamond$) We noticed that the demand in torque and in velocity increases when r is decreased. Actually, when not constraining the kinetic energy before impact, it can reach up to 50 J, which is quite important (for comparison, the kinetic energy level at impact is roughly equal to 20 J for a walk). Then, we added a constraint on the maximum level of kinetic energy before impact to diminish the violence of the impact. Similarly, the demand in torque and in velocity increases when the level of maximum kinetic energy before impact is decreased. This is maybe due to the fact that braking the robot requires more energy.

The motion is started from a walking cycle obtained with the HZD method at the beginning of the step.

Results. Motions with a relative energy dissipation rate comprised between -0.8 and -0.9 were found. Fgure 4.12 illustrates the case where the relative energy dissipation is equal to -0.82. After impact, the robot slightly rocks back and forth, and the multiple impacts with each leg dissipate the remaining energy. During this last phase the no slipping condition of the stance leg is violated: the robot slips. Then, one cannot rigorously conclude that the robot stops since the hypotheses of the model are violated. But, in 4.4.5.1, we use a more realistic compliant ground model allowing to properly confirm that the robot slips for a short duration ($\approx$ 0.6 s) until a complete stop.

Discussion. This study showed the possibility to stop the robot in "one step" by dissipating its kinetic energy at impact. Actually, since all the kinetic energy is not dissipated in one step, the robot stops for two reasons. First, because the majority of its kinetic energy is dissipated. Second, because the robot is servoed to a fixed configuration, that is statically stable and requiring energy to tip forward. The obtained configuration at impact correspond to relatively large steps. Actually, it is well known that a large step diminishes the velocity of a biped during the next step (see [90] and [118] (chapter 7)). Such configurations were naturally obtained with the motion planning algorithm.

The strategy that we proposed to stop a robot is novel. Others methods exist to stop a robot. See the theory of capturability in 2.3. This method uses approximative models of the robot and does not take the impact dynamics into account. Zutven et al. proposed an extension using the full model of the robot

[87, 118]. Westervelt et al. proposed to use an event-based PI control to slow down the robot RABBIT until it does not have enough energy to make a step (see 8.3.1.2, Experiment 6 in [127]). The robot needed five steps to slow down, and then rocked back and forth several times until its remaining kinetic energy was dissipated. On the contrary, our strategy allows to stop faster. But, obviously, it is more violent and may deteriorate the hardware. It would be interesting to investigate if a high relative energy dissipation at impact is also involved when humans stop in one step.

4.4.5 Robustness To Model Uncertainties

The results on the relative dissipation of energy at impact were obtained using an approximative model. As a matter of fact, the impact is not instantaneous. Furthermore, the parameter dependency (mass, length, inertia,...) has to be taken into account to conclude on the practical interest of our study. We investigate here the robustness to structural and parameter mismatch of our approach.

For that, we investigate the behavior of the motions previously obtained using the (more realistic) compliant ground model (non zero restitution coefficient) described in Section 1.7.1. Furthermore, we study the effects of using a set of dynamic parameters for the model used in simulation that is different from for the one used for the motion planning. The obtained results are close to the ideal case of a rigid ground model with a perfect knowledge of the dynamic parameters, that is to the results presented above.

4.4.5.1 Simulations Using a Compliant Ground Model

We present here a comparative study of the energy evolution between the rigid and the compliant ground model for some of the walking cycles described in figure 4.10. We use a high gain PD controller to track the virtual constraints in both cases. The results are depicted figure 4.13. It is not easy to estimate accurately the amount of kinetic energy dissipated when the swing foot hits the ground in the case of a compliant ground, since it is difficult to identify when the impact begins and stops. But, we notice that the shape of the kinetic energy is very similar in both cases. This supports the use of the rigid ground model for an analysis of the dissipation of kinetic energy at impact.

The scenario to stop the robot "in one step" is tested with a compliant ground model. See Figures 4.14 and 4.15. The behavior is also close to the case of a rigid ground model. After the step supposed to dissipate 82% of the kinetic energy at impact, the robot slips for 0.4 second on a distance of 3.4 cm until a complete stop. The slipping of the robot dissipates the remaining kinetic energy. Note that, contrary to the rigid ground model, the robot does not rock back and forth.

4.4.5.2 Robustness to Dynamic Parameters Uncertainties

Until now, simulations were performed assuming a perfect knowledge of the dynamic parameters of the robot. To investigate the robustness to variations of the dynamic

parameters, we multiply some of them by a scaling factor. It appears that, for one given state at impact, the relative energy variation depends little on these parameters. See figure 4.16, for which the mass, inertia, position of the center of the mass of the links of the robot, and the inertia of the mechanical transmission system are multiplied by the same scaling factor. Varying independently some dynamics parameters also gave similar results. Note that when changing only the dynamic parameters of the links, and so keeping the inertia of the mechanical transmission system unchanged, the relative energy variation changes more, see figure 4.17. But, the robustness is still relatively good.

Contrary to the variation of energy at impact (see the bottom plots in figure 4.16 and 4.17), the relative energy variation is robust to parameter mismatch. This is an additional reason that supports the use of the relative energy variation instead of the energy variation for control purposes.

4.5 Conclusion

In this chapter we studied the relative dissipation of kinetic energy at impact for biped robots. We first performed a theoretical study using a standard rigid impact model introduced by Hurmuzlu [56]. We formulated the problem as a generalized eigenvalue problem under constraints (= the impact model hypotheses). Ignoring the constraints of the generalized eigenvalue problem, we found that, for any configuration at impact, the sets of the velocities that dissipate either no, or the totality of the kinetic energy of the robot are vector subspaces of the tangent space. We gave sufficient conditions on the foot-ground contact before and after impact allowing the existence of such velocities. These results are only valid if the hypotheses of the impact model are verified. Next, we explained how to design motions bringing a maximally one degree underactuated robot at a desired value of relative energy variation at impact. The use of virtual constraints facilitate the problem, since a set of virtual constraints for one degree underactuated systems define a unique relative energy dissipation at impact, whatever the traveling velocity of the path they define in the configuration space.

This theoretical study was applied on a planar five-link walker with point feet. Using the HZD method, and our study, walking trajectories with a relative energy dissipation close to zero were designed. Contrary to walking trajectories obtained by minimizing the torque consumption without constraints on the relative energy dissipation, they have the interest of dissipating little energy at impact. This is an important point for the preservation of the structure of the robot, or to reduce the noise. But, this is at the expense of a higher torque consumption. We also showed that it was possible to design motions dissipating more than 80% of the kinetic energy at impact. Especially, we used the dissipative motion to stop the robot in "one step". Finally, we compared the previous results with a more realistic modeling. We used a compliant ground model, a high gain PD controller to track the virtual constraints, and introduced a mismatch in the dynamic parameters. This highlighted the robustness of the relative energy dissipation. This also supported the use of the standard rigid ground model for

estimating the dissipation of energy at impact.

This work showed that the relative energy dissipation offers interesting properties compared to the energy dissipation: possibility to analytically exhibit the states right before impact dissipating little or a lot of energy, easy control via the use of virtual constraints, and robustness to dynamic parameters uncertainties.

The presented results are promising. Here are some improvements for future works. First, our theoretical study does not always to conclude on the existence of states corresponding to a relative energy dissipation equal to 0 (no dissipation of energy) and -1 (total dissipation of energy) (see the theoretical result of Section 4.2.6). It would be interesting to refine the results. Then, the physical meaning of the vector subspaces of the states corresponding to a relative energy dissipation equal to -1 is abstruse. Maybe an other point of view may help to better understand this meaning. A better achievement is to analytically solve the generalized eigenvalue problem under constraints (4.12).

As for the control of the relative energy dissipation, it would be interesting to address the problem of robots with several degrees of underactuation. An example of a model with feet has also to be considered. Also, there exist maybe others practical interests for the control of the relative energy dissipation at impact. For instance, it would be interesting to investigate if the shape of the relative energy dissipation along a walking trajectory, around the nominal point of impact with the ground, has an influence on the robustness of the walk on uneven ground.

We proposed here to diminish the energy dissipation by the use of a feedback law. An other strategy to reduce impact losses is to design the robot with a mechanical compliance, for instance with passive springs at the end of each legs as for the DURUS robot [97]. The conjoint use of mechanical and control design may diminish further impact losses.

Finally, an experimental validation is still to be done. Unfortunately, we did not realize it on the VS robot, due to a lack of time, and since the two bars stabilizing mechanism makes the relative energy variation dependent on the position of the robot on the treadmill. This makes harder the design of motions bringing the robot at a given relative energy dissipation at impact.

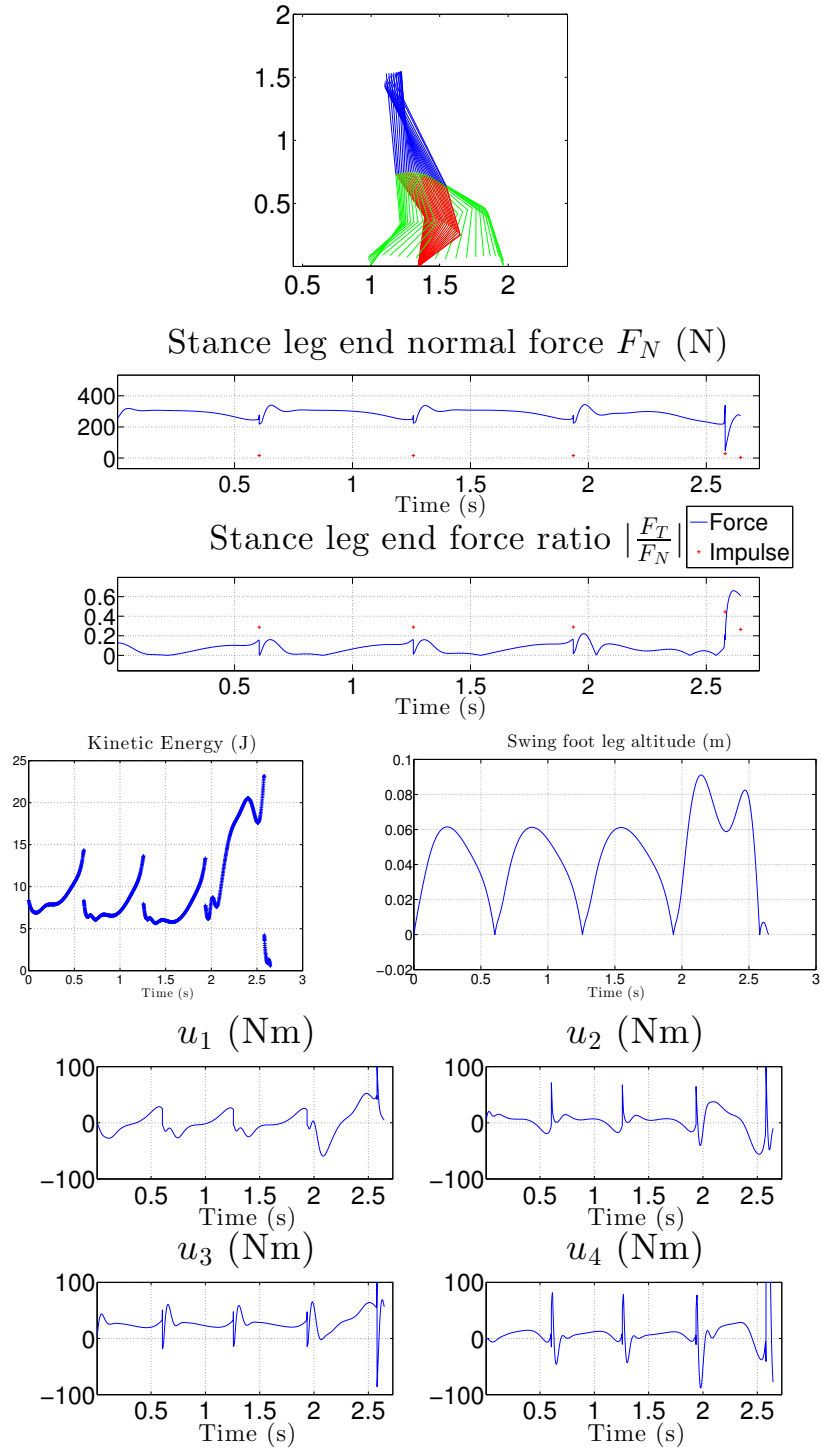


Figure 4.12: Illustration of a motion to stop the robot in "one step" from a walking cycle. The simulation uses a rigid ground model. Virtual constraints are tracked using a high gain PD controller. The robot walks three steps (0 s $\leq t \leq$ 1.94 s). The fourth step corresponds to a step with a relative energy motion at impact equal to -0.82 (1.94 s $\leq t \leq$ 2.58 s). Then, the robot is regulated to its standstill position ($t \geq$ 2.58 s). Note that the robot slightly rocks back and forth and slips ($|\frac{F_T}{F_N}| > 0.6$) during this last phase. For the sake of clarity, the stick diagram (top figure) represents only the step with a relative energy dissipation at impact equal to -0.82 (1.94 s $\leq t \leq$ 2.58 s).

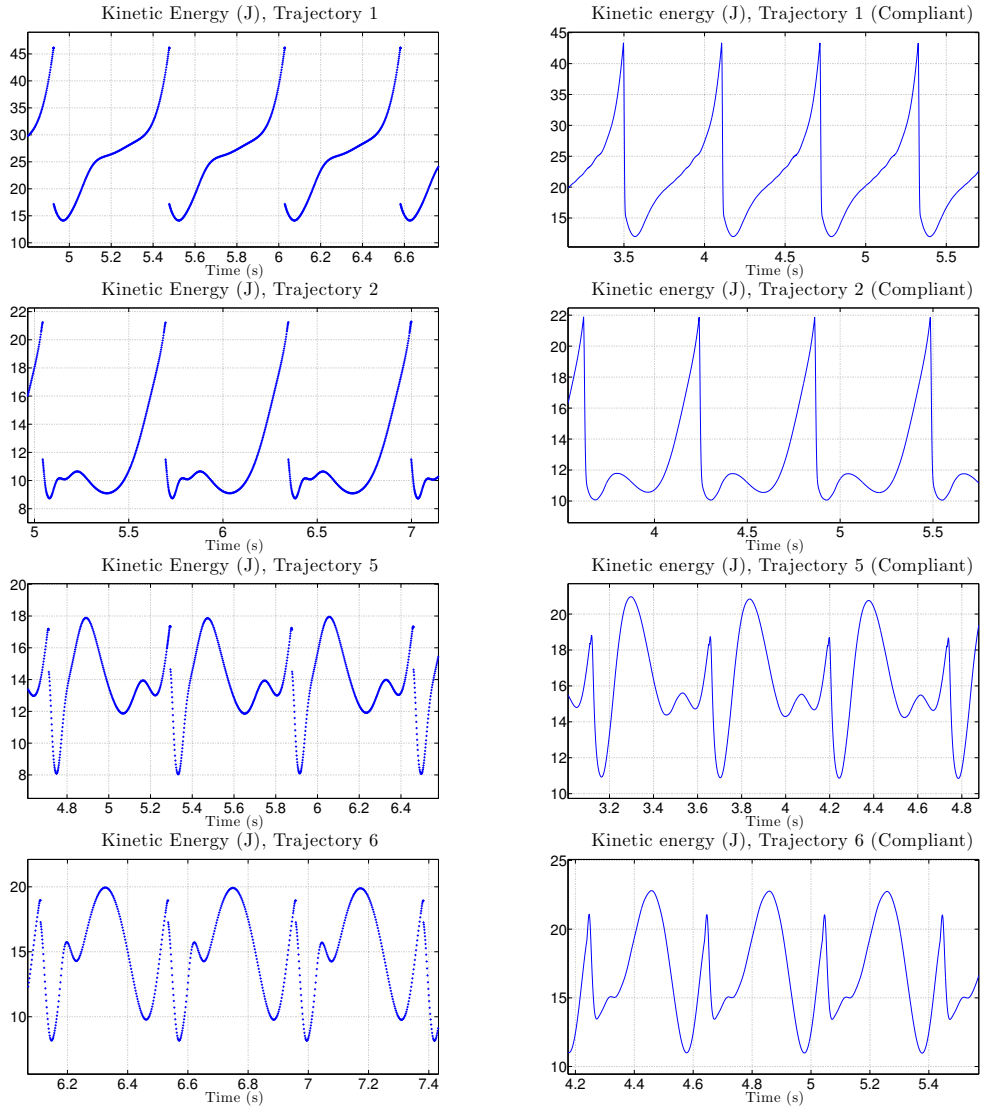


Figure 4.13: Comparison of the kinetic energy evolution for walking cycles using the rigid ground model (left), and the (more realistic) compliant ground model (right). The behavior is close for the two models. The biggest difference is seen for the trajectory 6. Note the existence of an offset on the upper and lower bounds of the kinetic energy between the two models.

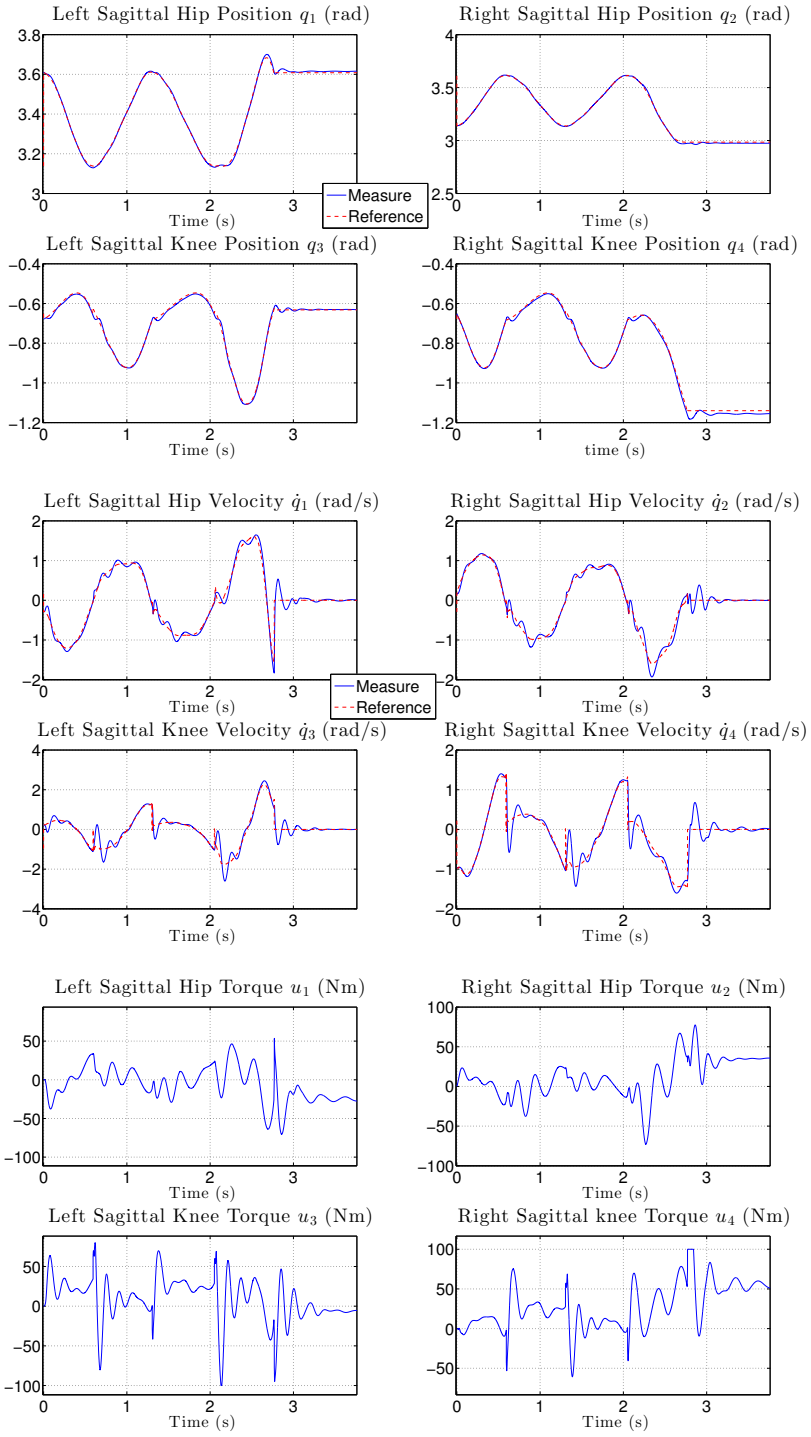


Figure 4.14: Simulation results using the compliant ground model to stop the robot in "one step". Joint positions, velocities and torques. Torques are saturated at 100 Nm. The robot walks three steps (0 s $\leq t \leq$ 2.05 s). The fourth step corresponds to a step with a relative energy motion at impact equal to -0.82 for a rigid ground model (2.05 s $\leq t \leq$ 2.8 s). In spite of tracking errors, especially for the joint velocity, the robot is capable to stop in "one step" (see figure 4.15).

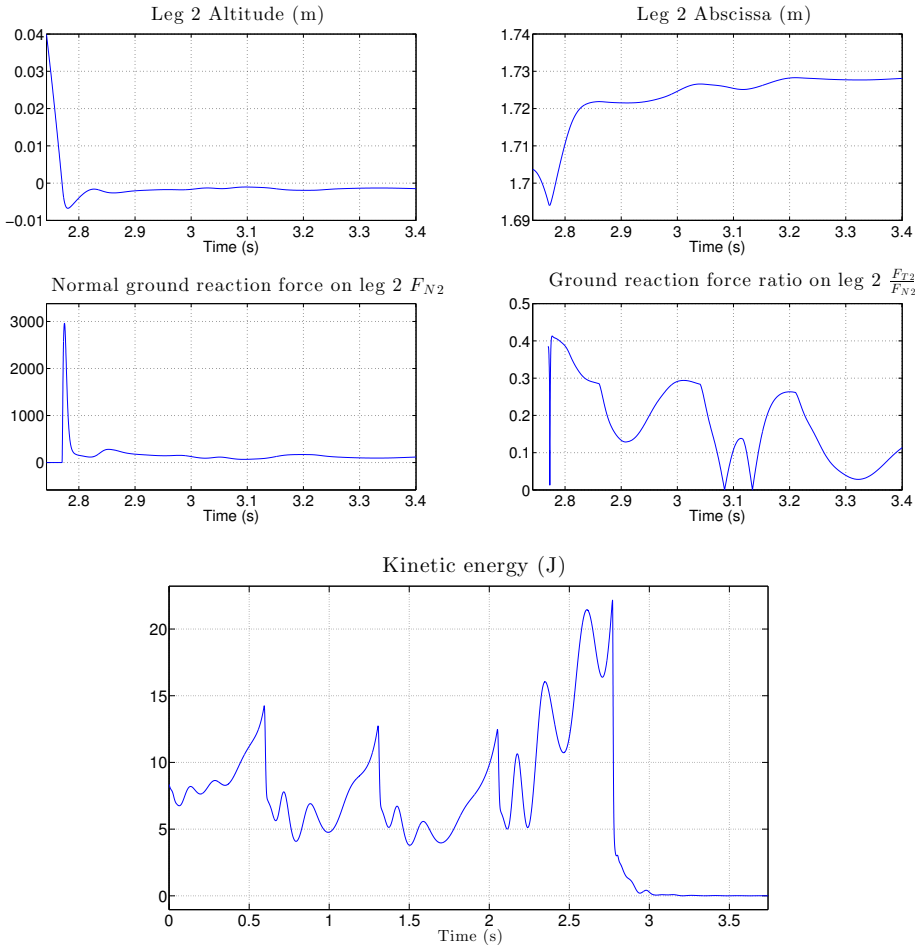


Figure 4.15: Simulation results using the compliant ground model to stop the robot in "one step". The robot walks three steps (0 s $\leq t \leq$ 2.05 s). The fourth step corresponds to a step with a relative energy motion at impact equal to -0.8 for a rigid ground model (2.05 s $\leq t \leq$ 2.8 s). The leg 1 hits the ground at $t = 2.8$ s. The relative energy dissipation is about to -0.82 as predicted by the rigid ground model (see figure 4.12). The leg slips over a distance of 3.4 cm for about 0.6 s (2.8 s $\leq t \leq$ 3.4 s), dissipating the remaining kinetic energy. The robot is then completely stopped.

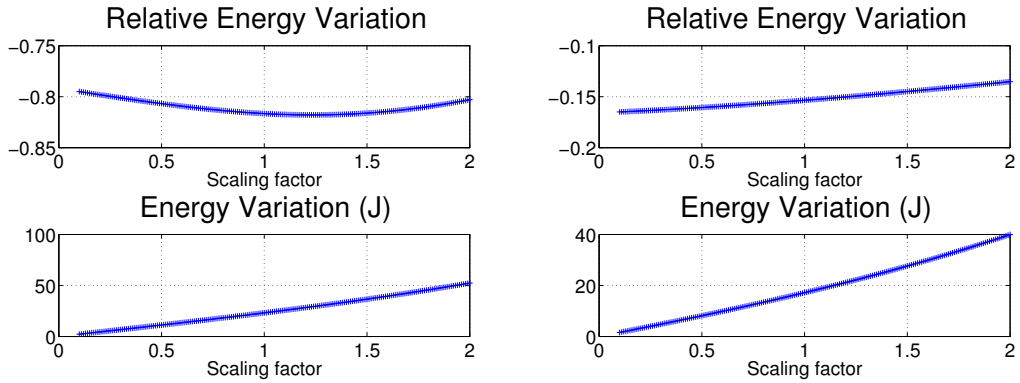


Figure 4.16: Evolution of the relative energy variation and of the variation of energy at impact when multiplying the mass, inertia, position of the center of mass of the links of the robot, and the transmission inertia, by a scaling factor. Left: For the state before impact used to stop the robot in "one step". Right: For the state before impact of the walking trajectory 5. Contrary to the energy variation, the relative energy variation remains approximately constant when the dynamic parameters are changed.

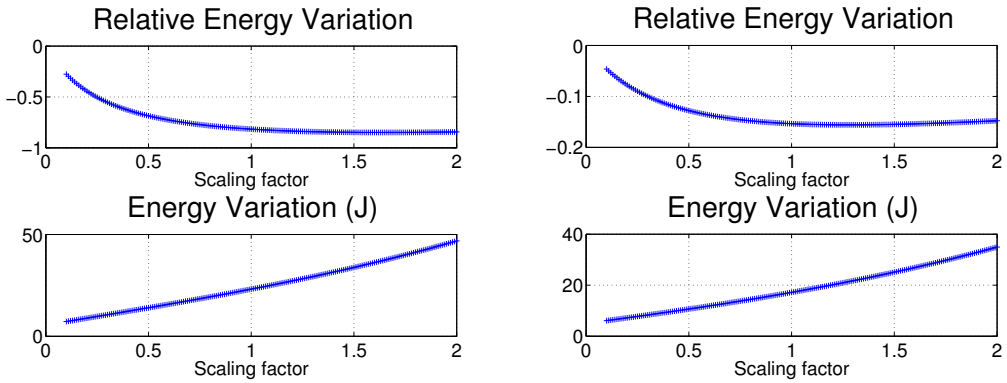


Figure 4.17: Evolution of the relative energy variation and of the variation of energy at impact when multiplying the mass, inertia, position of the center of mass of the links of the robot by a scaling factor. **Contrary to figure 4.16, the transmission inertia is left unchanged**. Left: For the state before impact used to stop the robot in "one step". Right: For the state before impact of the walking trajectory 5. The relative energy variation is less modified than the energy variation when the dynamic parameters are unchanged. Especially, it remains almost unmodified when the values of the dynamic parameters are higher than expected.

Chapter 5

Changing the Time Scale in the Equations of Motion: Application to Planar One Degree Underactuated Biped Robots With Point Feet

Résumé français Il peut être intéressant de modifier la forme des équations de la dynamique d'un système. En effet, certaines "formes normales" sont plus appropriées pour la planification de mouvement et la synthèse de contrôleurs. Il est généralement considéré que transformer, sans approximation, les équations sous la forme d'un système linéaire et controllable est la transformation la plus aboutie, puisque la littérature sur le contrôle des systèmes linéaires est riche. Quand obtenir une telle forme s'avère trop compliqué, des formes normales aussi linéaires que possibles sont dérivées. La dérivation de formes normales pour les robots bipèdes a été étudiée en particulier dans [48, 127] et les références qui y sont incluses.

En 1986, Sampei et Furuta ont montré que la technique dite de "Time-Scaling", qui consiste à changer l'échelle de temps dans les équations de la dynamique, peut modifier les conditions de linéarisation par feedback [103]. Ainsi, le Time-Scaling offre des nouvelles possibilités pour obtenir de nouvelles formes. Par exemple, [102] a construit une échelle de temps dans laquelle le compas est linéarisable par feedback. Il semblerait que cela soit la seule étude effectuée sur l'utilisation du Time-Scaling pour les robots bipèdes.

Pourtant, changer l'échelle de temps dans les équations de la dynamique des robots bipèdes semble d'autant plus intéressant que des travaux ont montré que le temps usuel t n'était pas forcément une bonne échelle de temps pour l'étude des robots bipèdes. Le succès de la méthode HZD repose sur le fait qu'elle utilise des contraintes virtuelles pour synchroniser les articulations du robots avec un paramètre géométrique qui joue le rôle d'un nouveau temps (appelé variable de phase, voir section 2.4.5). Ainsi, il semble cohérent d'utiliser ce paramètre géométrique comme échelle temps dans les équations de la dynamique.

Ce chapitre explore l'utilisation du "Time-Scaling" pour obtenir de nouvelles formes pour les robots bipèdes à un degré de sous actionnement et à pieds ponctuels, et son utilité pour la planification de trajectoires et la synthèse de contrôleurs. La section 5.2 présente la technique de Time-Scaling. Dans la section 5.3, nous dérivons les équations du mouvement pour les modèles considérés dans une nouvelle échelle de temps. Puis, dans les sections 5.4 et 5.5, nous dérivons des formes normales avec une dynamique des zéros de dimension deux et une. Nous n'avons pas réussi à linéariser de manière exacte les équations (dynamique des zéros de dimension nulle). Finalement, nous considérons le linéarisé des équations le long d'une trajectoire, mais dans une nouvelle échelle de temps. La forme obtenue est utilisée pour faire la synthèse d'un contrôleur linéaire quadratique dont les performances sont présentées en simulation sur un robot 5 liens plan.

5.1 Introduction

For control design purposes, it may be interesting to transform the shape of dynamical equations. Some shapes, or *normal forms*, are better suited for motion planning and stabilization applications. Actually, it is generally considered that transforming, without approximation, the equations into linear and controllable ones is the best achievement, since the literature on the control of linear systems is rich and accomplished. When this is not possible, normal forms as linear as possible are derived. The derivation of normal forms for biped robots has been especially studied in [48, 127] and the references therein.

In 1986, Sampei and Furuta showed that *Time-Scaling*, that is changing the time scale in the dynamical equations, leads to non trivial changes in terms of control, such that the conditions of feedback equivalence to a linear and controllable system [103]. Then, Time-Scaling offers opportunities to derive new normal forms. Especially, [102] designed a time scale in which the compass walker is statically feedback equivalent to a linear and controllable form. To the best of our knowledge, this is the only study performed on the use of Time Scaling for biped robots.

Yet, changing the time scale in the dynamical equations of biped robots seems to be all the more interesting as previous works showed that the ordinary time t does not seem to be an appropriate time scale for the study and control of biped locomotion. The success of the HZD method lies on the use of virtual constraints synchronizing the joints of the robot with a geometric parameter playing the role of a new time, and called phase variable (see Section 2.4.5). In the same idea, studies found that human joint patterns are best parameterized by a time-independent geometric variable than by the time (see [119] and the references therein). Then, it seems to be consistent to use this geometric variable as new time scale.

This chapter investigates the use of Time-Scaling to derive new normal forms for one degree underactuated biped robots with point feet and discusses their usefulness in terms of motion planning and trajectory stabilization. In Section 5.2, we present the concept of Time-Scaling. In section 5.3, we derive the equations of motion of one degree underactuated biped robots with point feet in a new time scale. Then, in Section 5.4, we derive normal forms with a two-dimensional

zero dynamics using a set of relative degree two outputs and a static feedback transformation. In Section 5.5, we derive a normal form with a one dimensional zero dynamics using outputs of relative degree two, one output of relative degree three, and a dynamic feedback transformation. We did not find a set of outputs resulting in a normal form with an empty zero dynamics (linear and controllable form). Exhibiting one output of relative degree four would allow to obtain such a form. The only result that we obtained on this research is presented in Section 5.6. We explain that there probably exists no time scale for which the antiderivative of the angular momentum σ is of relative degree four, except for the compass walker. Finally, after having considered exact normal forms, we consider the first-order approximation of the dynamics along a reference trajectory. This normal form is used to design a linear quadratic regulator under the form of a state feedback controller, that is a controller depending only on the state and not on t, to stabilize a walking trajectory. The performances of the controller are evaluated on a numerical example of a planar five-link walker.

5.2 Time-Scaling

Let us assume that the equations of motion are

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ \dot{t} = 1 \end{cases} \tag{5.1}$$

with $x \in \chi \subset \mathbb{R}^n$ is the vector of states, f and g are C^1 vector fields on χ and $u \in \mathbb{R}^m$ is the vector of control inputs. Since we are to change the time scale, we consider the time as an additional state component and add its (trivial) dynamics. Nonetheless, due to the peculiar role of t and due to the common usage, we will keep to continue to abusively call x as the state. To remove any ambiguity, we will precise when necessary if the term "state" stands for x or for (x, t).

Given an input $t \to u(t)$, defined in open or closed loop, let

$$\begin{cases} x(t) = \Phi(t) \\ t = t \end{cases} \tag{5.2}$$

be a solution of (5.1).

Let $x \to s(x) \in]0, \infty[$ be a continuous function, called *time scaling function*. Let $\tau(t)$ be a solution of

$$\frac{d\tau}{dt} = \frac{1}{s(x(t))}\,, \tag{5.3}$$

i.e.

$$\tau = \int_{t_0}^{t} \frac{dt}{s(x)} + \tau_0\,, \tag{5.4}$$

where τ_0 is the value of τ when $t = t_0$.

τ is strictly monotonic and defines a new time scale, i.e. it can replace the actual time t. Thus, there exists a C^1 function satisfying $x(t) = X(\tau(t))$. Integrating (5.3) gives the one-to-one mapping between τ and t

$$t = T(\tau) = t_0 + \int_{\tau_0}^{\tau} s(X(\tau))d\tau. \tag{5.5}$$

Let us derive now the equations of motion for this new time scale. The solution of (5.1) written in the new time scale is $(X(\tau), T(\tau))$. Differentiating this solution with respect to the time t gives

$$\begin{cases} \dot{x} = f(x) + g(x)u = X'(\tau)\dot{\tau} \\ 1 = s(x)\dot{\tau} \end{cases} \tag{5.6}$$

Since $X'(\tau) = \frac{dx}{d\tau}$ and $s(x) \neq 0$, we get

$$\begin{cases} \dfrac{dx}{d\tau} = s(x)\Big(f(x) + g(x)u\Big) \\ \dfrac{dt}{d\tau} = s(x) \end{cases} \tag{5.7}$$

Therefore, (5.7) are the equations of the dynamical system written in the new time scale τ. (5.1) and (5.7) are the equations of motion of the same dynamical system, but written in different time scales. They have the same solutions if and only if the new time scale does not go backward against the actual time t ($s(x) > 0$). $s(x) < 0$ is not valid, otherwise the system would evolve in the backward direction.

Remark 1: We can give a geometric interpretation for this derivation. Considering the time t as an additional state component boils down to work in the time-space $\chi \times \mathbb{R}$. The function $t \to \Phi(t)$ defines a curve in this space and the equation $x = \Phi(t)$ is a cartesian representation of the solution. The new time scale τ corresponds to a curvilinear abscissa of the curve and allows to define a parametric representation of the solution.

Remark 2: Even though Time-Scaling has been introduced for continuous systems, it is also applicable to hybrid systems like biped robots. The only difference is that the time derivative of the time scale τ, i.e. the time scaling function $s(x)$, may undergo a jump when the discrete dynamics is invoked. But τ is continuous.

Remark 3: It may be fruitful to consider the time scaling function s as a new input $v \in \mathbb{R}$ to the system, i.e. to let

$$\frac{dt}{d\tau} = v. \tag{5.8}$$

Remark 4: We insist on the fact that it is preferable to add the actual time t as a state component when using Time-Scaling. Indeed, we are interested to reconstruct the solutions $t \to x(t)$ of the dynamical equations in the time scale t from the solutions found in the new time scale τ: $\tau \to X(\tau)$. For that, we need to know the mapping between τ and t, i.e. $\tau \to T(\tau)$, that is obtained by integrating the quantity $\frac{dt}{d\tau}$.

Additionally, not considering the time t as a state component could lead to a misunderstanding of what the change of time scale really does. For example, let us consider the one dimensional system without input $\dot{x} = f(x)$ and the time scaling function (5.8). Choosing $v = \frac{v_2}{f(x)}$, where $v_2 \in \mathbb{R}$ is the new input, results in

$$\frac{dx}{d\tau} = v_2. \tag{5.9}$$

One could believe that the system is now entirely controllable with this feedback linearization. But, this is illusory. One cannot both control here the dynamics of x and of τ. The dynamics of the time scale τ, that is not trivial anymore due to the change of feedback, must be considered

$$\frac{dt}{d\tau} = -\frac{v_2}{f(x)}. \tag{5.10}$$

Especially, the strict positivity of $\frac{dt}{d\tau}$ must be checked.

5.3 Time-Scaling for Planar One Degree Underactuated Biped Robots

The modeling of a biped robot is described in Chapter 1. We remind here some useful notations and properties of the considered class of robots. We consider a biped robot with point feet in single support phase. The robot forms a tree structure pinned on the ground. It has n independent degrees of freedom, $n-1$ of them, the joints of the robot, are independently actuated. We use the reduced coordinates. We denote $q \in \mathcal{Q} \subset \mathbb{R}^n$ the generalized positions and $x = (q; \dot{q}) \in \mathcal{TQ} \subset \mathbb{R}^{2n}$ the state of the robot. Whithout loss of generality, we assume that the generalized positions are chosen such that the $(n-1)$ first components of $q = (q_1; q_2; ...; q_n)$ are the (actuated) joint coordinates of the robot. The last component q_n is the absolute orientation of the robot (not actuated). The actuated coordinates are denoted

$$q_a = (q_1; q_2; ...; q_{n-1})\,. \tag{5.11}$$

The equations of motion can be written under the form

$$D(q)\ddot{q} + H(q, \dot{q}) = Bu\,, \tag{5.12}$$

where $u \in \mathbb{R}^{n-1}$ is the vector of joint torques, $H(q, \dot{q})$ is a vector containing the Coriolis term and the gravity vector

$$H(q,\dot{q}) = C(q,\dot{q})\dot{q} + G(q)\,, \tag{5.13}$$

and

$$B = \begin{pmatrix} I_{(n-1)\times(n-1)} \\ 0_{1\times(n-1)} \end{pmatrix}. \tag{5.14}$$

A property that will be used next is that the unactuated position q_n is a cyclic coordinate. This means that the mass matrix D is independent of q_n

$$\frac{\partial D}{\partial q_n}(q) = 0\,. \tag{5.15}$$

See proposition B.8 in [127] for a proof.

The equations under the state space form are

$$\dot{x} = \begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} \omega \\ D^{-1}(q)[-H(q,\omega) + Bu] \end{pmatrix}. \tag{5.16}$$

For the sake of brevity, we let

$$l(q,\omega,u) = D^{-1}(q)[-H(q,\omega) + Bu]\,. \tag{5.17}$$

From (5.7), the dynamical equations in the new time scale τ are

$$\frac{dx}{d\tau} = \begin{pmatrix} \frac{dq}{d\tau} \\ \frac{d\dot{q}}{d\tau} \\ \frac{dt}{d\tau} \end{pmatrix} = \begin{pmatrix} s(q,\omega)\omega \\ s(q,\omega)l(q,\omega,u) \\ s(q,\omega) \end{pmatrix}, \tag{5.18}$$

where $s(q,\omega)$ is the time scaling function. The Lagrange's equations (5.12) become

$$\frac{1}{s(q,\omega)} D(q)\frac{d\omega}{d\tau} + C(q,\omega)\omega + G(q) = Bu + \Gamma_f\,. \tag{5.19}$$

The equations are valid in the subspace

$$\mathcal{TQ}_{s>0} = \{x := (q;\omega) \in \mathcal{TQ} \,|\, s(q,\omega) > 0\}. \tag{5.20}$$

From now on, we will denote the τ-derivative of x

$$\mathring{x} = \frac{dx}{d\tau}\,. \tag{5.21}$$

Remark: Actually, the modeling hypotheses that we adopt here correspond to the more general class of one degree underactuated mechanical systems, the mass matrix of which does not depend on the unactuated position.

5.3.1 Illustration on the Effect of a Change of Time Scale

We illustrate here the effect that a change of time scale may have on the properties of convergence of the hybrid zero dynamics of a planar one degree underactuated robot with point feet, and more precisely on the convergence of the square of the angular momentum of the robot σ^2.

5.3.1.1 The Angular Momentum σ

The angular momentum of the robot σ expressed at the point of contact with the ground has a strong physical meaning. It quantifies the rotating motion of the system about this point. We remind that the generalized positions introduced to describe the configurations of the robot are $(q_1, q_2, .., q_n)$, where q_n is the unactuated coordinate. Let us consider the generalized (or conjugate) momentum associated to q_n, that is to the angle of the pivot between the stance foot and the ground [43]

$$\sigma = \frac{\partial \mathcal{L}}{\partial \dot{q}_n} \,, \tag{5.22}$$

with $\mathcal{L} = \mathrm{K} - \mathrm{V}$, where $\mathcal{L}$ is the Lagrangian of the system, $\mathrm{K} = \frac{1}{2}\dot{q}^T D(q)\dot{q}$ the kinetic energy, and $\mathrm{V} = V(q)$ the potential energy. We obtain

$$\sigma = B^{\perp} D(q)\dot{q} \,, \tag{5.23}$$

where $B^{\perp}$ is a $(1 \times n)$ matrix such that

$$B^{\perp} B = 0_{1\times m} \,. \tag{5.24}$$

Due to the expression of B (see (5.14)), we have

$$B^{\perp} = \begin{pmatrix} 0_{1\times(n-1)} & 1 \end{pmatrix} \,. \tag{5.25}$$

Let us compute the time derivative of σ. We remind that the Lagrange's equations can be written under the form (see (1.6))

$$\frac{d}{dt}\Big(D(q)\dot{q}\Big) - \frac{1}{2}\Big(\frac{\partial}{\partial q}\Big(\dot{q}^T D(q)\dot{q}\Big)\Big)^T + G(q) = Bu \,. \tag{5.26}$$

By multipling the equations by $B^{\perp}$, and since q_n is a cyclic coordinate, that is $\frac{\partial D}{\partial q_n}(q) = 0_{n\times n}$, we have

$$B^{\perp}\Big(\frac{\partial}{\partial q}\Big(\dot{q}^T D(q)\dot{q}\Big)\Big)^T = 0 \,. \tag{5.27}$$

Then, we get from (5.26)

$$\dot{\sigma} = -B^{\perp} G(q) \,. \tag{5.28}$$

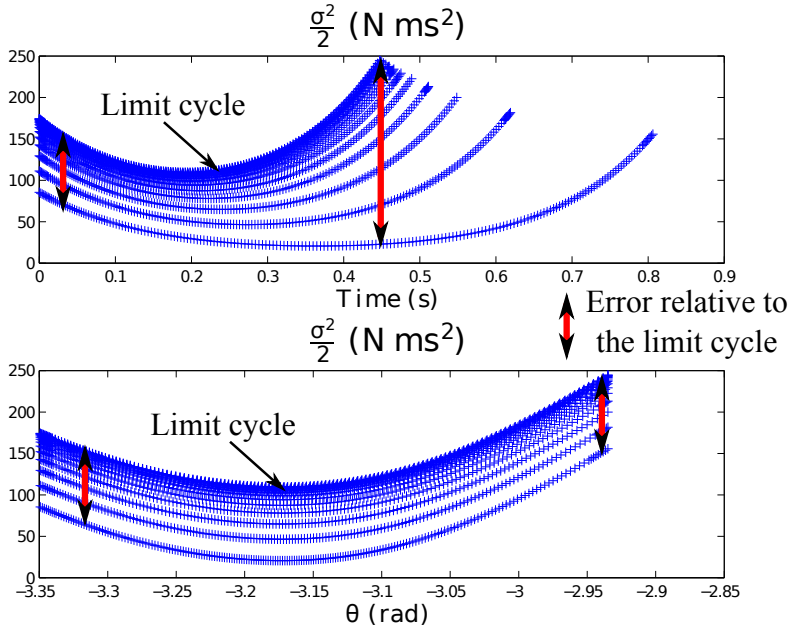


Figure 5.1: Evolution and convergence of $\frac{\sigma^2}{2}$ to a limit cycle. To allow a better comparison of the curves, the time t is reinitialized at zero at each beginning of a step in the above plot. One clearly remarks that over a step, $\frac{\sigma^2}{2}$ diverges from the limit cycle along the time, while it is marginally stable along θ.

5.3.1.2 Convergence of $\frac{\sigma^2}{2}$ to a Limit Cycle

Let us consider a walking trajectory with an instantaneous double support phase designed with the HZD method for the planar five-link walker with point feet described in Section 4.4.1. A set of outputs for the actuated coordinates q_a under the form

$$y = q_a - q_{a,ref}(\theta)\,, \tag{5.29}$$

where θ is the phase variable and $\theta \to q_{a,ref}(\theta)$ is the motion of reference, is imposed via a feedback controller. Then, the robot is constrained to evolve in the hybrid zero dynamics manifold. This manifold is parameterized by $(\theta, \dot{\theta})$, or equivalently by $(\theta, \frac{\sigma^2}{2})$ as long as $\dot{\theta} > 0$. Since $\dot{\theta} > 0$, then a new time scale τ can be defined as $\frac{dt}{d\tau} = \frac{1}{\dot{\theta}}$.

By design, the hybrid zero dynamics converges to a limit cycle. Let $t \to \frac{\sigma^2_{ref,t}(t)}{2}$ be the limit cycle parameterized by the ordinary time t, and let $\theta \to \frac{\sigma^2_{ref,\theta}(\theta)}{2}$ be the same limit cycle, but parameterized by θ. Figure 5.1 depicts the evolution of $\frac{\sigma^2}{2}$ along the time, and along θ. The shape of the convergence of $\frac{\sigma^2}{2}$ to the limit cycle is clearly different. $\frac{\sigma^2}{2}$ diverges from the limit cycle $t \to \frac{\sigma^2_{ref,t}(t)}{2}$ during each step: the error relative to the limit cycle increases, while the error is constant relative to $\theta \to \frac{\sigma^2_{ref,\theta}(\theta)}{2}$ (marginal stability).

The marginal stability of $\frac{\sigma^2}{2}$ can be proved analytically. In the time scale τ the error relative to the limit cycle $\theta \to \sigma^2_{ref,\theta}(\theta)$ is

$$e_\theta = \frac{\sigma^2}{2} - \frac{\sigma^2_{ref,\theta}(\theta)}{2}\,. \tag{5.30}$$

By definition, when the system evolves in the hybrid zero dynamics, q is only function of θ, i.e. under the form $q = h_d(\theta)$. Then, the dynamics of $\frac{\sigma^2}{2}$ in time scale τ is

$$\overset{\circ}{\overbrace{\frac{\sigma^2}{2}}} = -B^\perp G(h_d(\theta))B^\perp D(h_d(\theta))h_d^{'}(\theta)\,. \tag{5.31}$$

This quantity depend only on θ. Hence $\overset{\circ}{\overbrace{\frac{\sigma^2}{2}}} = \overset{\circ}{\overbrace{\frac{\sigma^2_{ref,\theta}(\theta)}{2}}}$ and

$$\overset{\circ}{e_\theta} = 0\,. \tag{5.32}$$

This proves that the error is constant during one step in the time scale τ (bottom plot in figure 5.1).

On the contrary, in the ordinary time t, the error relative to the limit cycle $t \to \sigma^2_{ref,t}(t)$ is

$$e_t = \frac{\sigma^2}{2} - \frac{\sigma^2_{ref,t}(t)}{2}\,. \tag{5.33}$$

Its time derivative is

$$\dot{e_t} = -\frac{1}{B^\perp D(q)h_d^{'}(\theta)}B^\perp G(h_d(\theta))B^\perp D(h_d(\theta))h_d^{'}(\theta)\sigma - \sigma^{'}_{ref,t}(t)\sigma_{ref,t}(t) \neq 0 \tag{5.34}$$

Then, the error relative to the limit cycle evolves during one step.

This illustrates the non obvious effects that a change of time scale (= of clock) may engender.

5.4 Static Feedback Equivalence to a Partial Linear System

In the following sections, we use feedback transformations. A brief presentation of these techniques can be found in Appendix G.

5.4.1 Static Feedback Equivalence in the Ordinary Time t

It is well known that the actuated dynamics of a mechanical system can be input-output linearized. The resulting normal form has been widely used for the study of underactuated mechanical systems [110, 84, 48, 99].

More precisely, for one degree underactuated planar robots with point feet, $y = q_a$ defines a vector of outputs with a vector relative degree equal to $(2, ...2)$, and the dynamics is globally static feedback equivalent to the following normal form

$$\begin{cases} \dot{q}_a = \omega_a \\ \dot{q}_n = \omega_n \\ \dot{\omega}_a = v \\ \dot{\omega}_n = f_n(q, \omega, v) \end{cases} \tag{5.35}$$

where v is the new input and f_n is a smooth vector field.

A set of outputs containing the unactuated coordinate can also be input-output linearized under certain conditions [110]. More precisely, let q_c be $n-1$ linear combinations of the joint coordinates q and q_u be a linear combination of the joint coordinates such that $(q_c; q_u)$ defines a set of generalized positions. We have

$$q_c = M_c q \, , q_u = M_u q \, , \tag{5.36}$$

where M_c is a constant $(n-1) \times n$ dimensional matrix and M_u is a constant n-dimensional row vector. Since $(q_c; q_u)$ defines a set of generalized coordinates, then

$$q = N_c q_c + N_u q_u \, , \tag{5.37}$$

with N_c, a constant $n \times (n-1)$ dimensional matrix, and N_u, a constant n-dimensional column vector, such that

$$N_c M_c + N_u M_u = I_{n \times n}. \tag{5.38}$$

If the decoupling matrix

$$M_c D(q)^{-1} B \tag{5.39}$$

is locally invertible, then the output $y = q_c$ has a vector relative degree equal to $(2, ..., 2)$ and the static change of feedback

$$v = M_c D(q)^{-1}(-H(q, \dot{q}) + Bu) \tag{5.40}$$

results in the following normal form

$$\begin{cases} \dot{q}_c = \omega_c \\ \dot{q}_u = \omega_u \\ \dot{\omega}_c = v \\ \dot{\omega}_u = f_u(q, \omega, v) \end{cases} \tag{5.41}$$

where f_u is a smooth vector field. The normal form is made of $n-1$ chains of integrators (the dynamics of $\dot{q}_c = v$), and of a two-dimensional zero dynamics. One component of the zero dynamics is linear ($\dot{q}_u = \omega_u$). In this section, we generalize this result to any time scale τ.

5.4.2 Generalization to a Time Scale τ

To get the normal form (5.41) in a new time scale, we perform a change of coordinates as a first step. Indeed, the dynamics in the ordinary time t, and in the coordinates formed of the generalized positions and velocities $(q;\omega)$, is naturally made of n linear equations ($\dot{q} = \omega$). See (5.16). But, as we can see in (5.18), changing the time scale breaks this structure. For obtaining the same structure as in the ordinary time t, and especially so that one component of the zero dynamics be linear, the dynamical equations should be written in the coordinates $(q;\mathring{q})$.

In section 5.4.3, we investigate the conditions on the time scale so that $(q;\mathring{q})$ defines a valid set of coordinates and derive the normal form (5.41) with the additional dynamics of t, in these coordinates. In Section 5.4.4, we treat the particular case where the time scale is a linear function of the generalized positions. This is an important case, since the phase variables used in the HZD method, which are generally defined as a linear combination of the generalized positions, define naturally a time scale in the neighborhood of the trajectories they parameterize (see Section 2.4.5). For this particular case, the set of coordinates $(q;\mathring{q})$ is not valid. We introduce an other set of coordinates, and derive a normal form with a two-dimensional zero dynamics, which is close to (5.41).

5.4.3 A state Under the Form $(q;\mathring{q})$.

From now on, we denote

$$\varpi = \mathring{q} = s(q,\omega)\omega\,. \tag{5.42}$$

We would like $(q;\varpi)$ to define a set of coordinates, or equivalently the mapping Φ between the original coordinates $(q;\omega)$ and $(q;\varpi)$ to be a local diffeomorphism

$$\begin{pmatrix} q \\ \varpi \\ t \end{pmatrix} = \begin{pmatrix} q \\ s(q,\omega)\omega \\ t \end{pmatrix} = \Phi(q,\omega)\,. \tag{5.43}$$

A standard result in differential geometry states that a necessary and sufficient condition is that the jacobian matrix of the transformation is invertible. The jacobian of Φ is

$$\frac{\partial \Phi}{\partial x}(q,\omega) = \begin{pmatrix} I_{n\times n} & 0_{n\times n} \\ \frac{\partial s}{\partial q}(q,\omega)\omega & \omega\frac{\partial s}{\partial \omega}(q,\omega) + s(q,\omega)I_{n\times n} \end{pmatrix}\,. \tag{5.44}$$

Then, it is clear that the change of coordinates is valid if and only if

$$\omega\frac{\partial s}{\partial \omega}(q,\omega) + s(q,\omega)I_{n\times n} \tag{5.45}$$

is an invertible matrix.

We make the assumption that this property is verified in an open set $\mathcal{U}$ of $\mathcal{TQ}_{s>0}$ and restrict our study to $\mathcal{U}$.

To define the dynamics in these new coordinates, we must calculate the τ derivative of ϖ. We have

$$\begin{aligned}\frac{d\varpi}{d\tau} &= \frac{d}{d\tau}\Big(s(q,\omega)\omega\Big)\\ &= \frac{d\Big(s(q,\omega)\Big)}{d\tau}\omega + s(q,\omega)\frac{d\omega}{d\tau}\\ &= \frac{1}{s(x)}\left[\frac{\partial s}{\partial q}(x)\varpi + \frac{\partial s}{\partial \omega}(.)l(.,u)s(.)\right]\varpi + s^2(x)l(x,u)\,,\ \text{using (5.18)}\\ &= \frac{1}{s(x)}\left[\frac{\partial s}{\partial q}(x)\varpi\right]\varpi + \left[\varpi\frac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\right]l(x,u)\,,\end{aligned} \tag{5.46}$$

where, for the sake of brevity, the argument x stands for $(q;\omega)$.

Hence, in $\mathcal{U}$, from the definition of ϖ, the dynamical equations written in the coordinates $(q;\varpi)$ have n linear equations

$$\left\{\begin{array}{l}\mathring{q} = \varpi\\ \mathring{\varpi} = \dfrac{1}{s(x)}\left[\dfrac{\partial s}{\partial q}(x)\varpi\right]\varpi + \left[\varpi\dfrac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\right]l(x,u)\\ \mathring{t} = s(x)\end{array}\right. \tag{5.47}$$

Remark: It is clear that the change of coordinates (q,ϖ) is valid on $\mathcal{TQ}_{s>0}$ if the time scaling function $s(q,\omega)$ does not depend on the velocity. Note that the ordinary time t corresponds to the particular case where $s(q,\omega)=1$.

5.4.3.1 A Normal Form With a Two-Dimensional Zero Dynamics

The dynamical equations (5.47) have the same structure as in the ordinary time t. We follow here the approach initially introduced by Spong [110] to derive a normal form with a two-dimensional zero dynamics.

In the coordinates $(q_c, q_u, \varpi_c, \varpi_u, t)$, the dynamical equations become

$$\left\{\begin{array}{l}\mathring{q}_c = \varpi_c\\ \mathring{q}_u = \varpi_u\\ \mathring{\varpi}_c = M_c\Big(\dfrac{1}{s(x)}\left[\dfrac{\partial s}{\partial q}(x)\varpi\right]\varpi + \left[\varpi\dfrac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\right]l(x,u)\Big)\\ \mathring{\varpi}_u = M_u\Big(\dfrac{1}{s(x)}\left[\dfrac{\partial s}{\partial q}(x)\varpi\right]\varpi + \left[\varpi\dfrac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\right]l(x,u)\Big)\\ \mathring{t} = s(x)\end{array}\right. \tag{5.48}$$

where x stands for $(q_c, q_u, \varpi_c, \varpi_u, t)$.

Assuming that the $(n-1) \times (n-1)$ decoupling matrix

$$M_c\Big(\frac{1}{s(x)}\Big[\frac{\partial s}{\partial q}(x)\varpi\Big]\varpi + \Big[\varpi\frac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\Big]D^{-1}(q)B\Big) \tag{5.49}$$

is invertible, the regular static feedback transformation

$$\begin{aligned} u = \psi^{-1}(x,v) = &\Big(M_c\Big(\frac{1}{s(x)}\Big[\frac{\partial s}{\partial q}(x)\varpi\Big]\varpi + \Big[\varpi\frac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\Big]l(x,u)\Big)\Big)^{-1} \\ &\times \Big(H(x) + v\Big). \end{aligned} \tag{5.50}$$

gives

$$\begin{cases} \mathring{q}_c = \varpi_c \\ \mathring{q}_u = \varpi_u \\ \mathring{\varpi}_c = v \\ \mathring{\varpi}_u = M_u\Big(\frac{1}{s(x)}\Big[\frac{\partial s}{\partial q}(x)\varpi\Big]\varpi + \Big[\varpi\frac{\partial s}{\partial \omega}(x) + s^2(x)I_{n\times n}\Big]l\Big(x, \psi^{-1}(x,v)\Big)\Big) \\ \mathring{t} = s(x) \end{cases} \tag{5.51}$$

The form is made of $n-1$ chains of integrators of dimension two, and of a two-dimensional zero dynamics (ignoring the dynamics of t). Note that the form introduced by Spong in the time scale t [110] is a particular case of (5.51), obtained when setting $s(x) = 1$.

5.4.4 Treating the Particular Case $\tau \sim M_\theta q$.

The choice of the time scale is an additional degree of freedom for the control of a dynamic system. But, it is not obvious how to choose this time scale. Nonetheless, in bipedal locomotion, there exist some geometric quantities, such as the virtual stance leg angle, which generally evolve strictly monotonically during one step. They can be used to parameterize the motion instead of the time t, and are called phase variables in the HZD method (see Section 2.4.5). Then, a natural choice is to choose as a new time scale a phase variable θ.

This choice is also motivated by the fact that the standard stability analysis of periodic walking trajectories is performed via the use of a Poincaré map (see Appendix A). This analysis is not based on the time t, since the state of the system is sampled when it crosses an hypersurface in the state space (the Poincaré surface). There are many ways of defining this surface. One possibility is to choose this surface as the set of all the states corresponding to a given value of the phase variable. In this case, the phase variable θ is the clock used for the stability analysis. Hence, for the sake of consistency, it seems more appropriate to use this

same clock in the equations of motion.

Phase variables are generally chosen as a linear combination of the generalized positions. In this section, we address this particular case. In that case, we see here that the time scaling function $s(q, \omega)$ derived from the phase variable does not verify the condition (5.45) and the coordinates (q, ϖ) cannot be used (see (5.57)). Then, we introduce a valid set of coordinates for this particular choice of time scale and such that the dynamical equations have $n-1$ linear equations. We derive a normal form with a two-dimensional zero dynamics in these new coordinates.

5.4.4.1 A Linear Combination of the Positions as a New Phase Variable.

Let θ be a linear combination of the positions

$$\theta = M_\theta q,\ \dot{\theta} = \omega_\theta = M_\theta \omega\,, \tag{5.52}$$

where M_θ is a constant n-dimensional row vector (see for example (3.11) or [127]).

Let us consider a motion realized by the robot, for instance a walking trajectory. Let us assume that $\dot{\theta} > 0$ in a neighborhood of this motion [1]. We carry out the study in this neighborhood. Then, using the term used in the HZD method, θ can play the role of a phase variable, i.e. it can be used to parameterize the motion instead of the time t. Additionally, θ is a natural candidate for defining a time scale τ as follows

$$\frac{dt}{d\tau} = \frac{1}{M_\theta \omega} = \frac{1}{\dot{\theta}}\,. \tag{5.53}$$

This is a valid choice as long as $\dot{\theta} > 0$.

Thus

$$\tau = \int \dot{\theta}\, dt. \tag{5.54}$$

Note that τ and θ are equal up to an additive constant. In the case of a periodic walking motion, the additive constant increases at each step. Then, the value of τ during the n^{th} step is

$$\tau = (\theta - \theta_n) + \tau_n\,, \tag{5.55}$$

where τ_n (resp. θ_n) is the value of τ (resp. θ) at the beginning of the n^{th} step.

Even though they are closely related, we insist that τ and θ have not the same meaning. The phase variable θ is a geometric parameter and is bounded

[1] The strict monotonicity of θ can be imposed through the use of a constraint in the motion planning algorithm described in Section 2.4.6. Otherwise, M_θ can be searched such that $M_\theta \dot{q} \neq 0$ along the considered motion.

to evolve in a given interval $[\theta^+, \theta^-]$ since it undergoes a jump at each step (see the case of the VS robot, and especially (3.11) and (3.12)). On the contrary, like the ordinary time t, τ is continuous and strictly monotonic. This is the "only" difference between τ and θ. That's why we denote $\tau \sim \theta$. Even if it is not fully rigorous, reasoning with $\tau = \theta$ may help to better understand what follows.

5.4.4.2 Using the Coordinates $(q_c; \theta; \varpi_c; \omega_\theta)$

The time scaling function is

$$s(q, \omega) = \frac{1}{M_\theta \omega} = \frac{1}{\omega_\theta}. \tag{5.56}$$

With this time scaling function, (q, ϖ) is not a valid set of coordinates. The problem comes from the fact that ω cannot be reconstructed from $\varpi = \frac{\omega}{M_\theta \omega}$. Indeed, this latter quantity is independent of the norm of ω. Then, ϖ cannot be used as coordinates anywhere.

This can also be seen using the jacobian of the coordinates transformation (5.43). The jacobian is invertible if and only if

$$\frac{1}{M_\theta \omega}\Big(- \frac{\omega M_\theta}{M_\theta \omega} + I_{n\times n}\Big) \tag{5.57}$$

is invertible. Remarking that ω belongs to the kernel of this latter matrix, one deduces that it is never invertible.

Then, to address this issue, we introduce the datum ω_θ in the coordinates. Additionally, to get $n-1$ chains of integrators $\mathring{q}_c = \varpi_c$, we introduce the generalized coordinates $(q_c; \theta)$, with

$$q_c = M_c q \,, \tag{5.58}$$

where M_c is a constant $(n-1) \times n$ dimensional matrix, and such that q is diffeomorphic to $(q_c; \theta)$. Then, we have

$$q = N_c q_c + N_\theta \theta \,, \tag{5.59}$$

with N_c, a constant $n \times (n-1)$ dimensional matrix, and N_θ, a constant n-dimensional column vector, verifying

$$N_c M_c + N_\theta M_\theta = I_{n\times n} \,. \tag{5.60}$$

Letting $\omega_c = M_c q$ and $\varpi_c = \frac{\omega_c}{\omega_\theta}$, one can easily see that

$$(q_c; \theta; \varpi_c; \omega_\theta) \,, \tag{5.61}$$

defines a valid set of coordinates on $\mathcal{TQ}_{s>0}$. The key point is that we keep ω_θ in the coordinates instead of ϖ_θ, which is trivially equal to 1.

Using (5.46), the dynamical equations in the coordinates $(q_c; \theta; \varpi_c; \omega_\theta; t)$ are

$$\begin{cases} \mathring{q}_c = \varpi_c \\ \mathring{\theta} = 1 \\ \mathring{\varpi}_c = \dfrac{1}{\omega_\theta^2}\Big(M_c - \varpi_c M_\theta\Big) l(x,u) \\ \mathring{\omega}_\theta = \dfrac{M_\theta}{\omega_\theta} l(x,u) \\ \mathring{t} = \dfrac{1}{\omega_\theta} \end{cases} \tag{5.62}$$

where x stands for $(q_c, \theta, \varpi_c, \omega_\theta)$ and

$$l(x,u) = D(q)^{-1}[-H(q,\omega) + Bu]\,. \tag{5.63}$$

Note that the dynamical system is made of $n-1$ linear equations and of $n+2$ nonlinear equations. Compared to (5.51), the additional nonlinear equation is $\mathring{\theta} = 1$. Interestingly, this dynamics is trivial. This is the dynamics of the time scale $\tau \sim \theta$ and is the equivalent of the trivial dynamics in the ordinary time t ($\dot{t} = 1$).

5.4.4.3 A Normal Form With a Two-Dimensional Zero Dynamics

Assuming that the decoupling matrix

$$\frac{1}{\omega_\theta^2}\Big(M_c - \varpi_c M_\theta\Big) D^{-1}(q) B \tag{5.64}$$

is invertible, the regular static feedback

$$u = \psi^{-1}(x,v) = \Big(\frac{1}{\omega_\theta^2}\Big(M_c - \varpi_c M_\theta\Big) D^{-1}(q) B)\Big)^{-1} \Big(v + \frac{1}{\omega_\theta^2}[M_c - \varpi_c M_\theta] D^{-1}(q) H(x)\Big) \tag{5.65}$$

transforms the equations (5.62) into

$$\begin{cases} \mathring{q}_c = \varpi_c \\ \mathring{\theta} = 1 \\ \mathring{\varpi}_c = v \\ \mathring{\omega}_\theta = \dfrac{M_\theta}{\omega_\theta^2} l\Big(x, \psi^{-1}(x,v)\Big) \\ \mathring{t} = \dfrac{1}{\omega_\theta} \end{cases} \tag{5.66}$$

As in the coordinates (q, ϖ), the form is made of $n-1$ chains of integrators of dimension two, and of a two-dimensional zero dynamics (ignoring the dynamics of t).

5.4.5 Using the Normal Forms With a Two-Dimensional Zero Dynamics For Motion Planning and Stabilization

We investigate here the potential applications of the normal forms with a two-dimensional zero dynamics (5.51) and (5.66).

5.4.5.1 Motion Planning

Formulating the motion planning problem as a parametric optimization problem under constraints has proven to be useful for the control of biped robots. The HZD method uses the normal form with a two-dimensional zero dynamics (5.51) in the ordinary time t to find motions parameterized by the geometrical parameter $\theta = M_\theta q$ (virtual constraints) (see Section 2.4.3).

With the forms (5.51) or (5.66), one can easily formulate the motion planning algorithm using the same time scale τ in the dynamical equations and for parameterizing the motion. Actually, even though it is another way to proceed, we see no fundamental difference with the approach used in the HZD method. The performances of the motion planning should be exactly identical for $\tau \sim M_\theta q$. Note that the selection of the time scale discriminates the motions that do not respect its strict monotonicity $\frac{dt}{d\tau} > 0$. Then, the obtained trajectory may depend on the choice of τ.

5.4.5.2 Design of Controllers

The normal forms (5.51) and (5.66) are suited for designing controllers stabilizing the controlled coordinates (q_c, ϖ_c). In the ordinary time scale t, an input-output feedback linearizing controller is commonly used (see Section 2.4.7 and [3, 110]). This controller can be trivially generalized to a new time scale τ. It consists in derivating a set of outputs, equal to the number of inputs, until the input u appears, and then using a change of feedback to impose the desired behavior on the output.

Let $\tau \to q_{c,ref}(\tau)$ be a reference trajectory for q_c. We define the vector of outputs

$$y = q_c - q_{c,ref}(\tau)\,. \tag{5.67}$$

The relative degree of this output is two. We have

$$\mathring{\mathring{y}} = v - q''_{c,ref}(\tau)\,. \tag{5.68}$$

Setting

$$v = q''_{c,ref}(\tau) - K_D \mathring{y} - K_P y\,, \tag{5.69}$$

where K_P and K_D are positive definite diagonal matrices, linearizes the dynamics of $\mathring{\mathring{y}}$. The original input u is obtained using the static feedback transformation

(5.50) or (5.65).

This input-output controller is valid as long as the feedback transformation is non singular.

5.5 Dynamic Feedback Equivalence to a Partial Linear System

We introduced previously normal forms made of a $2(n-1)$-dimensional linear and controllable system, and of a two-dimensional zero dynamics (ignoring the dynamics of t). The presented forms were derived using a set of $n-1$ outputs of relative degree two. To decrease the dimension of the zero dynamics, outputs with a higher relative degree must be used (see Appendix G). But, we did not find such outputs using a static feedback transformation.

We derive here a normal form with a one dimensional zero dynamics using a dynamic feedback transformation when the time scale is a linear combination of the positions $\tau \sim \theta = M_\theta q$. For that purpose, from the angular momentum σ, we design an output of relative degree three relative to one input.

5.5.1 Introducing σ in the Coordinates

It is well known that the angular momentum σ forms an output of relative degree three in the actual time scale t, i.e. the input does not appear in the first and second derivatives of σ relative to t, but does in the third derivative. Due to its properties, [48, 127, 84] introduced the angular momentum in the coordinates of mechanical systems to facilitate analysis and feedback design.

We will construct a relative degree three output from the angular momentum σ. Then, as a first step, we introduce here σ in the coordinates of the system.

The expression of σ in the coordinates $(q_c, \theta, \varpi_c, \omega_\theta, t)$ is

$$\sigma = B^\perp D(q)(N_c\omega_c + N_\theta\omega_\theta) = B^\perp D(q)(N_c\varpi_c + N_\theta)\omega_\theta. \tag{5.70}$$

This shows that when

$$\sigma \neq 0\ , \tag{5.71}$$

or equivalently [2] that when

$$B^\perp D(q)(N_c\varpi_c + N_\theta) \neq 0\,, \tag{5.72}$$

σ can be used as a coordinate replacing ω_θ. Note that this is equivalent to say that the sign of σ remains unchanged

$$\epsilon\sigma > 0\,, \tag{5.73}$$

[2] Since $\omega_\theta \neq 0$ by assumption

where $\epsilon = 1$ or $\epsilon = -1$ depending on which subset of the state space the study is performed.

The dynamical equations in the coordinates $(q_c, \theta, \varpi_c, \sigma, t)$ are

$$\begin{cases} \mathring{q}_c = \varpi_c \\ \mathring{\theta} = 1 \\ \mathring{\varpi}_c = v \\ \mathring{\sigma} = -\dfrac{B^\perp G(q)}{\sigma}[B^\perp D(q)(N_c\varpi_c + N_\theta)] \\ \mathring{t} = \dfrac{B^\perp D(q)(N_c\varpi_c + N_\theta)}{\sigma} \end{cases} \tag{5.74}$$

Remark: Since only a change of coordinates has been performed, the feedback transformation (5.65) remains unchanged.

5.5.2 Building a Relative Degree Three Output

We saw that the angular momentum σ has a relative degree three using the ordinary time scale t (see Section 5.3.1.1). But, in the time scale $\tau = \theta$, its first derivative is

$$\mathring{\sigma} = -\frac{B^\perp G(q)}{\sigma}\Big(B^\perp D(q)[N_c\varpi_c + N_\theta]\Big)\,. \tag{5.75}$$

Then, due to the presence of ϖ_c, using the expression of $\mathring{\varpi}_c$, the relative degree of σ is only two if

$$\frac{1}{\omega_\theta^2} B^\perp G(q) B^\perp D(q) N_c (M_c - \varpi_c M_\theta) D(q)^{-1} B \neq 0\,. \tag{5.76}$$

Nonetheless, we see here that one can build from σ a relative degree three output y_1 relative to one input component, that we call v_2. We also investigate under what conditions, y_1, its first, second and third derivatives in $\tau \sim \theta$ can replace some coordinates of the system. It is important to verify this latter point, otherwise the output cannot be used to change the shape of the equations.

To define a relative degree three output relative to one input component, we extract [3] a specific component from q_c, called q_{c_2}. We call q_{c_1} the $n-2$ remaining components of q_c. M_c is chosen such that q_{c_2} is the last component of q_c. More specifically, we define

$$M_c = \begin{pmatrix} M_{c_1} \\ M_{c_2} \end{pmatrix} \tag{5.77}$$

[3] As we shall see later, a proper choice of q_{c_2} may influence the domain of validity of the form that we present here.

with $M_{q_{c1}}$, a $(n-2)\times(n-1)$ constant matrix, and $M_{q_{c2}}$, a $(n-1)$-dimensional constant row vector verifying

$$\begin{pmatrix} q_{c_1} \\ q_{c_2} \end{pmatrix} = \begin{pmatrix} M_{c_1} \\ M_{c_2} \end{pmatrix} q_c \,, \tag{5.78}$$

and such that there exist $N_{q_{c1}}$ a $(n-1)\times(n-2)$ matrix and $N_{q_{c2}}$ a $(n-1)$-dimensional column vector satisfying

$$N_{c_1} M_{c_1} + N_{c_2} M_{c_2} = I_{(n-2)\times(n-2)} \,. \tag{5.79}$$

With this, we have

$$q_c = \begin{pmatrix} N_{c_1} & N_{c_2} \end{pmatrix} \begin{pmatrix} q_{c_1} \\ q_{c_2} \end{pmatrix} \,. \tag{5.80}$$

We reorder ϖ_c as

$$\begin{pmatrix} \varpi_{c_1} \\ \varpi_{c_2} \end{pmatrix} = \begin{pmatrix} M_{c_1} \\ M_{c_2} \end{pmatrix} \varpi_c \,, \tag{5.81}$$

and the input v as

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} M_{c_1} \\ M_{c_2} \end{pmatrix} v \,. \tag{5.82}$$

$(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}, \varpi_{c_2}, \sigma, t)$ defines a new set of coordinates. In these coordinates, the expression of $\mathring{\sigma}$ becomes

$$\mathring{\sigma} = -\frac{B^\perp G(q)}{\sigma}[B^\perp D(q)\Big(N_c(N_{c_1}\varpi_{c_1} + N_{c_2}\varpi_{c_2}) + N_\theta\Big)] \,, \tag{5.83}$$

or equivalently, since σ was assumed to be non zero (see (5.71)),

$$\mathring{\frac{\sigma^2}{2}} = -B^\perp G(q)[B^\perp D(q)(N_c(N_{c_1}\varpi_{c_1} + N_{c_2}\varpi_{c_2}) + N_\theta)] \,. \tag{5.84}$$

Here the argument q stands for $(q_{c_1}, q_{c_2}, \theta)$. Under (5.76), $\frac{\sigma^2}{2}$ is of relative degree two relative to the entire input v. To get a relative degree three output relative to the scalar input [4] v_2, one should remove the term $-(B^\perp G(q) B^\perp D(q) N_c N_{c_2})\varpi_{c_2}$ on the right hand side. For that, we replace $\frac{\sigma^2}{2}$ by the equivalent coordinate

$$y_1 = \frac{\sigma^2}{2} + \int_0^{q_{c_2}} (B^\perp G(q) B^\perp D(q) N_c N_{c_2}) \, \mathrm{d}l \,. \tag{5.85}$$

In this expression, the argument q appearing in the integrand represents (q_{c_1}, l, θ), with l the dummy version of q_{c_2}. Under the assumption that

$$y_1 > \int_0^{q_{c_2}} (B^\perp G(q) B^\perp D(q) N_c N_{c_2}) \, \mathrm{d}l \,, \tag{5.86}$$

[4] Ignoring the dependance to v_1.

and since we assumed that the sign of σ is constant (see (5.73)), we have

$$\sigma = \epsilon\sqrt{2\Big(y_1 - \int_0^{q_{c_2}} (B^\perp G(q) B^\perp D(q) N_c N_{c_2})\,\mathrm{d}l\Big)}\,, \tag{5.87}$$

where $\epsilon = 1$ or $\epsilon = -1$ depending on which subset of the state space the study is performed.

Then, under (5.86), y_1 can be used as a coordinate instead of σ.

The derivative of y_1 relative to τ, called y_2 is

$$\begin{aligned}
y_2 =& \mathring{y}_1 \\
=& - B^\perp G(q)[B^\perp D(q)(N_c(N_{c_1}\varpi_{c_1} + \cancel{N_{c_2}\varpi_{c_2}}) + N_\theta)] + \cancel{B^\perp G(q) B^\perp D(q) N_c N_{c_2}\varpi_{c_2}} \\
&+ \int_0^{q_{c_2}} \left[(N_c N_{c_2})^T \Big(\frac{\partial}{\partial q_{c_1}}\Big(B^\perp G(q) B^\perp D(q)\Big)^T \varpi_{c_1} + \frac{\partial}{\partial \theta}\Big(B^\perp G(q) B^\perp D(q)\Big)^T\Big)\right] \mathrm{d}l \\
=& \quad - (N_c N_{c_1}\varpi_{c_1} + N_\theta)^T (B^\perp G(q) B^\perp D(q))^T \\
&+ \int_0^{q_{c_2}} \left[(N_c N_{c_2})^T \Big(\frac{\partial}{\partial q_{c_1}}\Big(B^\perp G(q) B^\perp D(q)\Big)^T \varpi_{c_1} + \frac{\partial}{\partial \theta}\Big(B^\perp G(q) B^\perp D(q)\Big)^T\Big)\right] \mathrm{d}l \\
=& \quad f_2(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}).
\end{aligned} \tag{5.88}$$

Hence, we do obtain that $\mathring{y}_1$ depends on q_{c_2}, but not on ϖ_{c_2}. Then, the relative degree of y_1 is at least equal to three relative to v_2, since $\mathring{\varpi}_{c_2} = v_2$. Moreover, according to the implicit function theorem, the coordinate q_{c_2} can be replaced by y_2 if and only if we have

$$\frac{\partial f_2}{\partial q_{c_2}}(q_{c_1}, q_{c_2}, \varpi_{c_1}, \theta) \neq 0\,, \tag{5.89}$$

with

$$\begin{aligned}
&\frac{\partial f_2}{\partial q_{c_2}}(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}) = \\
&\quad - (N_c N_{c_1}\varpi_{c_1} + N_\theta)^T \frac{\partial}{\partial q_{c_2}}(B^\perp G(q) B^\perp D(q))^T \\
&\quad + (N_c N_{c_2})^T \Big(\frac{\partial}{\partial q_{c_1}}\Big(B^\perp G(q) B^\perp D(q)\Big)^T \varpi_{c_1} + \frac{\partial}{\partial \theta}\Big(B^\perp G(q) B^\perp D(q)\Big)^T\Big)\,.
\end{aligned} \tag{5.90}$$

Then, under this assumption, we work now with the coordinates $(q_{c_1}; \theta; \varpi_{c_1}; \varpi_{c_2}; y_1; y_2; t)$.

The derivative of y_2 with respect to τ, called y_3, is

$$\begin{aligned}
y_3 = \mathring{\mathring{y}}_1 = \mathring{y}_2 &= \frac{\partial f_2}{\partial q_{c_1}}\varpi_{c_1} + \frac{\partial f_2}{\partial q_{c_2}}\varpi_{c_2} + \frac{\partial f_2}{\partial \theta} + \frac{\partial f_2}{\partial \varpi_{c_1}} v_1 \\
&= f_3(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}, \varpi_{c_2}, v_1)\,.
\end{aligned} \tag{5.91}$$

If

$$\frac{\partial f_2}{\partial q_{c_2}} \neq 0\,, \tag{5.92}$$

then ϖ_{c_2} can be replaced by y_3

$$\varpi_{c_2} = \frac{1}{\frac{\partial f_2}{\partial q_{c_2}}}\Big(y_3 - \frac{\partial f_2}{\partial \varpi_{c_1}}\varpi_{c_1} - \frac{\partial f_2}{\partial \varpi_{c_1}}v_1 - \frac{\partial f_2}{\partial \theta}\Big)\,. \tag{5.93}$$

But this involves the inputs v_1. Then, to properly define the substation of ϖ_{c_2} with y_3 in the coordinates, we need to add v_1 in the state. This comes to perform a dynamic extension by adding an integrator to v_{c_1}.

5.5.3 Performing a Dynamic Extension

To cope with the fact that v_1 appears in $\mathring{\mathring{y}}_1$, we perform a dynamic extension by adding the $(n-2)$-dimensional vector of inputs v_1 to the state components, and we consider its derivative $\mathrm{w}_1 = \mathring{v}_1$ as the new input. Then, we define the dynamic extension as

$$\mathrm{w} = \begin{pmatrix} \mathrm{w}_1 \\ \mathrm{w}_2 \end{pmatrix} = \begin{pmatrix} \mathring{v}_1 \\ v_2 \end{pmatrix}\,. \tag{5.94}$$

The extended dynamical system is of dimension $(2n+1)+(n-2) = 3n-1$. The coordinates are $(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}; \varpi_{c_2}, v_1, t)$. It has $(n-1)$ inputs w.

Under (5.92), ϖ_{c_2} can be replaced by y_3. Hence, the coordinates $(q_{c_1}; \theta; \varpi_{c_1}; y_1; y_2; y_3; v_1; t)$ define a new set of coordinates for the extended dynamical system. The derivative of y_3 relative to τ gives

$$\begin{aligned}\mathring{\mathring{\mathring{y}}}_1 = \mathring{y}_3 = {} & \frac{\partial f_3}{\partial q_{c_1}}\varpi_{c_1} + \frac{\partial f_3}{\partial q_{c_2}}\varpi_{c_2} + \frac{\partial f_3}{\partial \theta} + \frac{\partial f_3}{\partial \varpi_{q_{c_1}}}v_1 + \frac{\partial f_3}{\partial \varpi_{q_{c_2}}}\mathrm{w}_2 \\ & + \frac{\partial f_3}{\partial v_1}\mathrm{w}_1.\end{aligned}$$

And the extended dynamics in these new coordinates are

$$\begin{cases} \mathring{q}_{c_1} = \varpi_c \\ \mathring{\theta} = 1 \\ \mathring{\varpi}_{c_1} = v_{c_1} \\ \mathring{y}_1 = y_2 \\ \mathring{y}_2 = y_3 \\ \mathring{y}_3 = \dfrac{\partial f_3}{\partial q_{c_1}}\varpi_{c_1} + \dfrac{\partial f_3}{\partial q_{c_2}}\varpi_{c_2} + \dfrac{\partial f_3}{\partial \theta} + \dfrac{\partial f_3}{\partial \varpi_{q_{c_1}}}v_1 + \dfrac{\partial f_3}{\partial \varpi_{q_{c_2}}}\mathrm{w}_2 + \dfrac{\partial f_3}{\partial v_{c_1}}\mathrm{w}_1 \\ \mathring{v}_1 = \mathrm{w}_1 \\ \mathring{t} = f_{\mathring{t}}(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1}, \varpi_{c_2}, v_1) \end{cases} \tag{5.95}$$

5.5.4 A Linear and Controllable Transverse Dynamics

The transverse coordinates are $(q_{c_1}, \varpi_{c_1}, y_1, y_2, y_3, v_1)$ and the transverse dynamics are

$$\begin{cases} \mathring{q}_{c_1} = \varpi_{c_1} \\ \mathring{\varpi}_{c_1} = v_1 \\ \mathring{y}_1 = y_2 \\ \mathring{y}_2 = y_3 \\ \mathring{y}_3 = \dfrac{\partial f_3}{\partial q_{c_1}}\varpi_{c_1} + \dfrac{\partial f_3}{\partial q_{c_2}}\varpi_{c_2} + \dfrac{\partial f_3}{\partial \theta} + \dfrac{\partial f_3}{\partial \varpi_{q_{c_1}}}v_1 + \dfrac{\partial f_3}{\partial \varpi_{q_{c_2}}}\mathrm{w}_2 + \dfrac{\partial f_3}{\partial v_1}\mathrm{w}_1 \\ \mathring{v}_1 = \mathrm{w}_1 \end{cases} \tag{5.96}$$

The output

$$z_1 = \begin{pmatrix} q_{c_1} \\ y_1 \end{pmatrix} \tag{5.97}$$

is of vector relative degree $(3, ..., 3)$ with respect to the input w and

$$\mathring{\mathring{\mathring{z}}}_1 = \begin{pmatrix} I_{(n-2)\times(n-2)} & 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial v_{c_1}} & \frac{\partial f_3}{\partial \varpi_{c_2}} \end{pmatrix} \begin{pmatrix} \mathrm{w}_1 \\ \mathrm{w}_2 \end{pmatrix} + \begin{pmatrix} 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial q_{c_1}}\varpi_{c_1} + \frac{\partial f_3}{\partial q_{c_2}}\varpi_{c_2} + \frac{\partial f_3}{\partial \theta} + \frac{\partial f_3}{\partial \varpi_{q_{c_1}}}v_1 \end{pmatrix}. \tag{5.98}$$

On the condition that the decoupling matrix $\begin{pmatrix} I_{(n-2)\times(n-2)} & 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial v_{c_1}} & \frac{\partial f_3}{\partial \varpi_{c_2}} \end{pmatrix}$ is invertible, i.e. $\frac{\partial f_3}{\partial \varpi_{c_2}} \neq 0$, the dynamics of z_1 is rendered linear via the static feedback

$$\begin{aligned} \mathrm{w} = &- \begin{pmatrix} I_{(n-2)\times(n-2)} & 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial v_{c_1}} & \frac{\partial f_3}{\partial \varpi_{c_2}} \end{pmatrix}^{-1} \begin{pmatrix} 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial q_{c_1}}\varpi_{c_1} + \frac{\partial f_3}{\partial q_{c_2}}\varpi_{c_2} + \frac{\partial f_3}{\partial \theta} + \frac{\partial f_3}{\partial \varpi_{q_{c_1}}}v_{c_1} \end{pmatrix} \\ &+ \begin{pmatrix} I_{(n-2)\times(n-2)} & 0_{(n-2)\times 1} \\ \frac{\partial f_3}{\partial v_{c_1}} & \frac{\partial f_3}{\partial \varpi_{c_2}} \end{pmatrix}^{-1} \Gamma. \end{aligned} \tag{5.99}$$

This gives

$$\mathring{\mathring{\mathring{z}}}_1 = \Gamma. \tag{5.100}$$

5.5.4.1 Conditions of Existence of the Normal Form

We recapitulate the conditions required to get the normal form with a linear transverse dynamics on an open subset $\mathcal{V}$ of the state space $\mathcal{TQ}$

- **Condition 1.** The time scaling function is strictly positive

$$\frac{1}{M_\theta \omega} > 0\,.$$

- **Condition 2.** The decoupling matrix

$$\frac{1}{\omega_\theta^2}\Big(M_c - \Big(\frac{M_c\omega}{\omega_\theta}\Big)M_\theta\Big)D^{-1}(q)B$$

is invertible.

- **Condition 3.** The angular momentum

$$\sigma = B^\perp D(q)\omega$$

has a constant sign.

- **Condition 4.** y_1 is such that

$$y_1 - (B^\perp G(q)B^\perp D(q)N_cN_{c_2})\,\mathrm{d}l > 0\,.$$

- **Condition 5.** The scalar

$$-(N_cN_{c_1}\varpi_{c_1} + N_\theta)^T\frac{\partial}{\partial q_{c_2}}(B^\perp G(q)B^\perp D(q))^T + (N_cN_{c_2})^T\Big(\frac{\partial}{\partial q_{c_1}}\Big(B^\perp G(q)B^\perp D(q)\Big)^T\varpi_{c_1} + \frac{\partial}{\partial\theta}\Big(B^\perp G(q)B^\perp D(q)\Big)^T\Big)\,,$$

is non zero and so has a constant sign, with $q_{c_1} = M_{c_1}M_cq$, $q_{c_2} = M_{c_2}M_cq$, $\varpi_{c_1} = \frac{M_{c_1}M_c\omega}{M_\theta\omega}$ and $\theta = M_\theta q$.

Remark 1: The condition 4 is trivially verified when going from the original coordinates (q, ω, t) to the coordinates $(q_{c_1}, \varpi_{c_1}, y_1, y_2, y_3, v_{c_1}, t)$.

Remark 2: The choice of the selection matrices M_c, M_{c_1}, M_{c_2}, N_c, N_{c_1}, N_{c_2}, M_θ, N_θ a priori influences the validity of these conditions on a given set. Then, a careful choice of them must be made.

5.5.5 Main result

We proved the following result:

Theorem 1. *Let a one degree underactuated mechanical system with n degrees of freedom, and such that the mass matrix does not depend on the unactuated variable (cyclic variable). In such a case, the time derivative of the generalized momentum σ associated to the cycle variable is under the form (5.28). The time scale τ is assumed to be a linear combination of the positions, i.e. $\tau \sim M_\theta q$. The dynamical equations of the system are (5.62). Let $\mathcal{V}$ be an open subset of the state space $\mathcal{TQ}$ such that the conditions* **(C1)**, **(C2)**, **(C3)**, **(C4)** *hold. Then, the $(2n+1)$-dimensional dynamical system (5.62) is dynamically feedback equivalent to the $(3n-1)$-dimensional dynamical system expressed in the coordinates $(z_1, z_2, z_3, \theta, t)$ on $\mathcal{V}$*

$$\begin{cases} \mathring{z}_1 = z_2 \\ \mathring{z}_2 = z_3 \\ \mathring{z}_3 = \Gamma \\ \mathring{\theta} = 1 \\ \mathring{t} = f_{\dot{t}}(z_1, z_2, z_3, \theta) \end{cases} \tag{5.101}$$

Γ is the $(n-1)$-dimensional vector of inputs and $f_{\dot{t}}$ is a non linear function of the state. (z_1, z_2, z_3) are called the transverse coordinates.

Remark: The complete dynamics is made of a $3(n-1)$-dimensional linear system and a one-dimensional trivial zero dynamics (ignoring the dynamics of t). Ignoring t and θ, i.e. keeping only the transverse coordinates, we get a linear and controllable system. Actually, the form obtained when ignoring t corresponds to what [69] call a *time-augmented linear system*, except that this form is obtained using a dynamic, and not a static feedback transformation.

5.5.5.1 Related Works

Actually, the derivation and the obtained normal form are similar to what is presented in [48] (section III.C). But, we see three differences. First, we work in a new time scale. The time scale that we use does not depend explicitly on the ordinary time t. Then, t cannot be even taken as a particular case in our approach. Secondly, the relative degree three output that we introduce is different than those previously used in the literature. This means that we use a new set of coordinates. Thirdly, the form we exhibit is suited for working with virtual constraints. Indeed, the form in [48] can be used for tracking an equilibrium point or time parameterized trajectories. But, it cannot be used for virtual constraints since the output formed of the difference between the relative degree three output and a virtual constraint parameterized by the phase variable θ is only of relative degree two. The second derivative of θ is actually a function of the input. On the contrary, since in our case $\tau \sim \theta$, our form is suited for tracking trajectories parameterized by θ, but not for tracking trajectories parameterized by the ordinary time t.

An other point is that the *transverse dynamics* of the normal form that we introduce is a linear and controllable system. Let us define the term of *transverse dynamics*. For a system whose trajectories can be parameterized by a scalar coordinate θ, the coordinates of the system, excluding the coordinate θ (and implicitly the time t), are called the *transverse coordinates*. Their dynamics are called *transverse dynamics* [7]. Generalizing this definition to dynamical systems written in a time scale $\tau \sim \theta$, we call *transverse coordinates* the coordinates of the system removing the time scales θ and t, and *transverse dynamics* the dynamics of these coordinates.

The transverse dynamics have already been used to stabilize one degree under-actuated mechanical systems like point-foot robots. But, an approximate linearized

transverse dynamics along the motion of reference is used. This method is called transverse linearization [41, 72]. On the contrary, the form that we propose is exact and independent of any motion of reference. It can then be used both for planning or stabilization applications.

5.5.5.2 Using the Normal Form for Motion Planning

A motion algorithm formulated as an optimization problem under constraints as used in the HZD method (see Section 2.4.6.1) can be used. The interest of using the normal form (5.101) is that the zero dynamics is of dimension one and is trivial ($\mathring{\theta} = 1$). Then, contrary to the case where (5.51) or (5.66) are used, there is no need to integrate the zero dynamics and to check its stability. From the knowledge of $\tau \rightarrow z_1(\tau)$, the coordinates (z_1, z_2, z_3) are easily deduced by derivations.

In others words, the transverse dynamics of the system is flat [40].Then, if one chooses to take $\tau \rightarrow z_1(\tau)$ under the form of polynomial functions, the coefficients of the polynomials, and the initial condition on θ, are the degrees of freedom of the motion planning algorithm. In some cases, the property of flatness allows, by an appropriate hand tuning of $z_1(\tau)$, to easily make sure that the constraints during the motion are respected. That's what makes the flatness theory so powerful, since it allows to trivialize the motion planning problem for some systems (see the examples in [40]).

But, in our case, the physical meaning of z_1 is abstruse, which makes difficult the formulation of kinematic constraints such as the step length or the maximum altitude of the foot during swing. Additionally, we did not manage to express some constraints, like the no take off and no slipping conditions, in the coordinates $(z_1, z_2, z_3, \theta, t)$. Then, we have to come back to the original coordinates (q, ω) to evaluate the constraint. As a consequence, we cannot take advantage of the flatness of the transverse dynamics. Note also that the dynamics of t must be integrated if one is interested in knowing the duration of the motion.

We implemented the previously described motion planning algorithm to find walking cycles for the five-link walker described in 4.4.1. We encountered singularity problems when choosing $M_{q_{c_2}} = (0\,0\,0\,1)$ and $M_{q_{c_1}} = (I_{3\times3}\,0_{3\times1})$. We did not test an other choice of selection matrices, since we noticed that coming back to the original coordinates was a major drawback[5]. Hence, the use of the normal form with a two-dimensional zero dynamics (5.66) is easier and leads to a faster motion planning algorithm.

5.5.5.3 Using the Normal Form for Stabilization

The normal form with a one-dimensional zero dynamics (5.101) may be powerful for stabilization, since the dynamics of all the state components is linear and controllable, except θ. The design must take into account the constraints on the state gathered in Section 5.5.4.1.

Contrary to the case of motion planning, the physical meaning of the coordinates (z_1, z_2, z_3) has little importance, since the motion is already computed.

[5]It requires to solve the implicit function $y_2 = f_2(q_{c_1}, q_{c_2}, \theta, \varpi_{c_1})$. This is time consuming.

5.6 On The Design of a Relative Degree Four Output From the Angular Momentum σ in a New Time Scale

In the previous sections, we derived normal forms using outputs with a relative degree two or three. Actually, using one output of relative degree four and $(n-2)$ outputs of relative degree two would result in a normal form with an empty zero dynamics, that is an exact linear and controllable form (see Appendix G). [102] designed a time scale in which the antiderivative of the angular momentum is a relative degree four output for the compass walker. We explain here that there probably exists no time scale such that this property can be generalized to the others planar one degree underactuated biped robots. This is the only result that we could exhibit on the equivalence of the dynamics of one degree undercactuated biped robots with point feet to a normal form with an empty zero dynamics.

5.6.1 On the Relative Degree of σ.

We give here sufficient conditions on the time scaling function $s(q,\omega)$ so that the angular momentum be of relative degree 3. From Section 5.3.1.1, the derivative of the angular momentum relative to τ is

$$\mathring{\sigma} = -B^{\perp}G(q)s(q,\omega)\,, \tag{5.102}$$

$$B^{\perp} = \begin{pmatrix} 0_{1\times(n-1)} & 1 \end{pmatrix}. \tag{5.103}$$

Then

$$\mathring{\mathring{\sigma}} = -B^{\perp}\frac{\partial G}{\partial q}(q)s^2(q,\omega)\omega - B^{\perp}G(q)\Big[\frac{\partial s}{\partial q}(q,\omega)s(q,\omega)\omega + \frac{\partial s}{\partial \omega}(q,\omega)s(q,\omega)\dot{\omega}\Big]. \tag{5.104}$$

Since $s(q;\omega) > 0$ by hypothesis, and $B^{\perp}G(q)$ is generally non zero, the necessary and sufficient condition so that the relative degree of σ be at least 3 is

$$\frac{\partial s}{\partial \omega}(q,\omega)D^{-1}(q)B = 0_{1\times m}. \tag{5.105}$$

Then, a sufficient condition is

$$\frac{\partial s}{\partial \omega}(q,\omega) = 0_{1\times n}\,, \tag{5.106}$$

that is s does not depend on the velocity. Under this assumption, we have

$$\mathring{\mathring{\sigma}} = -B^{\perp}\Big[\frac{\partial G}{\partial q}(q)s(q) + G(q)\frac{\partial s}{\partial q}\Big]s(q)\omega\,. \tag{5.107}$$

Hence, σ is of relative degree 3 when s depends on the position only[6].

[6]Except for the particular case $s(q) = \frac{\lambda}{B^{\perp}G(q)}$, $\lambda \in \mathbb{R}$, where $\mathring{\mathring{\sigma}} = 0$.

5.6.2 Conditions on the Existence of an Antiderivative of σ

We just saw that the relative degree of σ is equal to 3 when the time scaling function depends on the position only. We investigate here the conditions that the time scaling function $s(q)$ must verify so that the antiderivative of σ may exist. In this case, the antiderivative of σ would allow define a relative degree four output.

Let $f(q)$ be an antiderivative of σ. By definition, it must verify

$$\frac{d(f(q))}{d\tau} = s(q,\omega)\frac{\partial f}{\partial q}(q)\omega = B^{\perp}D(q)\omega. \tag{5.108}$$

Equivalently, we have

$$\frac{1}{s(q)}M(q) = \frac{\partial f}{\partial q}(q)\,, \tag{5.109}$$

with $M(q) = B^{\perp}D(q)$.

We denote the n-dimensional row vector $M(q)$ as

$$M(q) = (M_1(q)\ M_2(q)\ ...\ M_n(q))\,. \tag{5.110}$$

Note that from (5.103), $M(q)$ is the last row of the mass matrix $D(q)$. Then, $M_n(q) \neq 0$.

(5.109) corresponds to the set of n equations

$$\frac{1}{s(q)}M_i(q) = \frac{\partial f}{\partial q_i}(q)\,, i \in \{1,n\}\,. \tag{5.111}$$

From a standard theorem on differential forms, sometimes called the Poincaré's lemma, (5.111) has a solution if and only if the equality of mixed partials is verified, that is

$$\frac{\partial^2 f}{\partial q_i \partial q_j}(q) = \frac{\partial^2 f}{\partial q_j \partial q_i}(q)\,, (i,j) \in \{1,n\}^2,\ i \neq j\,. \tag{5.112}$$

Derivating (5.109) gives the set of $\frac{n(n-1)}{2}$ equations

$$\frac{\partial M_j}{\partial q_i}(q) - \frac{\partial M_i}{\partial q_j}(q) = M_j(q)\mu_i - M_i(q)\mu_j\,, (i,j) \in \{1,n\}^2,\ i < j\,, \tag{5.113}$$

where $\mu_i = \frac{1}{s(q)}\frac{\partial s}{\partial q_i}(q)$.

The problem consists now in finding if there exist a set of μ_i which satisfy (5.113).

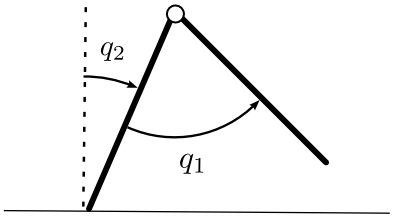


Figure 5.2: The planar two-link (or compass) walker.

5.6.3 Dimension 2

In dimension two, the biped model is the compass walker depicted in Figure 5.2. A description of this model and the corresponding equations can be found in [127] (page 67). Since q_2 is a cyclic coordinate, $\frac{\partial M_1}{\partial q_2}(q) = 0$. Then, (5.113) gives

$$\frac{\partial M_2}{\partial q_1}(q) = M_2(q)\mu_1 - M_1(q)\mu_2\,. \tag{5.114}$$

Since $M_2(q) \neq 0$

$$\begin{cases} \mu_1 = \dfrac{1}{M_2(q)}\dfrac{\partial M_2(q)}{\partial q_1} \\ \mu_2 = 0 \end{cases} \tag{5.115}$$

is a solution of (5.114). From the definition of μ_1, we have $s(q) = M_2(q)$. Hence, an antiderivative of σ exists for this choice of $s(q)$.

The expression of the antiderivative is obtained by solving (5.109)

$$\begin{cases} \dfrac{\partial f}{\partial q_1} = \dfrac{M_1}{M_2} \\ \dfrac{\partial f}{\partial q_2} = 1 \end{cases} \tag{5.116}$$

Hence

$$f(q) = q_2 + \int_0^{q_1} \frac{M_1}{M_2}(l)\,dl\,. \tag{5.117}$$

This is the same integral primitive as found in [102].

5.6.4 Dimension 3

5.6.4.1 The three-link planar walker.

The three-dimensional case corresponds to the three-link planar walker (a compass walker with an additional torso) depicted figure 5.3. A description of this model and the corresponding equations can be found in [127] (page 64). We have

$$\begin{cases} M_1 = a + b\cos(q_1 - q_2) + c\cos(q_1) \\ M_2 = d + b\cos(q_1 - q_2) \\ M_3 = 2c\cos(q_1) + b\cos(q_1 - q_2) + e \end{cases} \tag{5.118}$$

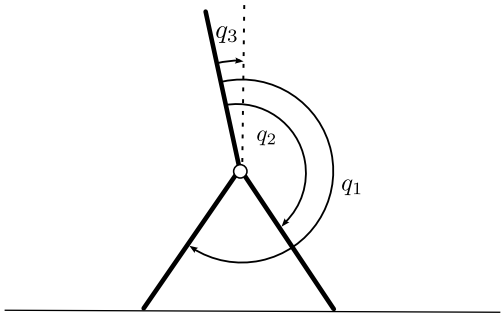


Figure 5.3: The planar three-link walker.

where a, b, c, d, e are strictly positive constants that depend only of the dynamical parameters (length, mass and inertia) of the model.

(5.113) gives the three following equations

$$\begin{cases} (i,j) = (1,2): & \dfrac{\partial M_2}{\partial q_1} - \dfrac{\partial M_1}{\partial q_2} = M_2\mu_1 - M_1\mu_2 \\ (i,j) = (1,3): & \dfrac{\partial M_3}{\partial q_1} = M_3\mu_1 - M_1\mu_3 \\ (i,j) = (2,3): & \dfrac{\partial M_3}{\partial q_2} = M_3\mu_2 - M_2\mu_3 \end{cases} \tag{5.119}$$

Or written under a matrix form

$$\begin{pmatrix} \frac{\partial M_2}{\partial q_1} - \frac{\partial M_1}{\partial q_2} \\ \frac{\partial M_3}{\partial q_1} \\ \frac{\partial M_3}{\partial q_2} \end{pmatrix} = \begin{pmatrix} M_2 & -M_1 & 0 \\ M_3 & 0 & -M_1 \\ 0 & M_3 & -M_2 \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} . \tag{5.120}$$

One remarks that

$$\begin{pmatrix} -M_3 & M_2 & -M_1 \end{pmatrix} \begin{pmatrix} M_2 & -M_1 & 0 \\ M_3 & 0 & -M_1 \\ 0 & M_3 & -M_2 \end{pmatrix} = 0_{1\times 3} . \tag{5.121}$$

Then, a necessary condition so that solutions μ_i exist is

$$\begin{pmatrix} -M_3 & M_2 & -M_1 \end{pmatrix} \begin{pmatrix} \frac{\partial M_2}{\partial q_1} - \frac{\partial M_1}{\partial q_2} \\ \frac{\partial M_3}{\partial q_1} \\ \frac{\partial M_3}{\partial q_2} \end{pmatrix} \tag{5.122}$$

or

$$\frac{\partial(\frac{M_2}{M_3})}{\partial q_1} = \frac{\partial(\frac{M_1}{M_3})}{\partial q_2} . \tag{5.123}$$

This is a sufficient condition, since when this condition is verified and choosing $s(q) = \frac{1}{M_3(q)}$, (5.111) gives

$$\frac{\partial f}{\partial q_1} = \frac{M_1}{M_3}\,, \frac{\partial f}{\partial q_2} = \frac{M_2}{M_3}\,, \frac{\partial f}{\partial q_3} = 1\,. \tag{5.124}$$

This set of equations have a solution from the Poincaré's theorem.

Using the dynamic equations (5.118), it appears that the condition (5.123) cannot be verified in the neighborhood of any point of the configuration space. Hence, the antiderivative of the angular momentum does not exist for the three-link walker.

5.6.5 Dimension 4 and Higher Dimensions

In dimension 4 (four-link walker), (5.113) gives

$$\begin{pmatrix} \frac{\partial M_2}{\partial q_1} - \frac{\partial M_1}{\partial q_2} \\ \frac{\partial M_3}{\partial q_1} - \frac{\partial M_1}{\partial q_3} \\ \frac{\partial M_4}{\partial q_1} \\ \frac{\partial M_3}{\partial q_2} - \frac{\partial M_2}{\partial q_3} \\ \frac{\partial M_4}{\partial q_2} \\ \frac{\partial M_4}{\partial q_3} \end{pmatrix} = \begin{pmatrix} M_2 & -M_1 & 0 & 0 \\ M_3 & 0 & -M_1 & 0 \\ M_4 & 0 & 0 & -M_1 \\ 0 & M_3 & -M_2 & 0 \\ 0 & M_4 & 0 & -M_2 \\ 0 & 0 & M_4 & -M_3 \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{pmatrix}. \tag{5.125}$$

Since we have

$$\begin{pmatrix} 0 & 0 & 0 & -M_4 & M_3 & -M_2 \\ 0 & -M_4 & M_3 & 0 & 0 & -M_1 \\ -M_4 & 0 & M_2 & 0 & -M_1 & 0 \end{pmatrix} \begin{pmatrix} M_2 & -M_1 & 0 & 0 \\ M_3 & 0 & -M_1 & 0 \\ M_4 & 0 & 0 & -M_1 \\ 0 & M_3 & -M_2 & 0 \\ 0 & M_4 & 0 & -M_2 \\ 0 & 0 & M_4 & -M_3 \end{pmatrix} = 0_{3\times 6}\,, \tag{5.126}$$

then

$$\begin{pmatrix} 0 & 0 & 0 & -M_4 & M_3 & -M_2 \\ 0 & -M_4 & M_3 & 0 & 0 & -M_1 \\ -M_4 & 0 & M_2 & 0 & -M_1 & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial M_2}{\partial q_1} - \frac{\partial M_1}{\partial q_2} \\ \frac{\partial M_3}{\partial q_1} - \frac{\partial M_1}{\partial q_3} \\ \frac{\partial M_4}{\partial q_1} \\ \frac{\partial M_3}{\partial q_2} - \frac{\partial M_2}{\partial q_3} \\ \frac{\partial M_4}{\partial q_2} \\ \frac{\partial M_4}{\partial q_3} \end{pmatrix} = 0_{3\times 1}\,. \tag{5.127}$$

This gives

$$\begin{cases} \dfrac{\partial(\frac{M_1}{M_4})}{\partial q_3} = \dfrac{\partial(\frac{M_3}{M_4})}{\partial q_1} \\ \dfrac{\partial(\frac{M_2}{M_4})}{\partial q_1} = \dfrac{\partial(\frac{M_1}{M_4})}{\partial q_2} \\ \dfrac{\partial(\frac{M_2}{M_4})}{\partial q_3} = \dfrac{\partial(\frac{M_3}{M_4})}{\partial q_2} \end{cases} \tag{5.128}$$

We did not derive the dynamical equations of the four-link walker. Nonetheless, it can be expected that, as for the three-link walker, these conditions are never met. For higher dimensions, a similar derivation can be performed, and it is also likely that the resulting conditions are never met.

5.7 Linearization Along a Reference Trajectory

The previous section illustrates the difficulty of obtaining a full exact linear form for one degree underactuated robots. Instead of attempting to get such a form, a standard approach consists in deriving an approximative linear form. To stabilize a reference trajectory, the dynamics of the system is linearized along the reference trajectory (first-order approximation) and a feedback controller is designed from the obtained approximative dynamics. More especially, let us consider a dynamical system in the time scale τ

$$\begin{cases} \mathring{x} = f(x, u, \tau) \\ \mathring{t} = s(x)\,. \end{cases} \tag{5.129}$$

Let $\tau \to \Big(x_{ref}(\tau), t_{ref}(\tau), u_{ref}(\tau)\Big)$ be a reference trajectory with the corresponding input, parameterized in the same time scale τ. The first-order approximation of the dynamics along the reference is

$$\begin{cases} \mathring{\widehat{\delta x}} = A(\tau)\delta x + B(\tau)\delta u \\ \mathring{\widehat{\delta t}} = \dfrac{\partial s}{\partial x}(x_{ref}(\tau))\,, \end{cases} \tag{5.130}$$

where $\delta x = x - x_{ref}(\tau)$, $\delta u = u - u_{ref}(\tau)$, $\delta t = t - t_{ref}(\tau)$, $A(\tau) = \frac{\partial f}{\partial x}\Big(x_{ref}(\tau), u_{ref}(\tau), \tau\Big)$, and $B(\tau) = \frac{\partial f}{\partial u}\Big(x_{ref}(\tau), u_{ref}(\tau), \tau\Big)$.

The obtained normal form is a τ-varying linear system. Then, using this form naturally leads to τ-varying feedback controllers. We propose here to design a Linear Quadratic Regulator (LQR) to stabilize a walking trajectory.

5.7.1 Designing a Linear Quadratic Regulator (LQR) For Biped Robots

5.7.1.1 Linear Quadratic Control

A complete description of the linear quadratic control can be found in [14]. Note that the choice of the time scale τ does not modify the way of solving this problem, as long as the time scale is the same as the one used for parameterizing the reference. We consider that the reference is defined over a finite time interval: $\tau \in [\tau^+, \tau^-]$, describing for instance a step that the robot must realize. Then, we solve here a finite horizon LQR problem.

Since stabilizing the instants for which the robot reaches a given configuration has little interest in the case of walking, we ignore the dynamics of δt. Only the errors on the state are stabilized, that is we consider the following first-oder approximation

$$\mathring{\widehat{\delta x}} = A(\tau)\delta x + B(\tau)\delta u\,. \tag{5.131}$$

We search to minimize the cost

$$J = \frac{1}{2}\Big(\delta x^-\Big)^T S_f\Big(\delta x^-\Big) + \frac{1}{2}\int_{\tau^+}^{\tau^-} \Big(\delta x^T R \delta x + \delta u^T Q \delta u\Big)\,\mathrm{d}\tau\,, \tag{5.132}$$

where R et S_f are symmetric positive matrices, Q is a symmetric definite positive matrix and τ^+ (resp. τ^-) corresponds to the value of τ at the beginning (resp. end) of the motion of reference. Q corresponds to the weights on the control effort and R on the tracking error. The final weight S_f increases the importance of having low tracking errors at the end of the motion.

The optimal control law is

$$\delta u = -Q^{-1}B^T(\tau)S(\tau)\delta x\,, \tag{5.133}$$

where $S(\tau)$ is the solution of a Riccati differential equation

$$\begin{cases} \dfrac{d}{d\tau}S(\tau) = -S(\tau)A(\tau) + S(\tau)B(\tau)Q^{-1}B^T(\tau)S(\tau) - R - A^T(\tau)S(\tau) \\ \quad S(\tau) = S_f \end{cases} \tag{5.134}$$

Note that, as it is formulated, the LQR does not seek to minimize the input u, but the deviation relative to the reference on the input $\delta u = u - u_{ref}(\tau)$. This choice is motivated by the fact that the ground reaction forces exerted on a biped robot are a function of u (see (1.31)). Then, keeping the input u close to its reference diminishes the risk of violating the contact constraints [7] (no take off and no slipping of the stance leg).

[7] Assuming that the reference trajectory $(x_{ref}(\tau), u_{ref}(\tau))$ respects these constraints.

5.7.1.2 Choice of the Time Scale

Generally, the time scale used for LQ control is the ordinary time, that is $\tau = t$. Nonetheless, a t-varying controller $u = \phi(x, t)$ is not always desirable, especially for the control of biped robots. Indeed, the underactuated dynamics may evolve slower or faster than expected by the motion planner, and a t-dependent controller may result in poor performances in such cases (see Section 2.4.5).

The use of the first-order approximation using a time scale τ independent of t, that is depending only on the state x, $\tau = \tau(x)$, can be used to derive a state feedback controller to stabilize the motion of reference.

We saw in 5.4.4 that a natural choice of time scale τ is a phase variable θ expressed as a linear combination of the generalized positions: $\tau \sim \theta = M_\theta q$. The reference is parameteried by the phase variable, i.e. $\theta \to x_{ref}(\theta)$, where $\theta \in [\theta^+, \theta^-]$, $-\infty < \theta^+ < \theta^- < +\infty$. Then, using (5.55), the reference expressed in the time scale τ is

$$\tau \to x_{ref}(\tau - \tau_n + \theta_n) \tag{5.135}$$

where τ_n and θ_n are constants updated at the beginning of each step. If $\tau - \tau_n + \theta_n$ goes out of the nominal bounds $[\theta^+, \theta^-]$, it is saturated at the bounds.

5.7.1.3 Choice of the Normal Form

We choose to use the normal form (5.74), that is

$$\begin{cases} \mathring{q}_c = \varpi_c \\ \mathring{\theta} = 1 \\ \mathring{\varpi}_c = v \\ \mathring{\sigma} = -\dfrac{B^\perp G(q)}{\sigma}[B^\perp D(q)(N_c \varpi_c + N_\theta)] \end{cases} \tag{5.136}$$

to derive the first-order approximation (5.131).

This normal form is chosen since it is partially feedback linearized. Then, the feedback linearized components in the first-order approximation are exact and not approximative. Only the dynamics of σ is approximative. Also, it includes the angular momentum σ. This quantity has a strong physical meaning and plays an important role in biped locomotion (see Section 5.3.1.1).

5.7.1.4 Differences Between LQ Control and the HZD Method

The HZD method and the LQR provide two different types of state feedback controllers. In the HZD method (see Section 2.4.4.2), the coordinates $(q_c; \varpi_c)$ are controlled to stabilize the system into a two-dimensional manifold: the hybrid zero dynamics (parameterized by $(\theta; \sigma)$). The reference for these state components is chosen such that the hybrid zero dynamics asymptotically converges to a periodic orbit.

On the contrary, the LQR stabilizes one more coordinate: σ. Then, contrary to the HZD method, the LQR stabilizes the full periodic orbit. There is no hybrid zero

dynamics anymore and so no requirement on the existence of an asymptotically stable periodic orbit into this submanifold. Additionally, such a controller allows to cope with the peaking phenomenon (see Section 2.4.7.3).

5.7.2 Application to the Planar Five-Link Walker

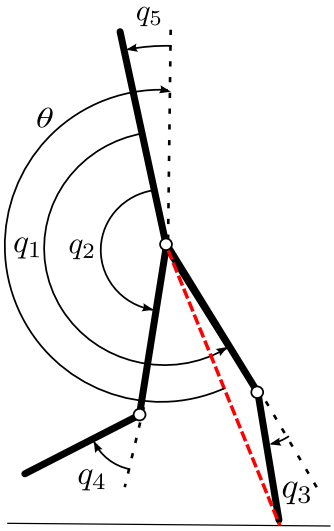


Figure 5.4: The planar five link walker. In the HZD method, the phase variable θ is the virtual stance leg orientation relative to the vertical. The virtual stance leg is the red dotted line.

We illustrate the previous derivation on the planar-five link walker described in 4.4.1 and depicted in figure 5.4. We compare the performances of the LQR with a high gain PD controller used to track the virtual constraints defined for q_c. This last controller is used in the HZD method to constrain the system to evolve in the hybrid zero dynamics (see Section 2.4.7). We remind (see (2.59)) that the expression of the high gain PD controller is

$$u = -K_P y - K_D \dot{y}\,, \tag{5.137}$$

where $y = q_c - q_{c,ref}(\theta)$ and $q_{c,ref}(\theta)$ is the motion of reference for q_c.

The walking trajectory to stabilize was designed using the motion planning algorithm of the HZD method (see Section 2.4.6). Then, an asymptotically periodic orbit exists in the hybrid zero dynamics.

5.7.2.1 Choice of Coordinates and Time Scale

We remind that the planar five-link walker has four actuated degrees of freedom: q_1, q_2, q_3 and q_4. For this specific example, we have

$$x = (q_1; q_2; q_3; q_4; q_5; \varpi_1; \varpi_2; \varpi_3; \varpi_4; \sigma)\,, \tag{5.138}$$

$$q_c = (q_1; q_2; q_3; q_4)\,, \tag{5.139}$$

$$\varpi_c = (\varpi_1; \varpi_2; \varpi_3; \varpi_4)\,, \tag{5.140}$$

$$N_c = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & -\frac{1}{2} & 0 \end{pmatrix},\ N_\theta = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{pmatrix},\ B^\perp = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{5.141}$$

The time scale τ is defined as

$$\frac{dt}{d\tau} = \dot{\theta}\,, \tag{5.142}$$

where θ is the standard phase variable used in the HZD method for the planar five-link walker, that is the virtual stance leg angle (see figure 5.4). θ is a linear combination of the positions $q = (q_1, q_2, q_3, q_4, q_5)$

$$\theta = M_\theta q = -q_1 - \frac{q_3}{2} - q_5\,. \tag{5.143}$$

Then, we have $\tau \sim \theta$.

5.7.2.2 Details on the implementation.

The expressions of $A(\tau)$ and $B(\tau)$ are analytically computed. The state x_{ref} and the input u_{ref} corresponding to one step[8] of the walking trajectory are interpolated by a 6 degree Bézier polynomial in θ. The Riccati equation (5.134) is solved backwards in the time scale τ using the solver *ode45* of Matlab. The resolution takes about one second on a laptop with an 1.3 GHz Intel core i5. The expression of the solution S is obtained on a discretized interval $[\theta^+, \theta^-]$. It is stored in an array. During the simulation, the expression of S corresponding to the closest value of θ is taken from the array to compute the control law

$$v = v_{ref}(\theta) - Q^{-1}B^T(\theta)S(\theta)\delta x\,. \tag{5.144}$$

The vector of the joint torques u is then obtained using the static feedback transformation (5.65).

An issue to be addressed is the action to perform when θ is out of its nominal interval of evolution $[\theta^+, \theta^-]$. For such values, no reference trajectory is defined. We simply decided to saturate θ at its bounds for the computation of S and of

[8]Since the walk is symmetric, the motion of reference is deduced by swapping the role of the legs.

the reference position. The reference velocity is set to zero. This proved to be acceptable, even though it sometimes cause torque peaks at the beginning or at the end of a step. Torques are saturated at 100 Nm.

5.7.2.3 Simulation results.

We considered five scenarios to compare the controllers:

1. Walking on flat ground without disturbances.
2. Walking on uneven ground. The altitude of the ground is varied randomly.
3. Walking on flat ground with an horizontal backward push applied on the hip of the robot during the second step.
4. Walking on flat ground with an horizontal forward push applied on the hip of the robot during the second step.
5. Walking on flat ground using a set of dynamical parameters for the design of the controller that is different than the one used for simulating the robot.

Each test is considered successful if the robot manages to walk fifty steps.

Several sets of weighting matrices for the LQR were tested by trial and error. Two sets of them were selected: W1 and W2. They are gathered in table 5.1. W1 corresponds to high weights on the actuated positions q_c and velocities ϖ_c errors and a low gain on the angular momentum error. Then, it correspond to a PD + feedforward controller (see 2.54). W2 has a high weight on the angular momentum error and on the actuated positions q_c. The motivation of such a choice is to get a controller stabilizing the angular momentum without deviating too much from the reference trajectory.

The performances of the high gain PD controller and of the LQR with these two sets of gains are given in table 5.2. Without surprise, the performances of W1 and of the high gain PD controller are very similar. W2 stabilizes better the angular momentum than W1 and the PD controller. It allows a better recovery to backward pushes (scenario 3). Figure 5.5 depicts simulation results for a backward push of 12 N. With W2, the swing leg quickly moves forward to compensate for the loss of angular momentum due to the push. This is a quite human-like behavior. [72] also noticed this behavior for the compass walker. The counterpart is that since joint tracking performance is deteriorated, the swing leg may impact the ground before midstance (the middle of the step), leading to a fall of the robot during the next step. This behavior was especially noticed when the angular momentum (its sign is negative) is inferior to its reference, i.e. when the robot moves forward too quickly. That's why W2 is less robust than the two others controllers on uneven ground and for forward pushes.

The fifth scenario aims at testing the robustness of the controllers to dynamical parameters uncertainties by introducing a parameter mismatch between the design

model and the “actual” model. Changing the dynamical parameters used to design the LQR and to estimate the angular momentum during the simulation while not changing those of the simulated robot showed that the PD controller and the LQR with W1 are robust to dynamical parameter uncertainties. On the contrary, the LQR with W2 is less robust. As a matter of fact, since the angular momentum depends on the dynamical parameters (see (5.23)), the bigger is the mismatch between the dynamical parameters, the greater is the difference between the angular momentum and its reference. Due to large errors, the robot deviate from its nominal trajectory in an attempt to stabilize its angular momentum. The consequence is that it does not impact the ground in the configuration of reference, causing the slippery of the impacting leg or a fall during the next step.

PD
$K_P = diag(3600, 3600, 3600, 3600)$ $K_D = diag(600, 600, 600, 600)$
W1
$R = diag(10^3, 10^3, 10^3, 10^3, 10^2, 10^2, 10^2, 10^2, 10^{-4})$ $S_f = 4R$ $Q = diag(1, 1, 1, 1)$
W2
$R = diag(5.10^4, 5.10^4, 5.10^4, 5.10^4, 10^{-1}, 10^{-1}, 10^{-1}, 10^{-1}, 100)$ $S_f = 4R$ $Q = diag(1, 1, 1, 1)$

Table 5.1: Gains of the high gain PD controller on the actuated coordinates (q_1, q_2, q_3, q_4) and the two sets of weights used for the LQR. *diag* stands for diagonal matrix. W1 corresponds to high gains on the actuated coordinates errors and offers similar performances to the high gain PD controller. A higher gain on the angular momentum error is set on W2. Gains on the position error are high too, so that the robot may not deviate too much from the reference walking trajectory. Higher weights are used for the final weight tracking errors matrix S_f since we consider that it is important for the robot to be close to the reference trajectory at the instant of impact. Nonetheless, the weights in S_f were not set too high, otherwise we noticed that torque peaks appear at the end of a step, leading to a risk of slipping of the stance leg.

5.7.2.4 Discussion

We are not the first to design a t-invariant LQR. Manchester et al. [72] designed a LQR using the linearization of the transverse dynamics written in the ordinary time t along a trajectory. The resulting optimal gains of the controller are time-dependent. But, to eliminate this time dependence, they evaluate the gains at the "new time" $s = arg_t\, min||x - x^*(t)||$, where x is the state of the robot and $x^*(t)$ is

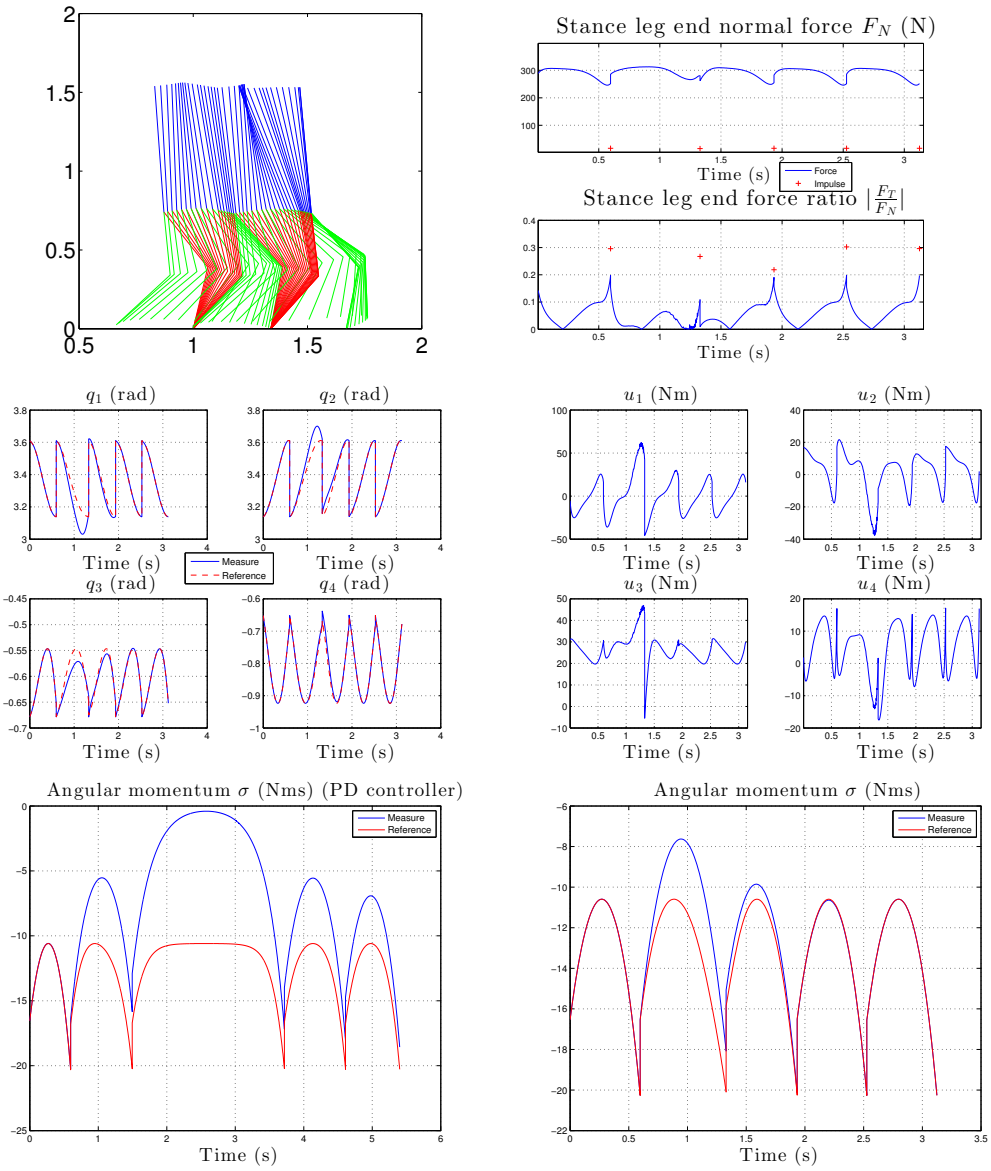


Figure 5.5: Simulation results for the scenario 4 (backward push) using the LQR with W2 (high weight on the angular momentum). The robot is pushed backward by a force of 12 N during the entire second step. The stick animation diagram (above) represents the 2 first steps of the robot (stance leg: red, swing leg: green, torso: blue). The first step is not disturbed. It corresponds to the walking motion. The push is applied during the second step. The robot quickly moves forward its swing leg to increase its angular momentum. See the difference of evolution of the angular momentum using a high gain PD controller (left bottom figure) and using a LQR with W2 (right bottom figure). The LQR better stabilizes the angular momentum. The contact constraints are respected ($F_N > 0$ and $\left|\frac{F_T}{F_N}\right| < 0.6$, idem for the impulses at impact).

	PD	W1	W2
1. Walking on flat ground	Same performances		
2. Walking on uneven ground	± 2.5 cm	± 2.5 cm	± 1.5 cm
3. Magnitude of a push forward	75 N	75 N	6 N
4. Magnitude of a push backward	12 N	8 N	20 N
5. Dynamical parameters uncertainties	-99 % / + 200 %	-99 % / + 200 %	+10 % / - 10 %

Table 5.2: Performances of the high gain PD controller, the LQR using W1 and the LQR using W2 for the five scenarios. For the scenario 2, the terrain is varied randomly. The table contains the maximum allowable deviation of ground altitude. For the scenario 5, the dynamical parameters (mass and inertia of the links) are multiplied by the same scaling factor. The transmission inertia is kept unchanged.

the time parameterized trajectory to track. Even though the obtained results are satisfactory for the compass walker, is maybe not a proper way to do. Especially, the choice of the norm to define s is not obvious. On the contrary, our approach simply and properly defines an (ordinary time)-invariant LQR without requiring any projection. But, contrary to [72], our approach is only valid when the time scale τ does not go backward in time.

Improvements to get a better stabilization of the angular momentum while avoiding too early impacts are an automatic search of the weighting matrices by solving an optimization problem taking into account various disturbances as in [45]. An other idea could be to control directly the swing leg altitude, by using for example coordinates that include this quantity. .

The LQR could be tested to stabilize walking trajectories that are not stable while using a PD controller on the actuated coordinates. This may be especially the case for walking trajectories of several degree underactuated 3D biped robots [22] (unstable hybrid zero dynamics) or when replaying recorded trajectories on an leg exoskeleton carrying a mobility-impaired person. Finally, especially due to a lack of time, we did not implement this controller on a real robot. As [72] noticed when working with the compass walker, the implementation of a LQR is more difficult than the one of a high gain PD controller. Indeed, a LQR requires a control of the joint torques. It would be interesting to further investigate how much more difficult the implementation is.

5.8 Conclusion

New normal forms for planar one degree underactuated biped robots with point feet were derived in this chapter using Time-Scaling. The phase variables used in the HZD method were used as new time scales. It is unclear if these forms can provide motion planning and stabilization techniques outperforming those derived from the standard forms in the ordinary time t. The normal form with a one dimensional zero dynamics (5.101) seems to be appealing. But, it has the drawback to use a set of coordinates with an abstruse physical meaning, and is not necessarily valid on the entire state space. The derivation of a LQR from the

first-order approximation along a trajectory allows to stabilize a walking cycle. But, this controller does not outperform a HZD controller, except in the case where the robot lost angular momentum.

As a matter of fact, this chapter should be viewed as a first step towards the use of Time-Scaling for the control of biped robots. Many issues are still to be addressed. The research of time scales with nice properties, such that a strict monotonicity in the entire state space was not addressed. The property of feedback equivalence to a linear system in a new time scale is still an open problem. A deeper investigation on the potentialities offered by the normal forms that we derived is to be done. Also, all the derivations that we presented are valid if and only if the time scale is strictly monotonic in the ordinary time t. A strategy to adopt when the time scale goes backward in time has to be found. Finally, after gaining maturity, it would be interesting to generalize this approach to the case of tridimensional robots, and especially to investigate if Time-Scaling may help to cope with the problem of the robust stabilization of several underactuated biped robots.

Conclusion and Perspectives

In this thesis, we addressed the design of robust and energy-efficient walking gaits for one degree underactuated biped robots with point feet. We found that the HZD method was the most appropriate for this study, since it is well documented and rigorous.

Following the experimental reports in [127], we implemented the HZD method on a real planar robot with point feet. For the considered robot, we found that the computation of the hybrid zero dynamics was not easy. Then, we preferred to find a periodic orbit for the full model. This is not exactly in the spirit of the HZD method which proposes to find a periodic orbit for the hybrid zero dynamics only. The simulation results were in accordance with experiments. But, while the literature reports the use of contact sensors to detect the impact of the swing leg onto the ground, we ended up not using such sensors since they deteriorated quickly during experiments. Instead, we preferred using the measure of the phase variable to decide when the robot should switch the role of the legs.

From these experiments, we noticed that more than a half of the kinetic energy of the robot was dissipated when the leg impacted the ground, causing noise, vibration and the deterioration of the hardware. This motivated us to introduce a strategy to control the relative kinetic energy dissipated at impact in Chapter 4, using the standard rigid impact model of Hurmuzlu [56]. Numerical results obtained on a planar five-link walker with point feet showed that minimizing the energy dissipation is contradictory with the minimization of the torque consumption during the walk. This explains why trajectories that dissipate so much energy at impact are obtained when one seeks to minimize the torque consumption only. Then, a compromise has to be found to find energy-efficient walking trajectories with moderate energy losses at impact. On the opposite, we also designed motions dissipating most of the kinetic energy and showed that they can be used to stop the robot in "one step".

Underactuation renders the design of robust walking gaits challenging [127]. Since a change of time scale in the dynamic equations (Time-Scaling) is known to offer new control design opportunities [103], in Chapter 5, we investigated the opportunities offered by Time-Scaling for planar one degree underactuated biped robots with point feet. We chose to use the phase variables introduced in the HZD method as new time scales. Then, using coordinate and feedback transformations, we derived new exact normal forms with a two and a one-dimensional zero dynamics. We also designed a linear quadratic regulator under the form of a state feedback controller using the linearization of the dynamics along a reference

trajectory. The controller was tested in simulation on a planar five link walker with point feet. It is capable of tracking a periodic walking trajectory, but it appeared to be less robust than a controller stabilizing the robot into an hybrid zero dynamics manifold asymptotically stable to a periodic orbit (HZD method).

Chapters 4 and 5 bring some new partial guidelines for the design of robust and energy efficient walking gaits for one degree underactuated biped robots with point feet. Chapters 4 highlights the importance of not only taking into account energy-efficiency, but also impact losses to preserve the hardware. Chapter 5 investigates a new way for the design of walking controllers. But, it is still unclear to us if this approach will result in controllers outperforming the existing ones.

As a conclusion, we present some future works that we could not investigate further:

- Experimental validation of the walking trajectories and controllers derived in Chapters 4 and 5, and investigation of their robustness.
- Investigating further the opportunities offered by the new normal forms derived in Chapter 5 for the design of controllers, especially the normal form with a one-dimensional zero dynamics (5.101).
- Investigating further the property of feedback equivalence to a linear and controllable normal form for planar one degree underactuated biped robots, which is still an open problem (except for the compass walker).
- Generalization of the study performed in Chapters 4 and 5 for tridimensional underactuated biped robots.

More generally, here are some ideas related to the HZD method:

- In Section 2.4.5.2, we remarked that the use of virtual constraints being functions of the complete state of the robot have never been used in the HZD method for tridimensional biped robots. Using such a parameterization, for instance virtual constraints under the form $y = q_a - q_{a,ref}(q_u, \sigma)$, where q_a are the actuated coordinates, $q_{a,ref}$ is the reference trajectory for q_a and $(q_u; \sigma)$ are the coordinates used to parameterize the hybrid zero dynamics manifold, may give more robust walking gaits. Indeed, in this case, the reference trajectory $q_{a,ref}$ would change and adapt to any error relative to the reference periodic orbit in the hybrid zero dynamics. It would be interesting to compare the robustness of walking gaits using these virtual constraints with the performances obtained when using the standard virtual constraints.
- The HZD method has recently been used for the design of walking gaits for a system {lower limb exoskeleton + user} in simulation [2]. New issues appear for the control of such a system compared to a standard biped robot. Among them are the uncertainties on the model of the user (length, mass,

inertia) and on the motion of the torso. Inspired by [2] and [47], we propose to design walking gaits robust to these uncertainties and disturbances using the cost function (2.43) and the virtual constraints mentioned just above ($y = q_a - q_{a,ref}(q_u, \sigma)$). It can be hoped that formulating the problem as a direct collocation problem as used for the DURUS robot [97] and in [2] will give robust walking gaits in a moderate computation time (maximally a few hours), which could be replayed online on the real system.

Appendix A

The Method of Poincaré Sections

The method of Poincaré sections allows for the study of existence and the stability of periodic orbits. The method allows to translate the search of periodic orbits for a continuous system in the full state space to the study of a discrete application (the Poincaré map) on a lower dimensional space. This greatly simplify things. We give here a short description of the method. A complete and rigorous presentation can be found in [52, 127, 89].

Let us consider a periodic orbit $\mathcal{O}$ for the system (1.85) that transversally intersects an hypersurface $\mathcal{S}_P$, called the Poincaré section, at the point x^*. Without loss of generality, we take the Poincaré section as the impact map, that is $\mathcal{S}_P = \mathcal{S}$. When slightly perturbating the orbit, we assume that the system still crosses the surface $\mathcal{S}$. More specifically, let $x_0 \in \mathcal{S}$ in the neighborhood of x^* and $t \to \Phi(x_0, t)$ be the flow of the hybrid system, i.e. the solution of (1.85) starting from x_0 at the time $t = 0$. We assume that the flow crosses $\mathcal{S}$ at the time $T_I(x_0) > 0$. The Poincaré map $P : \mathcal{S} \to \mathcal{S}$ is the application mapping x_0 to $x_1 = \Phi(x_0, T_I(x_0))$. An illustration is given figure A.1.

It can be proved that the existence of periodic orbits is equivalent to the existence of fixed points for the Poincaré map, i.e. $P(x^*) = x^*$. And the periodic orbit is locally exponentially stable if and only if the eigenvalues of the Jacobian of the Poincaré map $\frac{\partial P}{\partial x}(x^*)$ have a modulus stricly inferior to one. In practice, the Poincaré map and its jacobian can only be computed numerically.

Since, as we saw above, the existence of periodic orbits for the hybrid zero dynamics is equivalent to the existence of periodic orbits for the full system, the Poincaré map can be computed only in the hybrid zero dynamics, reducing the computation cost.

The method of Poincaré allows only to study the local stability of periodic orbits. Estimating the global stability and the size of the basin of attraction for an orbit is still an open problem.

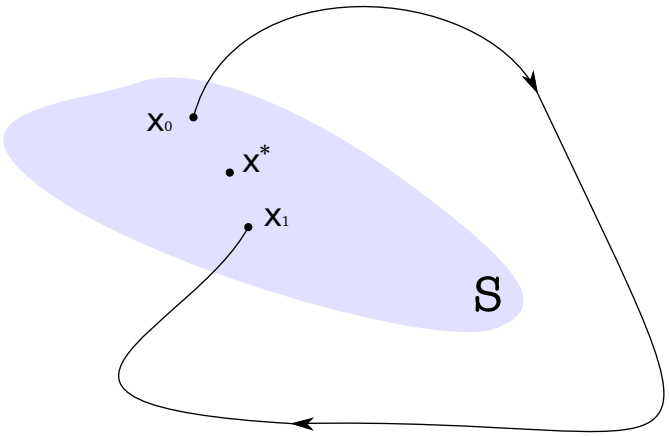
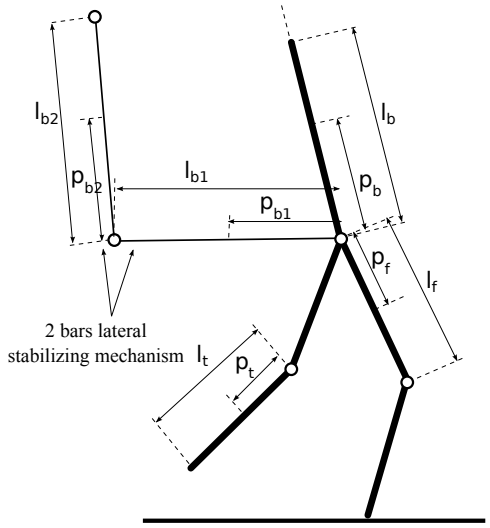


Figure A.1: Illustration of the Poincaré map.

Appendix B

Dynamic Parameters of VS



Torso	l_b	820 mm
	p_b	95 mm
	M_b	15.43 kg
	I_b^{com}	0.8567 kg.m^2
Thigh	l_f	400 mm
	p_f	201 mm
	M_f	5.42 kg
	I_f^{com}	0.1230 kg.m^2
Shin	l_t	400 mm
	p_t	215 mm
	M_t	2.25 kg
	I_t^{com}	0.029 kg.m^2
Transmission ratio	r_{knee}	101
	r_{hip}	100
Transmission inertia	$I_{eq,knee}$	1.89 kg.m^2
	$I_{eq,hip}$	1.46 kg.m^2
Bar 1	l_{b1}	500 mm
	p_{b1}	246 mm
	M_{b1}	2.25 kg
	I_{b1}^{com}	0.056 kg.m^2
Bar 2	l_{b2}	978 mm
	p_{b2}	494 mm
	M_{b2}	3.36 kg
	I_{b2}^{com}	0.32 kg.m^2

Figure B.1: Dynamical parameters of the VS robot (from CAD files). The transmission inertia is the reflected transmission inertia to the joint (see 1.6).

Appendix C

Estimation of the Positions q and Velocities $\dot{q}$ for the VS Robot

Multi-turn absolute encoders are mounted on the hip, the knee motor shafts, and on the joint angles of the 2 bars mechanism (to estimate the absolute orientation of the robot q_5). The servo drives are used to estimate the position and velocity of the encoder shaft. Unfortunately, the estimator used by the servo drive is not reported by the manufacturer. These data, expressed in counts and counts/s are sent to the embedded computer. The embedded computer computes the joint position and velocities from this data. We explain here how we proceeded to calibrate the encoders.

C.1 Brief Description of How Absolute Encoders Operate

Multi-turn absolute encoders encode the absolute position of a shaft. They keep in memory the number of revolutions performed. For the encoders mounted on the robot VS, a revolution of the encoder is equally divided in N segments, and each segment is encoded by a unique value. The shaft position q_{enc}, expressed in radian, is

$$q_{enc} = q_{enc,0} + \frac{2\pi}{N}(c - c_0)\,, \tag{C.1}$$

where c_0 (resp. c) is the value returned by the encoder (in counts) when the position of the shaft is $q_{enc,0}$ (resp. q_{enc}).

For the knees and the hips, assuming no mechanical backlash and flexibility in the mechanical transmission system, the joint position is

$$q_j = q_{j,0} + \frac{1}{r}(q_{enc} - q_{enc,0})\,, \tag{C.2}$$

where q_0 is the joint position when the position of the shaft is $q_{enc,0}$ and r is the transmission ratio of the mechanical transmission system.

The higher the resolution N is, the better the accuracy of the measure is. But the more expensive the sensor is. Interestingly, the accuracy of the measure of the joint position is r times higher than the measure of the corresponding motor shaft position.

Since the encoders are directly mounted on the joints of the lateral stabilizing mechanism, $q_{enc} = q_j$ for the joints of the mechanism.

C.2 Calibration of the Encoders

From (C.1), to decode the values sent by the encoder, the embedded computer needs to know the correspondence between one particular shaft position and one encoded value. In others words, the three values c_0, $q_{enc,0}$ and $q_{j,0}$ must be determined. Then, a calibration is necessary. Since the robot has no hard stop that could have been used as a position of reference for the calibration, we temporarily attached a low cost IMU to the links of the robot to measure. This allowed to get the joint position $q_{j,0}$.

But, IMU have biases that were not estimated. So, we do not know how this calibration method is accurate. We did not try to improve it, since we did not need to tune the parameters of the walking trajectory found in simulation (Bézier coefficients, lower and upper bounds of θ) for a successful experimental implementation.

C.3 Estimation of the Joint Position and Velocity

The embedded computer computes the joint positions using (C.3) and the joint velocities with the relation

$$\dot{q}_j = \frac{1}{r}\dot{q}_{enc}\,. \tag{C.3}$$

The obtained estimated hip and knee joint positions and velocities are directly used for the computation of the feedback law.

C.4 Estimation of the Absolute Orientation q_5 and $\dot{q}_5$

The torso absolute orientation q_5 can be reconstructed by measuring the three joint angles of the 2 bars mechanism α_3, α_4 and α_5

$$q_5 = \alpha_3 + \alpha_4 + \alpha_5 - \pi\,. \tag{C.4}$$

Using (C.1) gives a smooth estimate of q_5. But, using (C.3), the estimate of $\dot{q}_5$ is noisy. Indeed, contrary to the hip and knee, the encoders mounted on the 2 bars mechanism do not benefit of the transmission ratio that increases the accuracy

of the estimation of q_5. The estimation of the encoder position is not accurate enough so that the estimator of the servo drives return a smooth estimation of the encoder velocity.

Therefore, we use a linear regressor from the position measurements q_5 to estimate the velocity. The estimator runs at 1 kHz on the embedded computer. It performs a linear regression on the last 15 position measurements. The derivative of the regression line is taken as the estimate in velocity. Results are depicted in figure C.1. The estimation is smooth and is almost not delayed compared to the estimation computed by the servo drives.

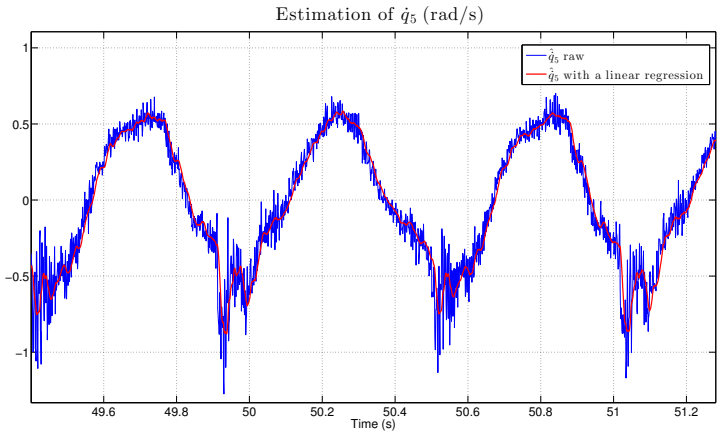


Figure C.1: Online experimental estimation of $\dot{q}_5$ using a linear regressor (in red). In blue is the estimation obtained using the servo drives reading the encoders of the bars.

Appendix D

Online Local Modifications of the Virtual Constraints for the VS Robot

This appendix presents the deadbeat hybrid extension that we implemented on VS (see chapter 3). An illustration is depicted figure D.1. Since the measure of the derivative of the phase variable $\dot{\theta}$ oscillates after impact, it is not taken into account and only the measure of the input after impact y^+ is considered. Virtual constraints are then locally modified so that the resulting new input after impact be zero. This prevents peaks of torques.

Let $h^*_{i,d}(s)$ be the nominal virtual constraint, i.e. computed offline. The normalized curvilinear abscissa is defined as

$$s = \frac{\theta - \theta^{*+}}{\theta^{*-} - \theta^{*+}}\,, \tag{D.1}$$

where θ^{*+} and θ^{*-} are the nominal values of θ at the beginning and the end of a step.

The virtual constraint with the hybrid deadbeat extension is

$$h_{i,d}(s) = h^*_{i,d}(s) + h_{i,c}(\theta, y_i^+, \theta^+)\,, \tag{D.2}$$

where $h_{i,c}(\theta, y_i^+, \theta^+)$ is the corrective term using the measure of the output y_i^+ and of the phase variable θ^+ just after impact.

The corrective term is used to join the position after impact to the reference trajectory on the position at the middle of the step, i.e. $s = \frac{1}{2}$. We choose $h_{i,c}(\theta, y_i^+, \theta^+)$ as a degree 4 Bézier polynomial

$$\begin{cases} \text{If } s_{i,bis} \in [0,1] : \\ h_{i,c}(s_{i,bis}) = a_{0i}(1-s_{i,bis})^4 + 4a_{1i}s_{i,bis}(1-s_{i,bis})^3 + 6a_{2i}s_{i,bis}^2(1-s_{i,bis})^2 \\ \qquad\qquad + 4a_{3i}s_{i,bis}^3(1-s_{i,bis}) + a_{4i}s_{i,bis}^4 \,. \\ \\ \text{If } s_{i,bis} > 1 : \\ h_{i,c}(s_{i,bis}) = 0 \,, \end{cases} \tag{D.3}$$

where

$$s_{i,bis} = \frac{\theta - \theta_0}{\frac{\theta^{*+}+\theta^{*-}}{2} - \theta_0} \,. \tag{D.4}$$

The polynomial must verify the following equalities

$$\begin{cases} h_{i,c}(0) = y_i^+. \text{ The corrective term zeroes the output after impact.} \\ h_{i,c}(1) = 0. \text{ The corrective term vanishes at the middle of the step.} \\ \dfrac{\partial h_{i,c}}{\partial s_{i,bis}}(0) = 0. \text{ The derivative of the corrective term does not change the} \\ \quad \text{derivative of the output after impact.} \\ \dfrac{\partial h_{i,c}}{\partial s_{i,bis}}(1) = 0. \text{ The derivative of the corrective term vanishes at the middle} \\ \quad \text{of the step.} \end{cases} \tag{D.5}$$

Then, we get

$$\begin{cases} \text{If } s_{i,bis} \in [0,1] : \\ h_{i,c}(s_{i,bis}) = y_i^+ \left[(1 - s_{i,bis}^4 + 4s(1-s_{i,bis})^3 \right] . \\ \\ \text{If } s_{i,bis} > 1 : \\ h_{i,c}(s_{i,bis}) = 0 \,. \end{cases} \tag{D.6}$$

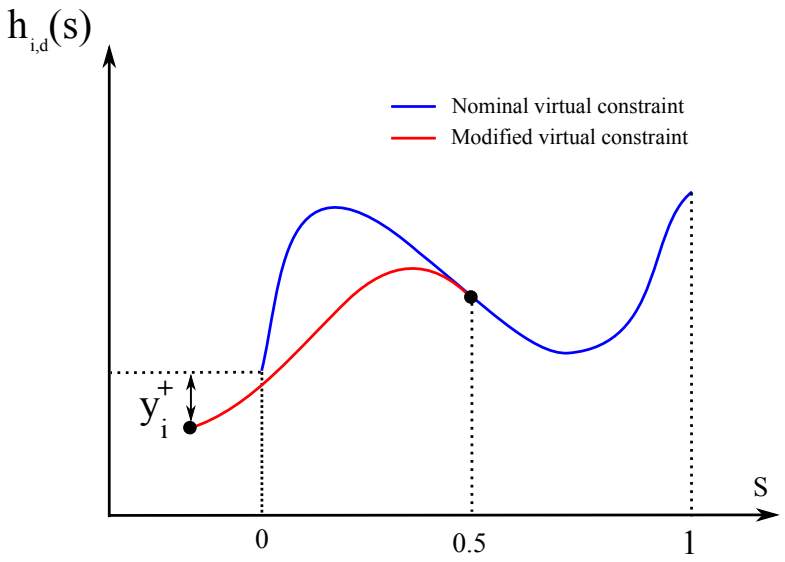


Figure D.1: Illustration of the deadbeat hybrid extension.

Appendix E

Classical Results of Linear Algebra

The rank-nullity theorem If A is a matrix with m rows and n columns, then

$$\text{rank}(A) + \dim(\text{Ker}(A)) = n\,. \tag{E.1}$$

The Sylvester's rank inequality If A is an $m \times n$ matrix and B is $n \times k$, then

$$\text{rank}(A) + \text{rank}(B) - n \leq \text{rank}(AB)\,. \tag{E.2}$$

Inequality on the rank of a matrix product If A is an $m \times n$ matrix and B is $n \times k$, then

$$\text{rank}(AB) \leq min(\text{rank}(A), \text{rank}(B))\,. \tag{E.3}$$

Appendix F

Details on the Motion Planning Algorithms Used to Control the Relative Energy Dissipation at Impact

This Appendix refers to Section 4.3.3. The motions are obtained using the HZD method (see Section 2.4.6.1) and the results established in Section 4.3.2. The optimization parameters are found by solving the nonlinear parametric optimization problem:

$$\boxed{\begin{array}{l}\text{Given a vector of unknown } \mathcal{P}\,,\\ \text{find } \min_{\mathcal{P}} J(x,u)\,,\text{subject to:}\\ \qquad C_e(x)=0\\ \qquad C_i(x)\leq 0\end{array}} \tag{F.1}$$

The cost function is the sum of the square torques divided by the step length over a step

$$J(x,u)=\frac{1}{L(x)}\int_0^{T(x)}||u(t)||^2\,dt\,. \tag{F.2}$$

Since the robot has four independent actuators, we define four virtual constraints for the actuated coordinates

$$y_i=q_i-h_{i,d}(\theta)\,,i\in\{1,...,4\}\,. \tag{F.3}$$

The functions $\theta\rightarrow h_{i,d}(\theta)$ are parameterized by a 6 degree Bézier polynomial [1]

$$\begin{aligned}h_{i,d}(s)=a_{0i}(1-s)^6+6a_{1i}s(1-s)^5+15a_{2i}s^2(1-s)^4+20a_{3i}s^3(1-s)^3\\ +15a_{4i}s^4(1-s)^2+6a_{5i}s^5(1-s)+a_{6i}s^6\,,\end{aligned} \tag{F.4}$$

[1] The degree of the polynomial was chosen sufficiently "high" to have a large space of search. A degree of 6 proved to be sufficient.

where

$$s = \frac{\theta - \theta^+}{\theta^- - \theta^+}\,, \tag{F.5}$$

is the normalized gait phasing variable. θ^+ (resp. θ^-) is the value of θ at the beginning (= right after impact) and at the end (= right before impact) of a step. The original set of optimization parameters is equal to 29: the $4 \times 7 = 28$ Bézier coefficients and the velocity of the phase variable at the beginning of the step $\dot{\theta}^+$.

The optimization problem is solved using the Matlab function *fmincon()*. The dynamical equations are integrated using *ode45*.

F.1 Design of Walking Trajectories with Little Energy Losses

F.1.1 Set of Optimization Parameters

The set of optimization parameters can be reduced by deriving some explicit relations between the Bézier coefficients from some constraints that the motion must verify.

F.1.1.1 The Configuration at Impact is not Optimized

The configuration of the robot at impact is fixed during the optimization. Indeed, we noticed that leaving it free results in poorer results. Several configurations at impact were tested. They were randomly chosen from anthropomorphic configurations.

Then, a_{01}, a_{02}, a_{03} and a_{04} are removed of the set of optimization parameters. Additionally, from the constraint of periodicity, we have

$$\begin{cases} q_2^+ = q_1^- \Leftrightarrow a_{0,2} = a_{6,1}\,, \\ q_1^+ = q_2^- \Leftrightarrow a_{0,1} = a_{6,2}\,, \\ q_4^+ = q_3^- \Leftrightarrow a_{0,4} = a_{6,3}\,, \\ q_3^+ = q_4^- \Leftrightarrow a_{0,3} = a_{6,4}\,. \end{cases} \tag{F.6}$$

Then, a_{61}, a_{62}, a_{63} and a_{64} are also removed.

F.1.1.2 Imposing the Relative Energy Dissipation

The desired relative energy dissipation r is imposed as explained in Section 4.3.2. More specifically, we select the direction of the velocity before impact $f'(\theta^-)$ in the four-dimensional vector subspace generated by the eigenvectors associated to the four largest eigenvalues: $\mathrm{Vect}(v_2, v_3, v_4, v_5)$. Indeed, we saw that three eigenvalues are 0, i.e. $\lambda_3 = \lambda_4 = \lambda_5 = 0$. Then, (4.60) gets

$$\alpha_2^2 = \frac{r \sum_{i=3}^{5} \alpha_i^2}{\lambda_2 - r}\,, \tag{F.7}$$

which always have solutions if $\lambda_2 < r$. Since we work with values of relative energy dissipation r close to zero, and that generally λ_2 is generally found to be smaller than -0.3, this condition is met.

To get the desired $f^{'}(\theta^-)$, the Bézier coefficients must verify

$$a_{5i} = a_{6i} - \frac{(\theta^- - \theta^+)}{6\dot{\theta}^-} f^{'}(\theta^-)\,, i \in \{1, 4\}\,, \tag{F.8}$$

where $f_i^{'}(\theta^-)$ is the i^{th} component of $f^{'}(\theta^-)$.

Then , a_{51}, a_{52}, a_{53} and a_{54} are removed from the optimization parameters. And α_2, α_3, α_4 are added instead.

Hence, $29 - 8 - 4 + 3 = 20$ optimization parameters are used in the motion planning algorithm

$$\mathcal{P} = (a_{11}, a_{21}, a_{31}, a_{41}, a_{12}, a_{22}, a_{32}, a_{42}, a_{13}, a_{23}, a_{33}, a_{43}, a_{14}, a_{24}, a_{34}, a_{44}, \alpha_3, \alpha_4, \alpha_5, \dot{\theta}^+)\,. \tag{F.9}$$

F.1.2 Constraints

We enumerate now the constraints $C_e(x) = 0$, and $C_i(x) \leq 0$ used for the parametric optimization and some of their typical corresponding thresholds. Note that depending on the kind of desired trajectory, some threshold values were changed.

Constraints to have a periodic orbit:

- Periodicity of the trajectory : θ and $\dot{\theta}$ periodic.

Constraints due to the model hypotheses:

- No take off of the stance foot: $F^z > 150N$.
- No slippage of the stance foot: $|\frac{F^x}{F^z}| < 0.5$.
- Swing foot strictly above the ground during a step: the swing foot altitude profile must be above a parabola with a height of 6 cm. ($\star$)
- Vertical velocity of the swing leg strictly positive at the beginning of the step.
- No take off of the leg that impacts the ground: $I^z > 0$.
- No slipping of the leg that impacts the ground: $|\frac{I^x}{I^z}| < 0.45$.

Constraints due to the physical limits of the robot:

- Joint limits. q_1, $q_2 \in [\frac{\pi}{2}, \frac{3\pi}{2}]$ and q_3, $q_4 \in [-\frac{3\pi}{5}, 0]$.
- Maximum joint torque: 100 Nm. ($\star\star$)

- Maximum joint velocity: 2.5 rad/s.

User's defined constraints:

- Mean walking speed between 0.3 m/s and 0.7 m/s.
- Step length between 0.25 m and 0.6 m.
- Torso inclination q_5 inferior to 0.15 rad.
- Torso movement inferior to 0.2 rad.
- Robustness to perturbations: Minimum kinetic energy level during a step superior to 1 J. ($\star\star\star$)

See Section 3.3 for further details on ($\star$), ($\star\star$) and ($\star\star\star$).

The stability of the walking trajectory is checked a posteriori by modifying the state of the robot at the beginning of a step (= right before impact), especially the value of $\dot{\theta}$, and integrating the dynamical equations to see if the robot converges to the walking cycle. This is roughly a Poincaré analysis.

Actually, as explained in Section 2.4.7, the stability of the hybrid zero dynamics is not difficult to obtain when the degree of underactuation is equal to one.

F.2 Design of a Motion to Stop the Robot in "One Step"

We design here a motion bringing the robot at impact with a relative energy dissipation close to -1 (high dissipation of energy). The motion starts from the beginning of a given walking trajectory (beginning of a step).

F.2.1 Set of Optimization Parameters

The set of optimization parameters can be reduced by deriving some explicit relations between the Bézier coefficients from some constraints that the motion must verify.

F.2.1.1 The State of the Robot at the Beginning of the Motion is Imposed

Since the motion starts from a given given walking trajectory a_{01}, a_{02}, a_{03}, a_{04} and a_{11}, a_{12}, a_{13}, a_{14} and $\dot{\theta}^+$ are imposed and removed from the optimization parameters.

F.2.1.2 Imposing the Relative Energy Dissipation

The desired relative energy dissipation r is imposed as explained in 4.3.2. More specifically, we select the direction of the velocity before impact $f^{'}(\theta^{-})$ in the one-dimensional vector subspace generated by the eigenvectors associated to the second lowest eigenvalue: Vect(v_2).

To get the desired $f^{'}(\theta^{-})$, the Bézier coefficients must verify (F.8). Then , a_{51}, a_{52}, a_{53} and a_{54} are removed from the optimization parameters.

Hence, $29-9-4=16$ optimization parameters are used in the motion planning algorithm

$$\mathcal{P} = (a_{2i}, a_{3i}, a_{4i}, a_{6i}), i \in \{1, 4\}. \tag{F.10}$$

F.2.2 Constraints

We enumerate now the constraints $C_e(x) = 0$, and $C_i(x) \leq 0$ used for the parametric optimization and some of their typical corresponding thresholds. Note that depending on the kind of desired trajectory, some threshold values were changed.

Constraints due to the model hypotheses:

- No take off of the stance foot: $F^z > 150N$.
- No slippage of the stance foot: $|\frac{F^x}{F^z}| < 0.5$.
- Swing foot strictly above the ground during a step: the swing foot altitude profile must be above a parabola with a height of 6 cm. ($\star$)
- Vertical velocity of the swing leg strictly positive at the beginning of the step.
- No take off of the leg that impacts the ground: $I^z > 0$.
- No slipping of the leg that impacts the ground: $|\frac{I^x}{I^z}| < 0.45$.

Constraints due to the physical limits of the robot:

- Joint limits. q_1, $q_2 \in [\frac{\pi}{2}, \frac{3\pi}{2}]$ and q_3, $q_4 \in [-\frac{3\pi}{5}, 0]$.
- Maximum joint torque: 100 Nm. ($\star\star$)
- Maximum joint velocity: 2.5 rad/s.

User's defined constraints:

- Mean walking speed between 0.3 m/s and 0.7 m/s.
- Step length between 0.25 m and 0.6 m.

- Torso inclination q_5 inferior to 0.15 rad.
- Torso movement inferior to 0.2 rad.
- Robustness to perturbations: Minimum kinetic energy level during a step superior to 1 J. ($\star\star\star$)
- The robot reaches the desired configuration at impact. ($\diamond$).
- The relative energy dissipation is equal to the desired one r. ($\diamond\diamond$).
- $K^- \leq K^-_{Max}$, where K^- is the kinetic energy right before impact, and K^-_{Max} is the maximum kinetic energy allowed right before impact. ($\diamond\diamond\diamond$).

See Section 3.3 for further details on ($\star$), ($\star\star$) and ($\star\star\star$) and Section 4.4.4.2 for further details on ($\diamond$), ($\diamond\diamond$) and ($\diamond\diamond\diamond$).

Appendix G

Deriving Normal Forms Using Feedback Equivalence

Normal forms for dynamical systems can be derived by applying both a transformation on the coordinates and on the input. This defines a *feedback equivalence relation* between the original system and the normal form [34]. This Appendix gives a brief presentation of the feedback equivalence and explains how it can be used to derive exact partial or full linear normal forms.

We will consider two dynamical systems. The first one is

$$\dot{x} = F(x) + G(x)u\,, \tag{G.1}$$

where $x \in \chi \subset \mathbb{R}^n$ is the state, $u \in U \subset \mathbb{R}^m$ is the m-dimensional vector of inputs, $x \to F(x)$ and $x \to G(x)$ are C^∞ vector fields on χ.

The second one is

$$\dot{\tilde{x}} = \tilde{F}(\tilde{x}) + \tilde{G}(\tilde{x})\tilde{u}\,, \tag{G.2}$$

where $\tilde{x} \in \tilde{\chi} \subset \mathbb{R}^n$ is the state, $\tilde{u} \in \tilde{U} \subset \mathbb{R}^m$ is the m-dimensional vector of inputs, $\tilde{x} \to \tilde{F}(\tilde{x})$, $\tilde{x} \to \tilde{G}(\tilde{x})$ are C^∞ vector fields on $\tilde{\chi}$.

G.1 Static Feedback Equivalence

Following the definition of [34], the system (G.1) is *statically feedback equivalent* to the system (G.2) if there exists a diffeomorphism

$$\begin{cases} \tilde{x} = \phi(x) \\ \tilde{u} = \psi(x, u) \end{cases} \tag{G.3}$$

such that if $(x(t), u(t))$ is a solution of (G.1), then $\Big(\phi(x(t)), \psi(x(t), u(t))\Big)$ is a solution of (G.2). This is equivalent to

$$\begin{cases} \tilde{F}(\phi(x)) + \tilde{G}(\phi(x))\psi(x,u) = \dfrac{\partial \phi}{\partial x}(x)[F(x) + G(x)u] \\ \tilde{s}(\phi(x)) = s(x) \end{cases} \tag{G.4}$$

Differently said, two systems are feedback equivalent if and only if there exist regular transformations on the coordinates and on the input allowing to switch from one system to the other.

G.1.1 Dynamic Feedback Equivalence

A dynamic feedback consists in adding a dynamics to some inputs, defining a *dynamic feedback transformation* This additional dynamics, generally called dynamic compensator, is defined as

$$\begin{cases} \dot{\kappa} = a(x, \kappa, \mathrm{w}) \\ u = k(x, \kappa, \mathrm{w}) \end{cases} \tag{G.5}$$

where $\kappa \in \mathcal{P} \subset \mathbb{R}^p$, $p \in \mathbb{N}^*$, $\mathrm{w} \in \mathrm{W} \subset \mathbb{R}^m$ is the new input, $(x, \kappa, \mathrm{w}) \to a(x, \kappa, \mathrm{w})$ and $(x, \kappa, \mathrm{w}) \to k(x, \kappa, \mathrm{w})$ are regular functions on $\chi \times \mathcal{P} \times \mathrm{W}$. Note that κ may be of any dimension.

As a matter of fact, κ is an additional state component. The new state is $\overline{x} = (x; \kappa)$. The *extended system* is the set of equations describing the dynamics of the extended state $\overline{x}$, that is the set of the $(n + p)$ equations (G.1) and (G.5)

$$\begin{cases} \dot{x} = F(x) + G(x)k(x, \kappa, \mathrm{w}) \\ \dot{\xi} = a(x, \kappa, \mathrm{w}) \end{cases} \tag{G.6}$$

or

$$\dot{\overline{x}} = \overline{F}\Big(\overline{x}, \mathrm{w}\Big) \tag{G.7}$$

with $\overline{F}(\overline{x}, \mathrm{w}) = \Big(F(x) + G(x)k(x, \xi, \mathrm{w})\,;\, a(x, \xi, \mathrm{w})\Big)^T$.

Let us define an other $(n + p)$-dynamical system

$$\dot{\tilde{x}} = \tilde{F}(\tilde{x}, \tilde{\mathrm{w}}) \tag{G.8}$$

where $\tilde{x} \in \tilde{\chi} \subset \mathbb{R}^{n+p}$ is the state, $\tilde{\mathrm{w}} \in \tilde{\mathrm{W}} \subset \mathbb{R}^m$ is the m-dimensional vector of inputs, $\tilde{x} \to \tilde{F}(\tilde{x})$, $\tilde{x} \to \tilde{G}(\tilde{x})$ are C^∞ vector fields on $\tilde{\chi}$.

The system (G.1) will be said to be *dynamically feedback equivalent* to (G.8) if there exits a regular dynamic compensator for (G.1) such that the extended system (G.7) is static feedback equivalent to (G.8).

G.2 Feedback Equivalence to a (Partial) Linear System

We consider here the particular case of feedback equivalence to a (partial) linear system. We especially describe the process used to transform a nonlinear system into a (partial) linear and controllable system. We present some concepts used to formalize this process, such as the concepts of output, relative degree and zero dynamics [3].

In the particular case where the nonlinear system is feedback equivalent to a (partial) linear system, we will use the term *feedback equivalence to a (partial) linear system*. This is also designed under the term *exact feedback linearization* (see [19, 3] and references therein). But, we will prefer to use the name *feedback equivalent to a (partial) linear system*, since the term "linearization" is misleading. Indeed, it could lead to the belief that the transformed system is an approximation of the original one. It has been intensively researched over the past 40 years if various classes of nonlinear systems are feedback equivalent to a (partial) linear system.

G.2.1 Output and Relative Degree

Let $y_1 = h_1(x)$ be a C^∞ function of the state, and called *output*. The time derivative of the output is

$$\mathring{y}_1 = L_F h_1 + (L_G h_1)u \,. \tag{G.9}$$

where $L_F h_1$ is the *Lie derivative* of the output h_1 along the vector field F

$$L_F h_1 = \frac{\partial h_1}{\partial x} F \,. \tag{G.10}$$

The symbol $L_F^2 h_1$ means $L_F(L_F h_1)$ and $L_F^k h_1$ means applying L_F k-times. We have $L_F^0 h_1 = h_1$. $L_F(L_G h_1)$ is simply denoted as $L_F L_G h_1$. The interest of the Lie derivative is that it allows compact notations. The k^{th} derivative of the output is

$$y_1^k = L_F^k h_1 + L_G L_F^{k-1} h_1 \,. \tag{G.11}$$

The output y_1 is said to be of *relative degree* r at $x = x_0$ if the input u appears for the first time in the r^{th} derivative of y_1, that is

$$\begin{cases} L_G L_F^k h_1 = 0, \, k = 1, ..., r-1 \\ L_G L_F^r h_1 \neq 0 \end{cases} \tag{G.12}$$

for all x in a neighborhood of x_0.

These definitions can be generalized to a vector of m outputs

$$y = h(x) = \begin{pmatrix} h_1(x) \\ : \\ h_m(x) \end{pmatrix} . \tag{G.13}$$

The *vector relative degree* of the vector of outputs y is $(r_1, ..., r_m)$ at a point $x = x_0$ if for all $1 \leq k \leq r_i - 1$, for all $1 \leq i \leq m$, for all x in a neighborhood of x_0

$$L_G L_F^k h_i(x) = 0 \tag{G.14}$$

, and the decoupling matrix

$$\begin{pmatrix} L_G L_F^{r_1} h_1(x) = 0 \\ : \\ L_G L_F^{r_m} h_m(x) = 0 \end{pmatrix} , \tag{G.15}$$

is nonsingular at $x = x_0$.

G.2.2 Building a new set of coordinates from the output

We consider here a m-dimensional vector of outputs, m being the number of inputs, whose vector relative degree is $(r_1, ..., r_m)$ at $x = x_0$. Let ξ_i be the r_i-dimensionnal vector containing the output y_i and its first $r_i - 1$ derivatives, we have

$$\xi_i = \begin{pmatrix} h_i \\ L_F h_i \\ ... \\ L_F^{r_i - 1} h_i \end{pmatrix} . \tag{G.16}$$

Let $r = (r_1 + ... + r_m)$ and ξ be the r-dimensional vector

$$\xi = \begin{pmatrix} \xi_1 \\ : \\ \xi_m \end{pmatrix} . \tag{G.17}$$

Then, from Proposition 5.2 in [3], $r \leq n$ and there exists a $(n - r)$-dimensional vector z such that (ξ, z, t) is a new set of coordinates for the system (G.1). In these coordinates, the dynamics become

$$\begin{cases} \dot{h}_1 = L_F h_1 \\ \vdots \\ \dot{\overbrace{L_F^{r_1-1} h_1}} = \alpha_1(\xi, z) + \beta_1(\xi, z) u \\ \vdots \\ \dot{h}_m = L_F h_m \\ \vdots \\ \dot{\overbrace{L_F^{r_m-1} h_m}} = \alpha_m(\xi, z) + \beta_m(\xi, z) u \\ \dot{z} = f_{zero}(\xi, z, u) \end{cases} \tag{G.18}$$

where $(\xi, z, u) \to f_{zero}(\xi, z, u)$ is a smooth r-dimensional vector field, $(\xi, z) = \phi(x)$ is the coordinate transformation, $(\xi, z) \to \alpha_i(\xi, z)$ and $(\xi, z) \to \beta_i(\xi, z)$ are smooth functions.

G.2.3 A Static Change of Feedback to Get a (Partial) Linear System.

Let

$$\alpha(\xi, z) = \begin{pmatrix} \alpha_1(\xi, z) \\ \vdots \\ \alpha_m(\xi, z) \end{pmatrix}, \tag{G.19}$$

and

$$\beta(\xi, z) = \begin{pmatrix} \beta_1(\xi, z) \\ \vdots \\ \beta_m(\xi, z) \end{pmatrix}, \tag{G.20}$$

where $\beta(\xi, z)$ is a $(m \times m)$ matrix, called *decoupling matrix*. If the decoupling matrix is invertible near $x = x_0$, then the static change of feedback

$$v = \psi(x, z, u) = \alpha(x, z) + \beta(x, z) u\,, \tag{G.21}$$

is invertible and transforms (G.18) into

$$\begin{cases} \dot{h}_1 = L_F h_1 \\ \quad \vdots \\ \dot{\overbrace{L_F^{r_1-1} h_1}} = v_1 \\ \quad \vdots \\ \dot{h}_m = L_F h_m \\ \quad \vdots \\ \dot{\overbrace{L_F^{r_m-1} h_m}} = v_m \\ \dot{z} = f_{zero}\Big(\xi, z, \psi^{-1}(\xi, z, v)\Big) \end{cases} \tag{G.22}$$

where v_i is the i^{th} component of v. Then, the system is made of m chains of integrators

$$\begin{cases} \dot{h}_i = L_F h_i \\ \quad \vdots \\ \dot{\overbrace{L_F^{r_i-1} h_i}} = v_i \end{cases} \tag{G.23}$$

of dimension r_i, $i \in \{1, m\}$, of a $(n-r)$-dimensional nonlinear dynamics $\dot{z} = f_{zero}(\xi, z, u)$, and of the dynamics of the ordinary time t.

Each output $y_i = \xi_i$ is controlled by the new input v_i through a chain of r_i integrators. The system can be written under the more compact form

$$\begin{cases} \dot{\xi} = A\xi + Bv \\ \dot{z} = f_{zero}\Big(\xi, z, \psi^{-1}(\xi, z, v)\Big) \end{cases} \tag{G.24}$$

The system $A\xi + Bv$ is under the so-called Brunovsky form, that is under the form of chains of integrators. It is a linear and controllable system.

G.2.3.1 The Zero Dynamics

When the output y is maintained at zero, that is when $\xi = 0_{r\times 1}$, the nonlinear $(n-r)$-dimensional dynamics

$$\dot{z} = f_{zero}\Big(0, z, \psi^{-1}(0, z, v)\Big), \tag{G.25}$$

is called *zero dynamics*. Its solutions are the solutions of the dynamics (G.24) when zeroing the output.

We will say that (G.1) is *feedback equivalent to a partial linear system* if it is feedback equivalent to a system under the form (G.24) with a non empty zero dynamics. If the zero dynamics is empty (ignoring the dynamics of t), (G.1) will be said to be *feedback equivalent to a linear system.*

G.2.4 Finding a Set of Outputs With the Highest Vector Relative Degree

To achieve feedback equivalence to a linear system, a set of m outputs with a vector relative degree $(r_1, ..., r_m)$ such that $r_1 + ... + r_m = n$ must be found, with m also to be chosen. The existence of such outputs is restrictive (Lemma 5.2.2 in [3]). Additionally, if they exist, these outputs are generally difficult to find since a set of partial differential equations must be solved. Then, intuition, trial and error are generally used in practice to find outputs with the highest vector relative degree.

G.2.4.1 Using Dynamics Extension for Increasing the Relative Degree

It's worth noting that using a dynamic feedback transformation as defined in G.1.1 may be helpful to increase the relative degree of an output. This property is especially used in this dissertation (see Section 5.5).

For instance, there exist sometimes outputs y_i with a relative degree r_i relative to $p < m$ inputs. Let $v_{1,p}$ be these inputs and $v_{p+1,m}$ be the $m - p$ inputs that do not appear in $y_i^{(r_i)}$. Adding in the state the inputs $v_{1,p}$ and defining the new vector of inputs as

$$\mathrm{w} = (\dot{v}_{1,p}, v_{p+1,m}), \tag{G.26}$$

makes the relative degree of y_i greater or equal than $r_i + 1$ for the extended system. For further details, see Chapter 5 (section 4) in [3].

Bibliography

[1] Wang A, Chevallereau A, Carlos F, and Rengifo B. Walking and steering control for a 3D biped robot considering ground contact and stability, 2013.

[2] Ayush Agrawal, Omar Harib, Ayonga Hereid, Sylvain Finet, Matthieu Masselin, Laurent Praly, Aaron D. Ames, Koushil Sreenath, and J. W. Grizzle. First steps towards translating hzd control of bipedal robots to decentralized control of exoskeletons. *IEEE Access (on review)*, January 2017.

[3] Isidori Alberto. *Nonlinear Control Systems*. Springer, 1995.

[4] J.J. Alcaraz-Jiménez, D. Herrero-Pérez, and H. Martínez-Barberá. Robust feedback control of zmp-based gait for the humanoid robot nao. *The International Journal of Robotics Research*, 2013.

[5] AMBER-Lab. Amber-lab website (september 2016). `http://www.bipedalrobotics.com/`.

[6] G. Arechavaleta, J. P. Laumond, H. Hicheur, and A. Berthoz. An optimality principle governing human walking. *IEEE Transactions on Robotics*, 24:5–14, Feb 2008.

[7] Andrzej Banaszuk and John Hauser. Feedback linearization of transverse dynamics for periodic orbits in r3 with points of transverse controllability loss. *Syst. Control Lett.*, 26(3):185–193, October 1995.

[8] EKSO Bionics. Ekso bionics website (september 2016). `http://eksobionics.com/`.

[9] REX Bionics. Arex bionics website (september 2016). `http://www.rexbionics.com/`.

[10] R. Blickhan. The spring-mass model for running and hopping. *J. of Biomech.*, 1989.

[11] Tomas De Boer. *FOOT PLACEMENT in robotic bipedal locomotion*. PhD thesis, TU Delft, 2012.

[12] B. Bona and M. Indri. Friction compensation in robotics: an overview. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 4360–4367, Dec 2005.

[13] B. Brogliato. *Nonsmooth Impact Mechanics. Models,Dynamics and Control.* Lecture Notes in Control and Information Sciences Springer-Verlag, 1996.

[14] Arthur E. Bryson and Yu-Chi Ho. *Applied Optimal Control.* Taylor & Francis, 1975.

[15] B. G. Buss, K. A. Hamed, B. A. Griffin, and J. W. Grizzle. Experimental results for 3d bipedal robot walking based on systematic optimization of virtual constraints. In *2016 American Control Conference (ACC)*, pages 4785–4792, July 2016.

[16] Brian Buss. *Systematic Controller Design for Dynamic 3D Bipedal Robot Walking.* PhD thesis, University of Michigan, 2015.

[17] J. Carpentier, F. Valenza, N. Mansard, and al. Pinocchio: fast forward and inverse dynamics for poly-articulated systems. `https://stack-of-tasks.github.io/pinocchio/`.

[18] J. Chen. The effects of gear reduction on robot dynamics. *JPL, California Inst. of Tech., Proceedings of the NASA Conference on Space Telerobotics*, 4:297–307, 1989.

[19] D. Cheng, A. Isidori, W. Respondek, and T. J. Tarn. Exact linearization of nonlinear systems with outputs. *Mathematical systems theory*, 21(1):63–83, 1988.

[20] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas de wit, and J. W. Grizzle. Rabbit: a testbed for advanced control theory. *IEEE Control Systems Magazine*, pages 57–79, 2003.

[21] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas de wit, and J. W. Grizzle. Rabbit: a testbed for advanced control theory. *IEEE Control Systems Magazine*, pages 57–79, 2003.

[22] C. Chevallereau, J. W. Grizzle, and C. L. Shih. Asymptotically stable walking of a five-link underactuated 3-d bipedal robot. *IEEE Transactions on Robotics*, 25(1):37–50, Feb 2009.

[23] Christine Chevallereau, J. W. Grizzle, and Ching-Long Shih. Asymptotically stable walking of a five-link underactuated 3D bipedal robot. *IEEE Transactions on Robotics*, 2008.

[24] Steven H. Collins, Peter G. Adamczyk, and Arthur D. Kuo. Dynamic arm swinging in human walking. *Proceedings of the Royal Society of London B: Biological Sciences*, 276(1673):3679–3688, 2009.

[25] X. Da, O. Harib, R. Hartley, B. Griffin, and J. W. Grizzle. From 2d design of underactuated bipedal gaits to 3d implementation: Walking with speed tracking. *IEEE Access*, 4:3469–3478, 2016.

[26] X. Da, O. Harib, R. Hartley, B. Griffin, and J. W. Grizzle. From 2d design of underactuated bipedal gaits to 3d implementation: Walking with speed tracking. *IEEE Access*, 4:3469–3478, 2016.

[27] X. Da, R. Hartley, and J. W. Grizzle. First steps toward supervised learning for underactuated bipedal robot locomotion, with outdoor experiments on the wave field. *ICRA*, 2017.

[28] Xingye Da, Ross Hartly, and J. W. Grizzle. First steps toward supervised learning for underactuated bipedal robot locomotion, with outdoor experiments on the wave field. In *ICRA*, submitted, 2016.

[29] Koolen de Boer Rebula Goswami and Jerry Pratt. Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. 2011.

[30] A. M. Dollar and H. Herr. Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art. *IEEE Transactions on Robotics*, 24(1):144–158, Feb 2008.

[31] C. Doppmann, B. Ugurlu, M. Hamaya, T. Teramae, T. Noda, and J. Morimoto. Towards balance recovery control for lower body exoskeleton robots with variable stiffness actuators: Spring-loaded flywheel model. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5551–5556, May 2015.

[32] Antonio El Khoury. *Planning Optimal Motions for Anthropomorphic Systems*. Theses, Université Paul Sabatier - Toulouse III, June 2013.

[33] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger. Bipedal walking control based on capture point dynamics. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4420–4427, Sept 2011.

[34] W. Respondek et I.A. Tall. *Feedback equivalence of nonlinear control systems: a survey on formal approach. Chapter 4 in "Chaos in Automatic Control" (J.-P. Barbot et W. Perruquetti)*. Taylor & Francis, 2006.

[35] EtherCAT. Ethercat website (september 2016). `https://www.ethercat.org/default.htm`.

[36] Ryan James Farris. *Design of a powered lower-lim exoskeleton and control for gait assistance in paraplegics*. PhD thesis, Vanderbilt University, 2012.

[37] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson. Optimization based full body control for the atlas robot. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 120–127, Nov 2014.

[38] Alexander L. Fetter and John Dirk Walecka. *Theoretical Mechanics of Particles and Continua*. Dover Publications,Inc., Mineola New York, 1982.

[39] Alexander L. Fetter and John Dirk Walecka. *Theoretical Mechanics of Particles and Continua*. Dover Publication, Inc., 2003.

[40] Michel Fliess, Jean Lévine, and Pierre Rouchon. Flatness and defect of nonlinear systems: Introductory theory and examples. *International Journal of Control*, 61:1327–1361, 1995.

[41] Leonid B. Freidovich, Anton S. Shiriaev, and Ian R. Manchester. Stability analysis and control design for an underactuated walking robot via computation of a transverse linearization. *{IFAC} Proceedings Volumes*, 41(2):10166 – 10171, 2008. 17th {IFAC} World Congress.

[42] F. R. Gantmacher. *The Theory of Matrices*. AMS CHELSEA PUBLISHING, 2000.

[43] Goldstein, Poole, and Safko. *Classical Mechanics, Third Edition*. Addison Wesley, 2000.

[44] D. Gouaillier, C. Collette, and C. Kilner. Omni-directional closed-loop walk for nao. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 448–454, Dec 2010.

[45] B. Griffin and J. Grizzle. Walking gait optimization for accommodation of unknown terrain height variations. In *2015 American Control Conference (ACC)*, pages 4810–4817, July 2015.

[46] Brent Griffin. *Nonholonomic Virtual Constraints and Gait Optimization for Robust Robot Walking Control*. PhD thesis, University of Michigan, 2016.

[47] Brent Griffin and Jessy Grizzle. Nonholonomic virtual constraints and gait optimization for robust walking control. *IJRR*, 2016.

[48] J. W. Grizzle, C. H. Moog, and C. Chevallereau. Nonlinear control of mechanical systems with an unactuated cyclic variable. *IEEE Transactions on Automatic Control*, 50(5):559–576, May 2005.

[49] Jessy W. Grizzle and Christine Chevallereau. *Virtual Constraints and Hybrid Zero Dynamics for Realizing Underactuated Biped Locomotion (proposed Chapter)*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[50] Jessy W. Grizzle, Christine Chevallereau, Ryan W. Sinnet, and Aaron D. Ames. Models, feedback control, and open problems of 3d bipedal robotic walking. *Automatica*, 50(8):1955 – 1988, 2014.

[51] K. H. Ha, S. A. Murray, and M. Goldfarb. An approach for the cooperative control of fes with a powered exoskeleton during level walking for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(4):455–466, April 2016.

[52] P. Hartman. *Ordinary differential equations*. Second Edition, Birkhäuser, 1982.

[53] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames. 3d dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1447–1454, May 2016.

[54] HONDA. Honda website : Achieving stable walking with asimo (september 2016). `http://world.honda.com/ASIMO/history/technology2.html/`.

[55] Yildirim Hurmuzlu. Dynamics of bipedal gait part i: Objective functions and the contact event of a planar five-link biped. *Journal of Applied Mechanics*, 60:331–336, 1998.

[56] Yildrim Hurmuzlu and Dan B. Marghitu. Rigid body collisions of planar kinematic chains with multiple contact points. *The International Journal of Robotics Research (IJRR)*, 13(1):82–92, February 1994.

[57] J.W.Grizzle. Jessy grizzle's homepage (september 2016). `http://web.eecs.umich.edu/faculty/grizzle/index.html`.

[58] J.W.Grizzle. Videos of the dynamic leg locomotion laboratory (university of michigan) (september 2016). `https://www.youtube.com/user/DynamicLegLocomotion`.

[59] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pages 1620–1626 vol.2, Sept 2003.

[60] 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 vol.1, 2001.

[61] Shuuji Kajita, Hirohisa Hirukawa, Kensuke Harada, and Kazuhito Yokoi. *Introduction to Humanoid Robotics*. Springer, 2014.

[62] Shuuji Kajita and Kazuo Tani. Dynamic biped walking control on rugged terrain using the linear inverted pendulum mode. *Transactions of the Society of Instrument and Control Engineers*, 31(10):1705–1714, 1995.

[63] W. Khalil and E. Dombre. *Modeling Identification and Control of Robots*. Taylor & Francis, 2002.

[64] Wisama Khalil and Etienne Dombre. *Modeling, Identification and Control of Robots*. Taylor & Francis, Inc., Bristol, PA, USA, 3rd edition, 2002.

[65] J. Kim, J. W. Han, D. Y. Kim, and Y. S. Baek. Design of a walking assistance lower limb exoskeleton for paraplegic patients and hardware validation using cop. In *International Journal of Advanced Robotic Systems*, 2013.

[66] N. Koceska, S. Koceski, P. B. Zobel, and F. Durante. Control architecture for a lower limbs rehabilitation robot system. In *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*, pages 971–976, Feb 2009.

[67] Arthur D Kuo. The six determinants of gait and the inverted pendulum analogy: A dynamic walking perspective. *Human movement science*, 26 4:617–56, 2007.

[68] Michael Yurievich Levashov. Modeling, system identification, and control for dynamic locomotion of the littledog robot on rough terrain. Master's thesis, MIT, 2012.

[69] Shun-Jie Li and Witold Respondek. Orbital feedback linearization for multi-input control systems. *International Journal of Robust and Nonlinear Control*, 25(9):1352–1378, 2015.

[70] B. Lim, Kyungrock Kim, Jusuk Lee, Junwon Jang, and Youngbo Shim. An event-driven control to achieve adaptive walking assist with gait primitives. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 5870–5875, Sept 2015.

[71] Bruno Siciliano & Luigi Villani Lorenzo Sciavicco. Lagrange and newton-euler dynamic modeling of a gear-driven robot manipulator with inclusion of motor inertia effects. *Advanced Robotics*, pages 317–334, 1995.

[72] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake. Stable dynamic walking over uneven terrain. *The International Journal of Robotics Research*, 30(3):265–279, March 2011.

[73] Tad McGeer. Passive dynamic walking. *IJRR*, 9(2):62–82, March 1990.

[74] S. Miossec and Y. Aoustin. *Chapter Dynamical synthesis of a walking cyclic gait for a biped with point feet*, page 233–252. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[75] Sylvain Miossec. *A contribution to the study of biped walk*. Theses, Ecole Centrale de Nantes (ECN)) ; Université de Nantes, November 2004.

[76] B. Morris and J. W. Grizzle. A restricted poincare map for determining exponentially stable periodic orbits in systems with impulse effects: Application to bipedal robots, Dec 2005.

[77] B. Morris and J. W. Grizzle. Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots. *IEEE Transactions on Automatic Control*, 54(8):1751–1764, Aug 2009.

[78] X. Mu and Q. Wu. On impact dynamics and contact events for biped robots via impact effects. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(6):1364–1372, Dec 2006.

[79] Roy Müller, Kevin Tschiesche, and Reinhard Blickhan. Kinetic and kinematic adjustments during perturbed walking across visible and camouflaged drops in ground level. *Journal of Biomechanics*, 47(10):2286–91, 2014.

[80] Richard M. Murray. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[81] P. D. Neuhaus, J. H. Noorden, T. J. Craig, T. Torres, J. Kirschbaum, and J. E. Pratt. Design and evaluation of mina: A robotic orthosis for paraplegics. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–8, June 2011.

[82] Quan Nguyen, Xingye Da, J. W. Grizzle, and Koushil Sreenath. Dynamic walking on stepping stones with gait library and control barrier. In *Workshop on Algorithimic Foundations of Robotics (WAFR)*, submitted, 2016.

[83] K. Nishiwaki and S. Kagami. Strategies for adjusting the zmp reference trajectory for maintaining balance in humanoid walking. In *2010 IEEE International Conference on Robotics and Automation*, pages 4230–4236, May 2010.

[84] Reza Olfati-Saber. *Nonlinear Control of Underactuated Mechanical Systems with Application to Robotics and Aerospace Vehicles*. PhD thesis, MIT, 2001.

[85] H. Olsson, K. J. Åström, M. Gäfvert, C. Canudas De Wit, and P. Lischinsky. Friction models and friction compensation. *Eur. J. Control*, page 176, 1998.

[86] Beresford N. Parlett. *The Symmetric Eigenvalue Problem*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1998.

[87] Henk Nijmeijer Pieter van Zutven, Dragan Kostic. Foot placement for planar bipeds with point feet. *IEEE ICRA*, 2012.

[88] F. Plestan, J. W. Grizzle, E. R. Westervelt, and G. Abba. Stable walking of a 7-dof biped robot. *IEEE Transactions on Robotics and Automation*, 19(4):653–668, Aug 2003.

[89] L. Praly. *Applications de Poincaré, Solutions périodiques et Moyennisation (Notes de cours)*. 2006.

[90] J.E. Pratt and R. Tedrake. *Velocity-Based Stability Margins for Fast Bipedal Walking*, pages 299–324. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[91] Jerry Pratt, Chee-Meng Chew, Ann Torres, Peter Dilworth, and Gill Pratt. Virtual model control: An intuitive approach for bipedal locomotion. *The International Journal of Robotics Research*, 20(2):129–143, 2001.

[92] Neuhaus Pratt Koolen de Boer Rebula Cotton Carff, Johnson. Capturability-based analysis and control of legged locomotion, part 2: Application to m2v2, a lower body humanoid. 2011.

[93] QNX. Qnx website (september 2016). `http://www.qnx.com/content/qnx/en.html`.

[94] Marc raibert. *Legged Robots that Balance*. MIT Press, 1986.

[95] Marc Raibert, Kevin Blankespoor, Gabriel Nelson, and Rob Playter. Bigdog, the rough-terrain quadruped robot. In *Proceedings of the 17th World Congress*, page 10823–10825, 2008.

[96] Hamed Razavi, Anthony M. Bloch, Christine Chevallereau, and J. W. Grizzle. Symmetry in legged locomotion: a new method for designing stable periodic gaits. *Autonomous Robots*, 2016.

[97] J. Reher, E. A. Cousineau, A. Hereid, C. M. Hubicki, and A. D. Ames. Realizing dynamic and efficient bipedal locomotion on the humanoid robot durus. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1794–1801, May 2016.

[98] ReWalk. Rewalk website (september 2016). `http://rewalk.com/`.

[99] M. Reyhanoglu, A. van der Schaft, N. H. Mcclamroch, and I. Kolmanovsky. Dynamics and control of a class of underactuated mechanical systems. *IEEE Transactions on Automatic Control*, 44(9):1663–1671, Sep 1999.

[100] Siavash Rezazadeh, Christian Hubicki, Mikhail Jones, Andrew Peekema, Johnathan Van Why, Andy Abate, and Jonathan Hurst. Spring-mass walking with atrias in 3d : Robust gait control spanning zero to 4.3 kph on a heavily underactuated bipedal robot. *SME 2015 Dynamic Systems and Control Conference*, 2015.

[101] Siavash Rezazadeh, Christian Hubicki, Mikhail Jones, Andrew Peekema, Johnathan Van Why, Andy Abate, and Jonathan Hurst. Spring-mass walking with atrias in 3d: Robust gait control spanning zero to 4.3 kph on a heavily underactuated bipedal robot. *ASME 2015 Dynamic Systems and Control Conference*, 2015.

[102] A. Saito, K. Sekiguchi, and M. Sampei. Exact linearization by time scale transformation based on relative degree structure of single-input nonlinear systems. In *49th IEEE Conference on Decision and Control (CDC)*, pages 5408–5413, Dec 2010.

[103] M. Sampei and K. Furuta. On time scaling for nonlinear systems: Application to linearization. *IEEE Transactions on Automatic Control*, 31(5):459–462, May 1986.

[104] Rafal Goebel Ricardo G. Sanfelice and Andrew R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Presse, 2012.

[105] A. Sano and J. Furusho. Realization of natural dynamic walking using the angular momentum information. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 1476–1481 vol.3, May 1990.

[106] J. B. Saunders, Verne T. Inman, and Howard D. Eberhart. The major determinants in normal and pathological gait. *The Journal of Bone & Joint Surgery*, 35(3):543–558, 1953.

[107] Nelson Saunders, Neville Swilling, Bondaryk Billings, Lee Palyter, and Marc Raibert : Boston Dynamics. Petman : A humanoid robot for testing chemical protective clothing. 2012.

[108] Rodolphe Sepulchre, Mrdjan Janković, and Petar V. Kokotovic. *Constructive nonlinear control*. Communications and control engineering. Springer, London, New York, 1997.

[109] B. Shen, J. Li, F. Bai, and C. M. Chew. Development and control of a lower extremity assistive device (lead) for gait rehabilitation. In *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, pages 1–6, June 2013.

[110] M. W. Spong. Partial feedback linearization of underactuated mechanical systems. In *Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI International Conference on*, volume 1, pages 314–321 vol.1, Sep 1994.

[111] Koushil Sreenath, Hae-Won Park, Ioannis Poulakakis, and Jessy W. Grizzle. Compliant hybrid zero dynamics controller for achieving stable, efficient and fast bipedal walking on MABEL. *The International Journal of Robotics Research (IJRR)*, 30(9):1170–1193, August 2011.

[112] T. Sugihara and Y. Nakamura. Whole-body cooperative balancing of humanoid robot using cog jacobian. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, 2002.

[113] J. Swevers, W. Verdonck, and J. De Schutter. Dynamic model identification for industrial robots. *IEEE Control Systems*, 27(5):58–71, Oct 2007.

[114] T. Swift, A.B. Zoss, K. Strausser, M. Rosa, H. Kazerooni, D.M. Fairbanks, M.V. Pillai, M. Schwartz, B.G.A. Lambrecht, and S. Kruze. Powered lower extremity orthotic and method of operation, 2013. US20130150980.

[115] T. Takenaka, T. Matsumoto, and T. Yoshiike. Real time motion generation and control for biped robot -1st report: Walking gait pattern generation-. In

2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1084–1091, Oct 2009.

[116] David Tlalolini, Yannick Aoustin, and Christine Chevallereau. Design of a walking cyclic gait with single support phases and impacts for the locomotor system of a thirteen-link 3D biped using the parametric optimization. *Multibody System Dynamics*, 23(1):33–56, January 2010.

[117] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, Oct 2012.

[118] Pieter van Zutven. *Control and identification of bipedal humanoid robots: Stability Analysis and Experiments*. PhD thesis, TU Eindhoven, 2014.

[119] D. J. Villarreal, H. A. Poonawala, and R. D. Gregg. A robust parameterization of human gait patterns across phase-shifting perturbations. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, PP(99):1–1, 2016.

[120] M. Vukobratovic. Zero moment point-thirty five years of its life. *International Journal of Robotics*, 2004.

[121] M. Vukobratovic, D. Hristic, and Z. Stojiljkovic. Development of active anthropomorphic exoskeletons. *Medical and Biological Engineering*, 12(1):66–80, 1974.

[122] Spong M. W. Modeling and control of elastic joint robots. *Journal of Dynamic Systems, Measurement, and Control*, 109(4):310–318, 1986.

[123] I. w. Park, J. y. Kim, and J. h. Oh. Online biped walking pattern generation for humanoid robot khr-3(kaist humanoid robot - 3: Hubo). In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 398–403, Dec 2006.

[124] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. van der Kooij. Design and control of the mindwalker exoskeleton. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(2):277–286, March 2015.

[125] T. Wang and C. Chevallereau. A new control law for a 3d biped robot based on regulation of the zero moment point and joint path. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 27–32, Dec 2010.

[126] Ting Wang, Christine Chevallereau, and Carlos Rengifo. Walking and steering control for a 3d biped robot considering ground contact and stability. *Robotics and Autonomous Systems*, 60(7):962–977, 2012.

[127] Eric R. Westervelt, Jessy W. Grizzle, and Christine Chevallereau. Feedback control of dynamic bipedal robot locomotion. taylor & francis/crc, 2007.

[128] Pierre-Brice Wieber, Russ Tedrake, and Scott Kuindersma. *Modeling and Control of Legged Robots*, pages 1203–1234. Springer International Publishing, Cham, 2016.

[129] David A. Winter. *Biomechanics and Motor Control of Human Movement*. Wiley, 2009.

[130] Martijn Wisse, Steven H. Collins, and Andy Ruina. A three-dimensional passive-dynamic walking robot with two legs and knees. *IJRR*, 20(7):607–615, July 2001.

[131] A. Young and D. Ferris. State-of-the-art and future directions for robotic lower limb exoskeletons. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, PP(99):1–1, 2016.

[132] H. H. Zhao, W. L. Ma, A. D. Ames, and M. B. Zeagler. Human-inspired multi-contact locomotion with amber2. In *Cyber-Physical Systems (ICCPS), 2014 ACM/IEEE International Conference on*, pages 199–210, April 2014.

Résumé

Cette thèse porte sur le développement de lois de commande pour la marche des robots bipèdes. Le sous actionnement engendré par le basculement, volontaire ou involontaire, du pied en appui sur le sol représente une difficulté majeure. Nous abordons ce problème par l'étude de robots plans avec pieds ponctuels.
La première partie de la thèse est une compilation des informations issues de la littérature que nous avons jugées intéressantes. Nous traitons dans un premier temps de la modélisation adoptée, puis effectuons une revue des différentes méthodes existantes, et présentons la mise en œuvre expérimentale de l'une d'entre elles : la méthode HZD.
Dans une deuxième partie, nous procédons à une étude de la dissipation relative d'énergie cinétique du robot lorsque le pied impacte le sol. Nous utilisons les résultats issus de cette étude pour planifier des trajectoires de marche dissipant peu d'énergie. De telles trajectoires ont a priori le mérite de préserver la structure du robot et de générer moins de bruit. A contrario, des trajectoires dissipant la majorité de l'énergie du robot sont utilisées pour un arrêt rapide. Une étude numérique a montré que ces résultats sont robustes à des incertitudes de modèle.
Enfin, dans une dernière partie, afin de compenser les difficultés liées au sous actionnement, nous proposons d'utiliser le degré de liberté supplémentaire offert par un changement de l'échelle de temps dans les équations de la dynamique (Time Scaling) pour la classe de robots considérée. En utilisant par ailleurs un changement de coordonnées et de feedback, nous dérivons de nouvelles formes exactes et approximatives.

Mots Clés

Robot bipède, planification de trajectoires, dynamique des zéros, impact, Time Scaling

Abstract

This thesis addresses the general problem of the walking control of biped robots. The foot of the robot in contact with the ground may tip over and cause the robot to be underactuated. This a major difficulty in term of control. This problem is addressed by considering planar biped robots with point feet.
In a first part, we present a standard way of modeling such systems, a literature review of the existing methods, and then report experimental results of the walking control of a planar biped robot using the HZD method.
In a second part, we perform an analytic and numerical study of the relative kinetic energy dissipation when the foot of the robot impacts the ground. Using this study, we design gaits with low energy dissipation at impact, which a priori result in a better preservation of the hardware and a reduction of the noise. And trajectories dissipating almost all the kinetic energy are used to quickly stop the robot. Finally, in an attempt to alleviate the burden due to underactuation, we propose to investigate the additional degree of freedom provided in the control design, by a change of time scale in the dynamic equations (Time Scaling) for the considered class of biped robots. Using feedback transformation, we derive new exact and approximative normal forms.

Keywords

Biped robot, motion planning, zero dynamics, impact, Time Scaling