# Controlled walking of planar bipedal robots

Spoelstra, M.P.A.

*Award date:*
2018

# Controlled walking of planar bipedal robots

M.P.A. Spoelstra

DC 2018.066

Master's thesis

Coach: dr.ir. P.W.M. van Zutven

Supervisor: prof.dr. H. Nijmeijer

Committee: prof.dr.ir. W.P.M.H. Heemels[1]
dr. A. Saccon[2]
dr.ir. A.A.J. Lefeber[2]

[1]Eindhoven University of Technology
Department of Mechanical Engineering
Control Systems Technology Group

[2]Eindhoven University of Technology
Department of Mechanical Engineering
Dynamics and Control Group

Eindhoven, August 2018

# Summary

The control of bipedal robots, and legged robots in general, is motivated by the potential to give robots complete mobility, at least in the environments that are designed by and for humans. However, bipedal robots are only useful if they can be controlled in a robust way, because disturbances are inevitable in such environments. Part of what makes the control of bipedal robots difficult is their non-smooth dynamics, which comes from the repeated changes in the contact with the ground, and the associated collisions between the swing foot and the ground. Another reason is that the contact constraint with the ground is easily broken by the actions of the robot, but also by external disturbances from the environment, both of which can cause the robot to fall over. In this thesis, the contact constraint is described by Coulomb's model of static friction. The objective is to design a controller for a planar bipedal robot that produces robust walking to a desired location.

A five-link planar bipedal robot with point feet is considered that consists of a torso and two links for each leg. The motion of the robot is constrained to the sagittal plane. Furthermore, the orientation of the torso is fixed so that normally the robot is fully actuated during walking, whereby the gait is defined to have exactly one foot in contact with the ground at all times. However, the stance foot of the robot can still slip or lose contact with the ground according to Coulomb's model. In addition, torque limits are imposed on all four actuated joints of the robot. The kinematics of the robot is described using absolute coordinates to enable direct control over the robot's location. This also allows for a relatively straightforward derivation of the multibody dynamics using the Newton-Euler equations of motion. Lastly, the hybrid systems framework is used to combine the multibody dynamics with a state-reset law for the impacts with the ground, in order to provide a complete model of the robot.

The desired walking gait is defined as a function of the state of the robot to let the robot dynamically react to disturbances and to achieve robust walking. Hereby, the desired position of the swing foot is defined as a function of the location of the torso and the horizontal velocity of the torso. These two state variables are also regarded as the location and the velocity of the robot, respectively. Thereafter, an input-output linearization is performed with four output functions that together represent the error in the position of the torso and the error in the position of the swing foot. Note that the input-output linearization only holds if the applied torques stay within the torque limits. Subsequently, bounded controllers are used for the stabilization of the origin of the output error dynamics, which enforces the desired walking gait and allows the robot to walk to its desired location.

The bounded controllers enable the robot to stumble if the desired walking speed of the robot is set too high. An analytical approximation of the maximum achievable walking speed is derived for the given control design, which is used as a parameter constraint for the specification of the desired walking speed. In addition, an optimization problem is formulated that maximizes the controller bounds, subject to the torque limits and the contact constraint. The result of the optimization is used as a parameter constraint for the specification of the controller bounds. Both parameter constraints are taken into account to try to make sure that the robot can walk to the desired location without slipping or stumbling. This is the case for a region in the parameter space, which is called the region of controlled walking. The aforementioned parameter constraints provide an approximation for the boundary of the region of controlled walking. Lastly, multiple simulations are performed to show the behavior of the closed-loop system.

# Samenvatting

De motivatie achter het besturen van tweebenige robots, en robots met benen in het algemeen, is de mogelijkheid om robots complete mobiliteit te geven, tenminste in de omgevingen die ontworpen zijn door en voor mensen. Tweebenige robots zijn echter alleen nuttig als ze op een robuuste manier bestuurd kunnen worden, omdat verstoringen onvermijdelijk zijn in zulke omgevingen. Een deel van wat het besturen van tweebenige robots moeilijk maakt is hun niet-gladde dynamica, welke tot stand komt door de herhaaldelijke veranderingen in het contact met de grond, en de botsingen tussen de zwaaivoet en de grond. Een andere rede is dat het contact met de grond eenvoudig verloren kan worden door de acties van de robot, maar ook door externe verstoringen vanuit de omgeving, die beide ervoor kunnen zorgen dat de robot omvalt. In deze scriptie wordt het contact met de grond beschreven door Coulomb's model voor de statische wrijving. De doelstelling is om een regelaar te ontwerpen voor een planaire tweebenige robot die het robuust lopen naar een gewenste locatie oplevert.

Een vijf-delige planaire tweebenige robot met puntvoeten wordt beschouwd, die bestaat uit een torso en twee delen voor elk been. De bewegingen van de robot zijn begrensd tot het sagittale vlak. Verder is de oriëntatie van de torso vastgezet zodat normaliter de robot volledig aangedreven is tijdens het lopen, waarbij de loopbeweging is gedefinieerd om altijd precies een voet aan de grond te hebben. Echter kan de standvoet nog steeds uitglijden of contact verliezen met de grond volgens Coulomb's model. Daarnaast zijn er torsie limieten opgelegd aan alle vier de aangedreven gewrichten van de robot. De kinematica van de robot is beschreven met absolute coördinaten om de directe besturing van de locatie van de robot mogelijk te maken. Dit leidt ook tot een relatief eenvoudige afleiding van de zogenaamde multibody dynamica met behulp van de Newton-Euler bewegingsvergelijkingen. Tenslotte is gebruik gemaakt van de structuur van hybride systemen om de multibody dynamica te combineren met een toestand-reset wet voor de botsingen met de grond om een compleet model van de robot te krijgen.

De gewenste loopbeweging is gedefinieerd als een functie van de toestand van de robot om de robot dynamisch te kunnen laten reageren op verstoringen, en om robuust lopen te bewerkstelligen. Hierbij is de gewenste positie van de zwaaivoet gedefinieerd als een functie van de locatie van de torso en de snelheid van de torso. Deze twee toestandsvariabelen worden ook gezien als de locatie en de snelheid van de robot. Daarna is een ingang-uitgang linearisatie uitgevoerd met vier uitgangsfuncties die samen de positiefout van de torso en de positiefout van de zwaaivoet voorstellen. Merk op dat de ingang-uitgang linearisatie alleen geldig is als de uitgevoerde torsies binnen de torsie limieten zitten. Vervolgens zijn er begrensde regelaars gebruikt voor de stabilisering van de oorsprong van de uitgangsdynamica, wat de gewenste loopbeweging afdwingt en ervoor zorgt dat de robot naar de gewenste locatie kan lopen.

De begrensde regelaars maken het mogelijk dat de robot struikelt als de gewenste loopsnelheid van de robot te hoog is. Een analytische benadering van de maximaal haalbare loopsnelheid is afgeleid voor het gegeven ontwerp van de regelaar, en deze is gebruikt als restrictie bij de specificatie van de gewenste loopsnelheid. Daarnaast is een optimalisatieprobleem geformuleerd die de begrenzingen van de regelaar maximaliseert, onderhevig aan de torsie limieten en het contact met de grond. Het resultaat van de optimalisatie is gebruikt als restrictie bij de specificatie van de begrenzingen van de regelaar. Er is rekening gehouden met beide parameter restricties om te proberen zeker te stellen dat de robot naar de gewenste locatie kan lopen zonder uit te glijden of te struikelen. Dit is het geval voor een gebied in de parameter ruimte dat het gebied van het beheerst lopen wordt genoemd. De bovengenoemde parameter restricties verschaffen een benadering voor de grens van het gebied van het beheerst lopen. Tenslotte zijn er meerdere simulaties gedaan om het gedrag van het gesloten-lus systeem te laten zien.

# Nomenclature

**Symbols**

| | |
|---|---|
| $\boldsymbol{\lambda}_\sigma$ | vector of kinematic constraint forces |
| $\boldsymbol{\tau}$ | vector of applied torques |
| $\boldsymbol{e}_1, \boldsymbol{e}_2$ | unit vectors of the inertial frame |
| $\boldsymbol{e}_\sigma$ | output error vector |
| $\boldsymbol{q}$ | vector of generalized coordinates |
| $\boldsymbol{r}$ | position vector |
| $\boldsymbol{u}$ | input vector |
| $\boldsymbol{x}$ | state vector |
| $\boldsymbol{y}_\sigma$ | output vector |
| $\lambda$ | kinematic constraint force |
| $\mathbb{R}$ | set of real numbers |
| $\mathbb{Z}$ | set of integers |
| $\mathcal{B}$ | set of rigid bodies |
| $\mathcal{J}$ | set of joints |
| $\Delta$ | step size |
| $\mu$ | coefficient of static friction |
| $\sigma$ | mode |
| $\tau$ | torque |
| $\theta$ | orientation |
| $d$ | damping coefficient |
| $F$ | force |
| $g_\mathrm{a}$ | gravitational acceleration |
| $J$ | moment of inertia |
| $L$ | maximum torque |
| $l$ | length |
| $m$ | mass |

| | |
|---|---|
| $n$ | number of generalized coordinates |
| $p$ | number of actuators |
| $s$ | number of kinematic constraints |
| $t$ | time |
| $x, y$ | pair of position coordinates |

**Abbreviations**

| | |
|---|---|
| bfp | both feet pinned |
| CM, COM | center of mass |
| COP | center of pressure |
| COT | cost of transport |
| CP | Capture Point |
| d | desired |
| h | hip |
| k | knee |
| lf | left foot |
| lfp | left foot pinned |
| lh | left hip |
| lk | left knee |
| lsh | left shank |
| lth | left thigh |
| rf | right foot |
| rfp | right foot pinned |
| rh | right hip |
| rk | right knee |
| rsh | right shank |
| rth | right thigh |
| sh | shank |
| stf | stance foot |
| stl | stance leg |
| swf | swing foot |
| th | thigh |
| tr | torso |
| ZMP | Zero-Moment Point |

# Contents

# Chapter 1

# Introduction

In this chapter the main subjects of the thesis are introduced. These subjects are legged robots and the characteristics of two-legged walking. Then, the objectives of the thesis are stated, and an outline of the report is given. Lastly, the necessary terminology is introduced, which is used throughout the report.

## 1.1 Legged robots

Mobile robots that use legs for their mobilization are called legged robots. It is worthwhile to study and develop legged robots because of their wide variety of potential use cases, which range from prosthetics to space exploration. Some of the interest in legged robotics stems from the appeal of having robots that operate in anthropomorphic or animal-like ways. However, what fundamentally motivates the study of legged robots is the need for robots that are capable of moving across rough terrain. Incidentally this makes them also suitable for use in the environments that are designed by and for humans. Legged robots are in particular useful for walking over non-smooth terrain, which can be a cluttered environment, a flight of stairs, a ladder, or even something as simple as a curb. Most wheeled robots are unable to cross such terrain or overcome such obstacles. Clearly, legged locomotion offers a higher degree of mobility than wheeled locomotion does. However, an advantage of wheeled robots is that they do not tip over as easily as legged robots, nor do they require any power to stay upright.

Legged robots are complex machines; they consist of many interconnected links. In addition, the legs can only be unilaterally constrained to the ground. Simplified models are sometimes used to approximate the complex dynamics of a legged robot. For example, if just one leg is in contact with the ground, then the robot might be modeled as an inverted pendulum. Also, if the robot is moving slowly, then all standing legs might be considered to be temporarily bilaterally constrained to the ground. This allows the mobile robot to be modeled as a stationary robot for the period of time that these legs remain in contact with the ground.

A legged robot also undergoes a collision with every step that it takes. In general, these collisions are noticeable for the robot. Therefore, the effect of the collisions on the dynamics cannot be neglected. It shows that the dynamics of a legged robot is non-smooth, where the non-smoothness is due to the collisions, which are often considered as instantaneous events. One can imagine that it is difficult to model, let alone control a legged robot. Fortunately, our understanding of non-smooth dynamical systems has expanded significantly in the last two decades with the emergence of the field of non-smooth mechanics [32]. Currently, there are multiple mathematical frameworks for modeling non-smooth dynamical systems. These frameworks can be used to accurately model the dynamics of a legged robot. In the hybrid systems framework, for example, a non-smooth system is represented as a combination of differential equations with occasional resets of the state variable. As such, hybrid systems are characterized by the interaction between continuous-time dynamics and discrete transitions. In the case of legged robots, the continuous-time dynamics corresponds to the multibody dynamics of the robot, and the discrete events correspond to the collisions with the ground and the related changes in the contact points.

## 1.2 Bipedal walking

In the context of robotics, locomotion is the movement or displacement of a robot from one location to another. Hereby, the specific goal is to obtain a displacement of the robot's center of mass in the desired direction. Legged locomotion is a particular form of terrestrial locomotion that involves using legs for propulsion. There are several types of legged locomotion, such as hopping, running, and walking. This thesis focuses on bipedal walking, which means walking on two legs, and a robot that has two legs is similarly called a bipedal robot. In addition, the term "controlled walking" is used to refer to walking in a controlled manner, without the robot falling over. A bipedal robot can fall over after slipping or stumbling, for example. The two legs of a bipedal robot provide both the vertical support and the horizontal propulsion. During walking, the legs alternately push off against the ground, thereby exerting a force on the ground. According to Newton's third law, the ground simultaneously exerts an equal but opposing reaction force on the robot that actually propels the robot forward.

Research and development of bipedal robots arises from diverse sociological and commercial interests, such as the desire to assist or replace humans in hazardous environments, and to help augment and restore mobility for the disabled [64]. A bipedal robot is useful in particular, because it has a suitable shape for operating in the structured environments that are designed for humans, such as households and care homes. First of all, a bipedal robot can go to places where a wheeled robot cannot go, as mentioned in the previous section. In addition, a legged robot with two or more legs can in principle go everywhere a human can go, both indoors and outdoors. Indeed, having more than two legs is useful, because it significantly increases the robot's ability to stay balanced. Furthermore, a bipedal robot can go to places that require its slender figure and small footprint, such as with climbing a ladder, navigating a room filled with closely spaced objects that are not easily moved, and maneuvering and turning in tight spaces. Thus, in terms of practical use and applicability, bipedalism is the preferred form of legged locomotion. However, this only holds if bipedal robots can be controlled in a robust way.

A stride of bipedal walking can be subdivided into multiple phases based on the number of degrees of freedom that a bipedal robot has. The number of degrees of freedom can change based on the unilateral constraints between the robot and the ground. For a bipedal robot with feet, it depends on whether the feet are flat on the ground. It might be the case that the contacting feet are rotating about the heel or the toe. During natural human walking, for example, each foot hits the ground with a heel strike, after which the foot rotates to gain full contact with the ground, and the foot lifts off the ground with a toe roll-off. The different phases of walking can be distinguished based on which unilateral constraints are active. As a result of the varying number of degrees of freedom, it is possible that a walking bipedal robot is fully actuated in some phases and underactuated, or over-actuated, in other phases. In general, a mechanical system is said to be fully actuated if the number of degrees of freedom of the system is equal to the number of independent actuators. A mechanical system is said to be underactuated if the number of degrees of freedom exceeds the number of actuators. So, if all joints of a bipedal robot are actuated and one foot is in full contact with the ground, then the robot is fully actuated. However, if that foot starts rotating about an edge, such as the heel or the toe, then the robot becomes underactuated. That edge can be modeled as an additional revolute joint, which increases the number of degrees of freedom of the robot by one, while the number of actuators remains the same.

If it is desired to have full actuation at all times, then the controller of the robot must carefully consider the amount of torque that is applied to the joints, especially the ankle joints, because of the unilateral connection between the feet and the ground. In particular, if too much torque is applied to the ankle of the foot that is in contact with the ground, then the foot can start to rotate around an edge and cause the robot to become underactuated. In addition, if too much torque is applied to the other joints, then the foot can slip or lose contact with the ground altogether. Thus, a bipedal robot can fall over if too much torque is applied. On the other hand, a bipedal robot can also fall over if not enough torque is applied, because then the legs might not move quickly enough to maintain the robot's balance, and so the robot can stumble and fall over. In addition, a significant disturbance can cause the robot to lose its balance and fall over as well. In this case, the limited amount of torque from the actuators is not enough to reject the disturbance, so the robot falls over. One could simply assume that the bipedal robot is always underactuated and design a controller based on this assumption. However, it is not a trivial task to design a controller for the stabilization and/or regulation of underactuated systems. Finally, note that if the actuators in the ankle joints are turned off to allow the ankles to rotate passively so that the feet

remain flat on the ground in the presence of disturbances, then the robot is effectively walking on stilts. As a result, the robot is underactuated again, because even though the number of degrees of freedom remains the same, now the number of actuators has decreased.

In the absence of disturbances, a bipedal robot can be controlled simply by walking slowly. In this way the critical torque will not be reached, and so the robot stays fully actuated. The control of fully actuated robots is relatively straightforward, see for example [50]. However, in practice a bipedal robot must be able to reject disturbances. A measure for how well a robot deals with disturbances is robustness. Robustness is one of three aspects with which the performance of a walking bipedal robot can be assessed, see [8]. The other two aspects are the energy efficiency and the versatility of the bipedal locomotion. However, robustness is paramount for enabling the use of bipedal robots in everyday environments, where disturbances are inevitable. One of the reasons why bipedal robots are not widely used is that walking is not robust enough yet. Therefore, it seems reasonable to study walking and focus on the robustness of walking first. There is no point in having an energy efficient and/or versatile bipedal robot if it is not robust against disturbances.

## 1.3 Objectives

The main objective of this thesis is to design a controller for a bipedal robot that produces robust walking to a desired location. This is achieved by using a model of a bipedal robot that accurately captures the non-smooth dynamics of the robot. A model of such complexity has been used previously in [4, 60], where the multibody dynamics is formulated using coordinates that describe the motion of the links of the robot relative to each other. However, the use of relative coordinates has limitations if the goal is to control the bipedal robot to a specific location. Therefore, absolute coordinates are used here instead, as they enable direct control of the robot's location. Then, a bounded feedback controller is designed for the model such that a dynamic walking behavior of the closed-loop system is obtained. Walking is more robust if the desired walking motion is defined as a function of the state of the robot, whereby the motion of the legs is coordinated independent of time. This enables the robot to dynamically react to unexpected disturbances, such as a push. In addition, the desired location of the robot must be an asymptotically stable equilibrium point of the closed-loop system. So, any effort to move the robot away from its desired location should ultimately be rejected by the controller, whether it takes one step or several steps to do so. Intuitively, asymptotic stability is only possible if the robot is not underactuated. In short, the main goals of this thesis are to

i. model the non-smooth dynamics of a bipedal robot using absolute coordinates,

ii. specify the desired walking motion as a function of the state of the robot,

iii. design a bounded feedback controller for robust walking to the desired location,

iv. analyze the closed-loop system with simulations.

Figure 1.1 depicts the fundamental objective, which is the creation of an abstraction layer for legged locomotion.

## 1.4 Outline

The outline of the remainder of this thesis is as follows. In the section directly hereafter, the reader is familiarized with the terminology that is used throughout the report. The terminology pertains to legged robots and human anatomy, with in particular the anatomy of legs. In Chapter 2, an introduction is given on the topic of the control of bipedal robots, followed by a literature study on this subject. In the conclusion of the literature study it is stated in which area a contribution can be made to the field of bipedal robot control. In Chapter 3, a model is formulated for a planar bipedal robot, in which the non-smooth dynamics is modeled using the hybrid systems framework. Subsequently, a bounded feedback controller is designed for this robot in Chapter 4. The resulting closed-loop system is analyzed in Chapter 5, which is done mostly with simulations. In chapter 6, the conclusions are presented and several recommendations are given for future work.

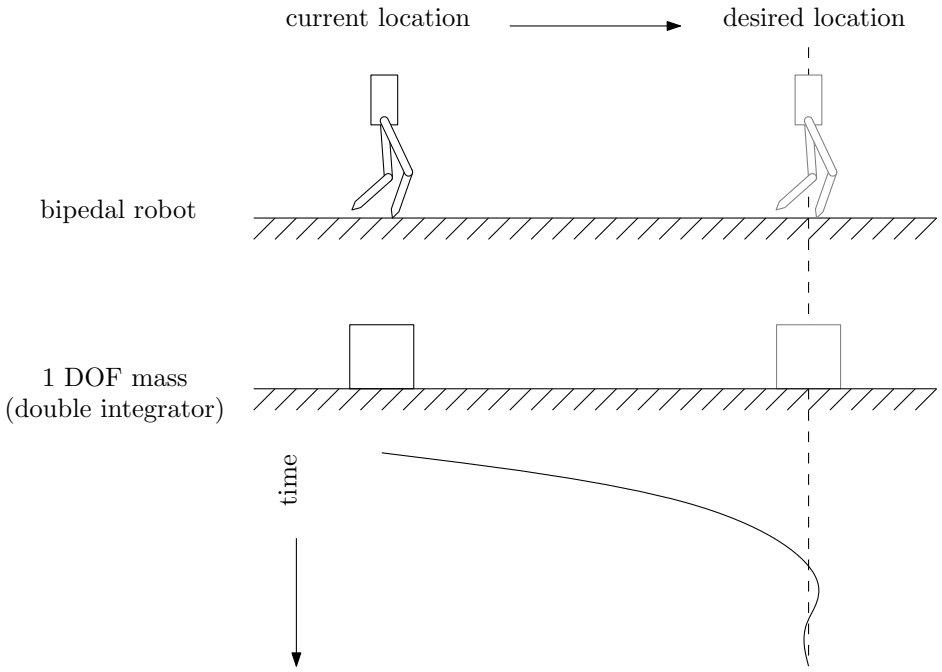



Figure 1.1: The control problem of walking to a desired location is meant to be simplified to the extent that it is analogous to the control of a mass with 1 degree of freedom (DOF). Along this single DOF, the closed-loop behavior of the robot should look something like that of a (nonlinear) mass-spring-damper system, as illustrated with the graph at the bottom of the figure. Ultimately, the only required input should be the setpoint for the desired location of the robot. In this way an abstraction layer is created, because an operator of the robot will not have to deal with the difficulties that are associated with the legged locomotion itself.

## 1.5 Terminology

It is assumed that the reader is familiar with the general terminology of robotics, where a *link* means an interconnected rigid body, for example. The terminology of bipedal robotics mainly consists of words taken from human anatomy to refer to different parts of a bipedal robot.

Like a general robot, a bipedal robot consists of multiple links that are connected together to form a chain of links. For a bipedal robot this chain contains two branches of sub-chains, which are called the *legs*. Like a human leg, a robotic leg can consist of an upper link, which is called the *thigh*, and a lower link, which is called the *shank*. Actually, the lower link is called the "leg" in human anatomy, which is somewhat confusing. Therefore, the word shank is used here instead. Usually, a leg also has a third link in the sequence called the *foot*. The joint that connects the thigh with the shank is called the *knee* joint. Likewise, the joint that connects the shank with the foot is called the *ankle* joint. However, a robotic leg does not have to match the anatomy of a human leg. Some bipedal robots do not have feet, for example, while others only have one link for each leg. In both these cases the robot is said to have *point feet*, because there are no links that act as the feet of the robot, and the robot must stand on the pointy ends of the legs instead. The legs branch off from a common sub-chain called the *torso*, which usually consists of one link. Unsurprisingly, the joints that connect the legs to the torso are called the *hip* joints.

At any given moment during walking, either one or both feet are in contact with the ground. The case where both feet are in contact with the ground is referred to as *double support*, and the case where one foot is in contact with the ground is referred to as *single support*, or the *swing phase*. During single support, the contacting foot is called the *stance foot* and the corresponding leg is called the *stance leg*. The other foot and leg are called the *swing foot* and *swing leg*, respectively.

The manner in which a bipedal robot moves is called the *gait*. Here, the focus is on walking of bipedal robots in the *sagittal plane*, which is one of the three perpendicular anatomical planes. Specifically, the sagittal plane is the plane of symmetry that divides the robot into mirrored left- and right-halves.

# Chapter 2

# A literature study of bipedal robot control

In this chapter the results of a literature study are presented that give an overview of notable motion control approaches for balancing, disturbance rejection, and walking on flat and level terrain. Both planar and unconstrained bipedal robots are discussed, whereby the planar bipedal robots are constrained to the sagittal plane.

## 2.1 Introduction to bipedal robot control

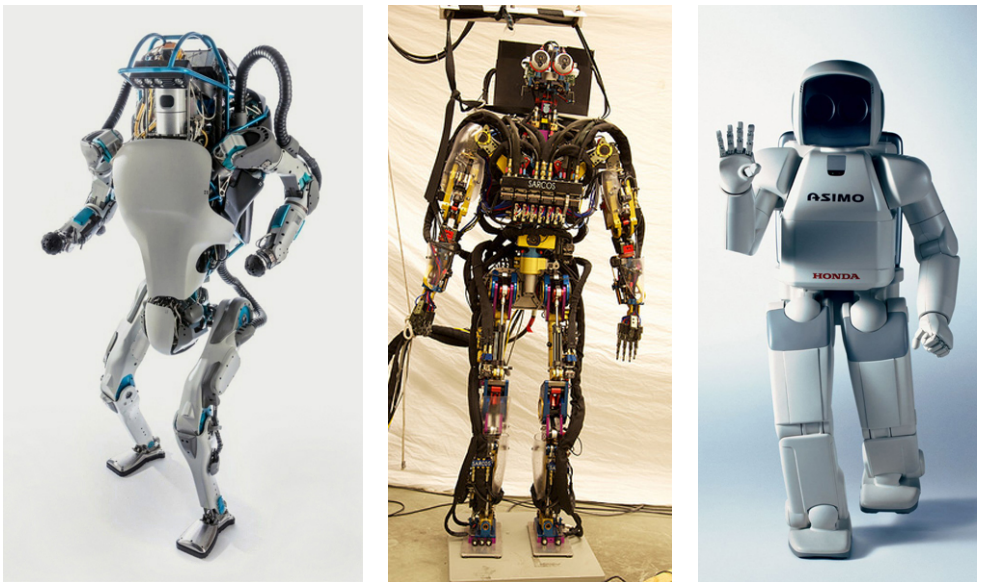
Motion control of legged robots has been studied actively since the 1980's. An impressive amount of technology has been specifically developed to build walking robot prototypes. A quick search in the literature reveals that over one hundred walking mechanisms have been built over the years by universities, companies and public research laboratories [64]. Some of the recent and more interesting bipedal robots that have successfully demonstrated walking include ATLAS by Boston Dynamics (Figure 2.1a), TORO by the German Aerospace Center (Figure 2.2b), ATRIAS by the Oregon State University (Figure 2.3b), and Cassie by Agility Robotics (Figure 2.4a).

There are several ways in which a bipedal robot can actively move around, namely by walking, running, and hopping. In this literature study, the focus is on walking. However, the balancing of bipedal robots is also discussed, because the control design for balancing can provide insights in how to reject disturbances that can prevent the robot from falling over. The ability to handle unexpected disturbances is also important for safe interactions with humans [53]. Note that balancing and walking both require that at least one foot is in contact with the ground.

According to Wieber [66], walking is the displacement of a legged robot by means of regularly detaching and restoring contact with the ground. In general, walking involves the horizontal displacement of the center of mass (COM) of a legged robot [22, 64]. The motion of the COM is influenced mainly by the unilateral contact forces between the feet and the ground [66]. Therefore, the foot placement is crucial for balancing and walking without falling over, as also stated in [3, 8, 59].

In order to reject disturbances, a controller must change the configuration of the robot in such a way that the robot does not start to tip over. This can be done by balancing on the feet that are already in contact with the ground, or by stepping in a certain direction. Controllers that are specifically designed to reject disturbances are called push recovery strategies in the literature. These strategies are commonly divided into three categories: ankle strategies, hip strategies or angular momentum strategies, and stepping strategies [51].

Over the years several criteria have been formulated for the balance of bipedal robots in order to determine when such robots are not going to fall over. These so-called balance criteria basically represent conditions for which a bipedal robot is fully actuated, if not over-actuated. A balance criterion can be used to try to stay balanced in the presence of disturbances, or it can be used for the synthesis of the walking gait by using it as a constraint in the gait design process. The resulting gait is often called stable or




(a) ATLAS by Boston Dynamics, weighs 75 kg and is 150 cm tall.

(b) Sarcos Primus by Sarcos, weighs 95 kg and is 160 cm tall.

(c) ASIMO by Honda, weighs 48 kg and is 130 cm tall.

Figure 2.1: Featured bipedal robots from different companies.

balanced. Unfortunately, the former naming leads to an ambiguity of the concept of stability in the sense of Lyapunov stability, as also stated in [15, 66]. Therefore, the latter naming is preferred here instead.

The simplest balance criterion involves projecting the COM of the bipedal robot onto the ground. The criterion states that if the projected COM stays within the convex hull of all ground contact points, i.e. the smallest convex set that contains all contact points, then the robot is statically balanced. The convex hull of all contact points is also called the support polygon in the context of legged robotics.

Another, more elaborate balance criterion is based on a point on the ground called the center of pressure (COP). The COP is the weighted average of the normal contact forces. Given $N$ contact points, each with a normal contact force $F_{\mathrm{n},i} \geq 0$, $i = 1, \ldots, N$, the total normal force $F_{\mathrm{n}} = \sum_{i=1}^{N} F_{\mathrm{n},i}$. Then, the position of the COP is given by

$$\boldsymbol{r}_{\mathrm{COP}} = \frac{1}{F_{\mathrm{n}}} \sum_{i=1}^{N} F_{\mathrm{n},i} \boldsymbol{r}_{\mathrm{c},i}, \tag{2.1}$$

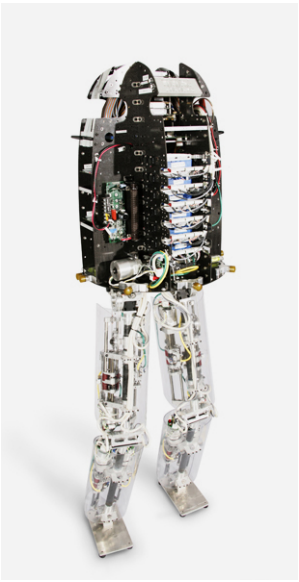
where $\boldsymbol{r}_{\mathrm{c},i}$ denotes the position of the $i$-th contact point.

The COP is related to another point called the Zero-Moment Point (ZMP). The ZMP has been introduced by Vukobratovic et al. in 1972 [63], and it can be defined as follows. First, assume that there exists a single point on the ground where both the ground reaction force (GRF) and the ground reaction moment (GRM) act on the robot. The vertical component of the GRF is simply equal to the total normal force $F_{\mathrm{n}}$. The two horizontal components of the GRF and the vertical component of the GRM represent the friction forces and the friction torque that keep the stance feet stationary. Now, if the two remaining horizontal components of the GRM are equal to zero, then the aforementioned point on the ground is called the Zero-Moment Point. Notice that the two horizontal components of the GRM are equal to zero if the stance feet are at rest and flat on the ground.

It has been shown that the COP coincides with the ZMP if the stance feet are flat on the ground, i.e. if the stance feet are in full contact with the ground [10, 15, 60, 63]. The ZMP can be calculated independent of the COP by using the appropriate momentum balances, as shown in [63]. The ZMP is used for the well-known ZMP criterion. Formally, the ZMP criterion states that if the ZMP lies in the interior of the support polygon, then all stance feet are at rest and in full contact with the ground [63]. This means that if the ZMP lies on the boundary of the support polygon, then according to the ZMP criterion it is not known whether the stance feet are at rest and in full contact with the ground. In

(a) HRP-4 by the Humanoid Robotics Project (HRP), weighs 39 kg and is 150 cm tall.

(b) TORO by the German Aerospace Center (DLR), weighs 75 kg and is 160 cm tall.



(c) M2V2 by the Institute for Human and Machine Cognition (IHMC), weighs 45 kg and is 180 cm tall.

Figure 2.2: Featured bipedal robots from different research institutes.

this case it is certainly possible that the robot is tipping around an edge of one of the stance feet. If this happens, then an additional degree of freedom is introduced to the system, which usually results in underactuation. The ZMP is not defined for the case where the stance feet are not flat on the ground. The ZMP criterion is predominantly used for the control of bipedal robots, and legged robots in general. Yet, its usefulness for real-world applications is sometimes questioned, see for example [15, 38].

The definition of the ZMP requires that the stance feet are flat on the ground. This restriction has led Goswami to introduce the Foot Rotation Indicator in [15], which specifies the direction and magnitude of the moment of rotation that the robot experiences if the stance feet are not flat on the ground. In [66], a generalization of the ZMP criterion is presented, called the Viability Kernel, which gives an alternative definition of balance for legged robots. Hereby, Lyapunov functions are used to determine the accompanying Viability Margin, which represents the boundary between balance and imbalance.

There are two main approaches to the control of bipedal robots that are found in the literature. The conventional approach is to generate reference trajectories for the joints of the bipedal robot that satisfy a balance criterion. These reference trajectories are usually generated in advance, because the computations for it are quite time-consuming, especially for the case of a complex robot. Various existing control methods can then be used to track the reference trajectories, such as PID control, or sliding mode control [36, 37, 55].

However, there is an alternative approach that has been developed in recent years, whereby an effort has been made to apply the control theory of non-smooth dynamical systems to bipedal robots. For example, in the book [64] by Westervelt, Grizzle, et al., the steady state walking of a bipedal robot is viewed as a periodic orbit in the state space of the robot. The control approach consists of a feedback linearization with certain output functions. The output functions represent virtual holonomic constraints on the configuration of the robot. These virtual constraints guide the motion of the robot through the walking gait. The constraints are enforced by zeroing the outputs through feedback control. The use of feedback linearization leads to the notion of zero dynamics. In [65], this notion has been extended to a class of non-smooth dynamical systems called systems with impulsive effects. The extended notion is called hybrid zero dynamics, because of the hybrid nature of impulsive systems. Note that bipedal robots belong to this class of non-smooth dynamical systems. Poincare sections and Lyapunov functions are used to assess the stability of the periodic orbit in the hybrid zero dynamics. The result is an

(a) Spring Flamingo by the Massachusetts Institute of Technology, weighs 13.5 kg and is 90 cm tall.

(b) ATRIAS by the Oregon State University, weighs 62 kg and is 170 cm tall.

(c) MABEL by the University of Michigan, weighs 56 kg and is 160 cm tall.

Figure 2.3: Featured (planar) bipedal robots from different universities.

asymptotically stable periodic walking gait.

In the next section, many important and notable publications are discussed about the control of planar bipedal robots, with an emphasis on balancing and walking. In the section thereafter, the same is done for unconstrained bipedal robots, which are in general more difficult to control.

## 2.2 The control of planar bipedal robots

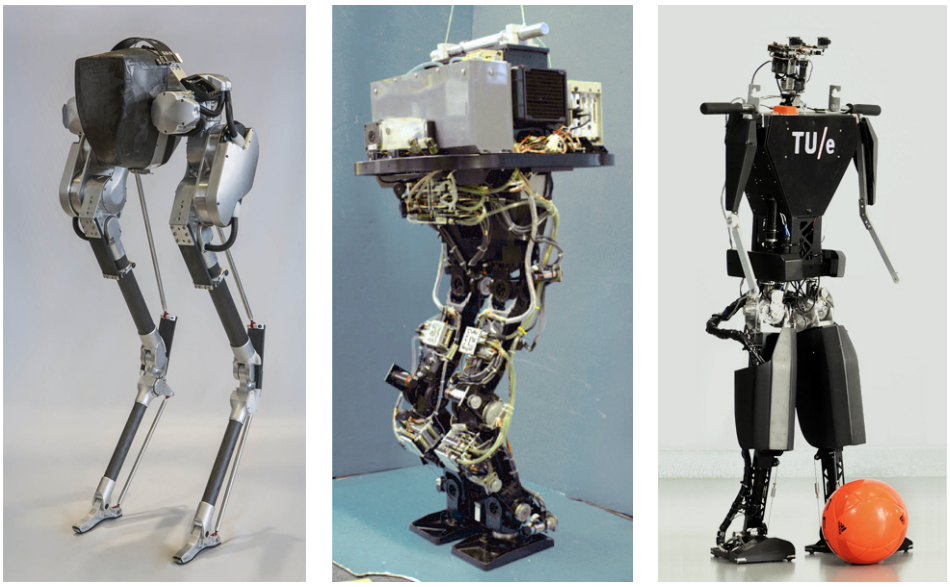
In this section, the literature about the control of planar bipedal robots is discussed, which tends to focus only on the locomotion in the sagittal plane.

### 2.2.1 Disturbance rejection and balancing

Controllers that enable a bipedal robot to balance in the presence of disturbances are collectively called push recovery strategies in the literature. Hereby, if a bipedal robot does not take a single step while it reacts to a disturbance, then the push recovery strategy is called a fixed-support strategy [33]. Among fixed-support strategies, a distinction is also made between a so-called ankle strategy and a hip strategy, where the ankle and the hip are predominately used to reject the disturbances, respectively [52].

However, in some cases a fixed-support strategy is not sufficient to reject a disturbance. In this case the robot has to resort to a change-of-support strategy, i.e. a stepping strategy. Thus, altogether there are three categories of balance strategies: the ankle strategy, the hip strategy, and the stepping strategy. In [51], it is suggested that these three strategies can be used sequentially, starting with the ankle strategy, and advancing to the next one if the current one is inadequate. Decision regions are derived for each strategy using a simple model of a bipedal robot. These regions can be interpreted as part of a balance criterion, because they can be used to choose the appropriate strategy for staying balanced [51].

An ankle strategy is based on the COP, while a hip strategy is based on the angular momentum of the robot. In human balance experiments, it is observed that humans use a combination of both these strategies in order to deal with disturbances, according to Stephens in [52]. Stephens proposes to use integral control to execute both strategies together. Integral control has been used previously in [30] in




(a) Cassie by the Oregon State University and its spin-off Agility Robotics, weighs 30 kg and is 100 cm tall.

(b) HRP3L-JSK by the University of Tokyo and the HRP, weighs 53 kg and is 100 cm tall.

(c) TUlip by the Eindhoven University of Technology and Philips, weighs 31 kg and is 134 cm tall.

Figure 2.4: Featured bipedal robots from different universities in association with other organizations.

the context of bipedal robot control. An advantage is that the dynamics of the COP and the COM can be decoupled, which makes it easier to keep the COP within the support polygon [52]. Simulations show that the control action of this method is similar to the reaction of humans. However, de Boer points out in [8] that the effectiveness of this decoupled control method typically depends on the fine-tuning of the associated control parameters. Also note that the method proposed by Stephens is based on simple models of bipedal robots. To apply this method to a more complex bipedal robot, Stephens proposes to use a "model tracking control algorithm", whereby feedback controllers are used to let the robot track the response of the double inverted pendulum model.

Stepping strategies are often studied separate of fixed-support strategies. In the context of push recovery, a stepping strategy is used in order to come to a stop after a push by placing the swing foot at the appropriate location on the ground. In [33], Maki and McIlroy have shown with push recovery experiments that people prefer to take a step even if a change in posture, i.e. a fixed-support strategy, is sufficient to restore the balance. It is also pointed out that people prefer to use a stepping strategy over a hip strategy when specifically given the choice. Out of all balance strategies, stepping strategies are arguably the most effective, and the most energy efficient as well [3, 8, 62].

A stepping strategy for a bipedal robot is usually based on either the Capture Point (CP) [8, 39] or the Foot Placement Estimator (FPE) [62, 68]. Both the CP and the FPE are points that indicate where to place the swing foot on the ground so that the robot comes to a standstill above the stance foot after a single step. The CP has been introduced by Pratt et al. in 2006 [39]. It is derived using the energy conservation of the linear inverted pendulum model (LIPM), without taking into account the effect of impacts on the dynamics.

The FPE has been introduced by Wight et al. in 2008 [68]. The FPE is derived using the nonlinear inverted pendulum model, and does take the energy-loss of impacts into account. However, Wight et al. consider the legs to be massless. It is questionable whether stepping strategies that are based on massless legs are applicable to real bipedal robots with non-massless legs [2]. That is why an extension of the FPE is proposed in [2], whereby the leg mass is not neglected. Simulations show that this extension works better than the original FPE. Lastly in [62], van Zutven et al. introduced the Foot Placement Indicator (FPI), which is an extension of the FPE to planar bipedal robots with point feet and an arbitrary number of non-massless links. Simulations show that the FPI is more accurate than the FPE.

Push recovery experiments show that the FPE is highly correlated with human foot placements in the sagittal plane [34]. However, it is also observed that humans step before the FPE during fast walking. Stepping some distance before the FPE allows one to maintain a certain walking speed [34]. Also, the FPE only indicates where to place the swing foot on the ground in order to come to a standstill. It is natural to think that it is more energy efficient to make slightly smaller steps in the case that one wants to keep on walking. Lastly, in [68] it is shown with simulations that the FPE can be used for disturbance rejection during walking in the sagittal plane. Similarly, it is shown that the CP can be used for disturbance rejection during walking in the unconstrained case [8, 31].

### 2.2.2 Walking

A number of different methods have been proposed in the literature to achieve planar bipedal walking. One recent approach by Ames et al. uses motion capture data of human walking to achieve human-like walking on a bipedal robot [69, 70]. The approach is called human-inspired walking control, and it is stated that the main motivation for using data of human walking is to utilize the inherent walking robustness present in the captured data. It is also observed that the motion of certain points on the human body can be characterized by the time response of a linear mass-spring-damper system [47]. In [69], an efficient algorithm is presented that determines the best fit of this model to the motion capture data. In addition, constraints can be added to ensure that the gait can be realized on a given bipedal robot. The method has been experimentally verified as well, see [69, 70].

A similar but earlier approach to achieve planar bipedal walking is the so-called virtual model control method [40, 41, 42], which has been introduced by Pratt in 1995 [41]. In this approach, virtual mechanical components, such as virtual springs and dampers, are added to the bipedal robot and between the robot and its environment. In [24, 42], specific rules are given to switch between different sets of virtual components, which is meant to increase the robustness of walking. Virtual model control is known for being computationally efficient and straightforward to implement. Successful walking has been demonstrated on the planar bipedal robot Spring Flamingo, which is depicted in Figure 2.3a.

Recall that steady state walking can be viewed as a periodic orbit in the state space of a bipedal robot. In the early 2000's, Grizzle et al. have developed a method for establishing local asymptotic stability of the steady state walking motion of planar bipedal robots, see for reference [16, 17, 19, 35, 65]. Herein, feedback linearization is used with certain output functions that are specifically designed to generate a certain walking gait if the outputs are equal to zero. Both fully actuated and underactuated phases of walking are possible. Ultimately, the research has resulted in a book that contributes a framework for the control design and the stability analysis of walking and running planar bipedal robots [64]. The book gives a complete description of how to design the feedback controllers for walking at a given speed. Grizzle et al. have demonstrated their results on the planar bipedal robot MABEL, see Figure 2.3c, and later also on the unconstrained bipedal robot ATRIAS, see Figure 2.3b.

Lately, Ames, Galloway and Grizzle have shown how control-Lyapunov functions can be used to exponentially stabilize the periodic orbit of a walking gait [1]. Previously, Poincare sections have been used to analyze the stability of the periodic orbit. In [48], Song and Zefran introduce a computational approach for the stabilization of periodic orbits for systems with impulsive effects. Therein, the orthogonality transform is used to decompose the dynamics along a periodic orbit into its transverse and tangential components. The effectiveness of this approach is demonstrated with simulations of an underactuated 5-link planar bipedal robot.

## 2.3 The control of unconstrained bipedal robots

In this section, an overview is given of the literature about the control of unconstrained bipedal robots.

### 2.3.1 Disturbance rejection and balancing

In [54], Stephens and Atkeson have successfully demonstrated balancing on the humanoid robot Sarcos Primus. This robot is equipped with hydraulic actuators, as seen in Figure 2.1b, as well as force sensors in the joints of the robot to measure the joint torques. The required joint torques to maintain a balance

are computed using a model of the COM dynamics that is similar to the LIPM [53]. Subsequently, force control is used to realize the computed joint torques.

Hyon et al. have successfully demonstrated balancing on an earlier version of the same robot. However, they use a passivity-based approach to achieve balancing [26, 27, 28]. In [26], gravity compensation is proposed as part of the balance controller. It is shown that the robot is able to maintain its balance, even if unexpected external forces are applied at various points on the robot [27]. A simple stepping controller, which is based on symmetric stepping [43], is implemented for the attenuation of larger disturbances. In [28], Hyon et al. have extended their balance controller with a control law that is similar to the one proposed in [54]. The control law is added in parallel to the existing controller, which has resulted in a significant improvement of the balancing capability of the robot.

In the context of push recovery, de Boer et al. have introduced the notion of capturability, which puts Capture Point-based stepping in a general framework [8]. This framework is used to analyze and synthesize foot placement points that can prevent a bipedal robot from falling over. The most important concept within this framework is $N$-step capturability, which is the ability of a bipedal robot to come to a standstill in $N$ steps or less without falling over. The capturability framework is derived using the three-dimensional LIPM. It is a model where the COM dynamics in the sagittal plane is not coupled to the COM dynamics in the frontal plane. It is stated in [8, 9] that simple models, such as the LIPM, can be used effectively to synthesize robust stepping strategies. This has been suggested in bio-mechanical studies as well, where such simple models have been found to be good predictors of human foot placements [21, 34]. A particular stepping strategy with $N$-step capturability has been successfully implemented on a bipedal robot called M2V2. A picture of this robot is shown in Figure 2.2c.

However, some aspects of push recovery are often neglected in the literature. In [3], it is shown experimentally that the mass of the swing leg and the impacts of the swing foot with the ground have a significant effect on the motion of the robot during stepping. Furthermore, it is observed that undesired sideways motion can occur during forward stepping. Therefore, it is useful to place both feet side-by-side after a forward step to try to prevent the robot from tipping over to one side. It is concluded that the dynamics in the sagittal plane is coupled to the dynamics in the frontal plane. This conclusion is based on experiments that are performed with the humanoid robot TUlip from the Eindhoven University of Technology [61]. A picture of TUlip is shown in Figure 2.4c.

In [56], Urata et al. have introduced a method of fast trajectory generation for the foot placement of bipedal robots. This method for foot placement is specifically designed to be able to respond quickly to unexpected disturbances in order to stay balanced, but it can be used for walking as well. The ZMP criterion is used to determine where to place the swing foot on the ground. Hereto, a set of ZMP trajectories are computed for a given current state of the robot. These trajectories must satisfy the ZMP criterion and ensure that the COM trajectory does not diverge, otherwise the robot will fall over. The trajectories are computed using preview control with the LIPM, see for reference [29]. An optimization is performed to obtain the optimal pair of ZMP and COM trajectories according to a cost function. The whole process is fast enough that it can be executed online in the control loop, where a feedback controller tracks the generated trajectories in real-time. The control method is validated on the bipedal robot HRP3L-JSK, which is shown in Figure 2.4b. The results are impressive, although it should be noted that the proposed method requires relatively powerful actuators to track the generated trajectories. For this reason, the robot is fitted with customized, liquid-cooled DC motors.

Another bipedal robot that shows impressive balancing and walking capabilities is the humanoid robot ATLAS, see Figure 2.1a. This robot has been developed by the company Boston Dynamics. The founder of Boston Dynamics, Marc Raibert, is also the author of the book [43]. Boston Dynamics is not very open about their control methods. However, it is reasonable to assume that the general control approach that is presented in [43] is used as a foundation for their current control methods.

### 2.3.2 Walking

Recall that the aforementioned method by Urata et al. is based on the ZMP criterion. It turns out that ZMP-based walking is a commonly used form of walking. In fact, out of all the unconstrained bipedal robots that are featured in this chapter, the ZMP criterion is used to control most of them, including Sarcos Primus, ASIMO, HRP-4, TORO, HRP3L-JSK, and TUlip. The specific use of the ZMP

criterion on the humanoid robot TUlip is described in [10, 57]. The ZMP criterion can be used to create a range of desired motions for a legged robot. The desired motion can then be tracked by a suitable controller. However, note that ZMP-based control will always result in flat-footed walking. Furthermore, the precomputed gaits often result in poor robustness against unexpected disturbances [8].

Some research has focused on the generation of ZMP trajectories in real-time. For example in [29], Kajita et al. have used preview control with the LIPM to control the motion of the COM of a bipedal robot. Preview control has similarities with model predictive control (MPC). In this case, the jerk of the COM is minimized over a finite prediction horizon. A stable walking gait is obtained by solving the optimization problem online. In addition, a desired foot placement location is provided by a separate "foot step planner". In [67], Wieber has extended the preview control method so that the constraints on the COP can be taken into account explicitly. This has resulted in a more general linear MPC scheme with an improved ability to reject disturbances during walking. Lastly, in [11] the existing MPC scheme is slightly modified to enable continuous adjustments of the desired foot placement location.

A different approach to achieve walking, which does not involve the ZMP criterion, is to try to decouple the dynamics in the frontal plane from the dynamics in the sagittal plane. The motivation behind this is that walking can be done already in the sagittal plane, see for example [64]. Therefore, an existing controller can be used to obtain an asymptotically stable periodic walking gait in the sagittal plane. However, a separate controller must be designed to stabilize the motion in the frontal plane. In [12], Fukuda et al. have taken an approach that is similar to the one that is described in [64] to control the motion of a bipedal robot in the sagittal plane. Subsequently, the sideways rocking motion that is inherent to bipedal walking, is controlled using the inverted pendulum model in the frontal plane. Furthermore, in [13] an event-based controller is proposed to synchronize the motion in the two planes. Sinnet and Ames show in [45] that the functional Routhian reduction can be used to decouple the dynamics of a fully actuated bipedal robot. In [18], an additional controller is designed for the case that there are underactuated phases of walking. Finally, in [46] it is shown how this reduction-based control can be combined with human-inspired walking control, of which the latter has been mentioned already in Section 2.2.2.

Unconstrained walking can also be achieved without decoupling the dynamics. For this, Song and Zefran have presented a general computational framework for the stabilization of periodic orbits of nonlinear systems with impulsive effects [48, 49]. The framework has been validated with simulations of unconstrained bipedal robots with more than one degree of underactuation. In [44], Grizzle et al. have designed a state feedback control law with an asymptotically stable periodic gait that results in so-called directional dynamic walking. The controller is designed for an unconstrained bipedal robot with point feet that consists of three links. An extension of the control approach described in [64] is used to compute a periodic orbit and to determine a control law that stabilizes that orbit. A linearized Poincare map is used to show that the periodic orbit of the closed-loop system is exponentially stable. Furthermore, in [6] it is shown that directional dynamic walking can produce walking gaits that are input-to-state stable for sufficiently small steering angles.

## 2.4 Conclusions

The control of bipedal robots is a difficult problem. That is why a significant amount of research is comprised of creative control solutions that are based on relatively simple models of bipedal robots. One of the difficulties in the control of bipedal robots is that such robots need to repeatedly change their contact with the ground in order to achieve walking. Clearly, the dynamics of a walking bipedal robot is non-smooth because of these repeated changes in the contact with the ground, and also because of the collisions that occur between the swing foot and the ground at every step along the way. Another difficulty is that a bipedal robot, which is supposed to be fully actuated, can still become underactuated if the stance feet do not remain entirely flat on the ground in the presence of disturbances. As a result, the control methods for fully actuated robots are not well-suited for bipedal robots. Fortunately, the ZMP criterion provides the conditions for which the stance feet of a legged robot are at rest and in full contact with the ground, so that the robot is not underactuated. As long as the walking gait is designed to satisfy the ZMP criterion, then the control methods for fully actuated robots can still be used for the control of bipedal robots.

A controller that is based on the control theory of non-smooth dynamical systems cannot be readily applied to bipedal robots either. For example, the notion of hybrid zero dynamics has been introduced specifically for the control of underactuated bipedal robots that are modeled in the framework of hybrid dynamical systems, or systems with impulsive effects. Nonetheless, the benefit of such a control approach is that the desired walking gait can be defined as a function of the state of the robot, which makes the closed-loop system inherently more robust against disturbances. Another benefit is that this control approach can be applied to both fully actuated bipedal robots and underactuated bipedal robots. Thus, it is not necessary to make sure that the feet remain flat on the ground at all times during walking. However, the control design and the stability analysis is more complicated if there is more than one degree of underactuation, such as in the case of unconstrained bipedal robots with point feet.

A possible shortcoming of the above control approach is that it can only be used for continuous walking, while it is often desirable to let the robot walk to a specific location instead. However, this desirable behavior cannot easily be achieved if walking is represented as a periodic orbit in the state space of the robot. Thus, herein lies an opportunity to contribute to the field of bipedal robot control. Let us take the existing control approach that consists of a feedback linearization with output functions that represent virtual constraints on the configuration of the robot, so that the desired walking gait is defined as a function of the state of the robot. The main difference is that the control design is done in such a way that the bipedal robot is stabilized to a desired location instead of a periodic orbit. Note that this necessarily restricts our attention to fully actuated bipedal robots, because underactuated bipedal robots cannot come to a standstill at a desired location.

In addition, it is observed that some control methods require relatively powerful actuators to keep the bipedal robot balanced. Many bipedal robots are also quite heavy and need powerful actuators to provide the necessary vertical support. The need for powerful actuators is an issue when it comes to energy efficiency, because bipedal robots are only useful if they can operate for long periods of time without a tether. The amount of torque that is applied by the controller must be carefully considered as well. For example, if the controller applies too much torque, then the robot can fall over due to a loss of contact with the ground. On the other hand, if the controller does not apply enough torque, then the robot might not be able to respond quickly enough to stay balanced after a push, or it might not be able to track the desired walking gait during relatively fast walking. Therefore, it makes sense to also have controller bounds in the control design.

# Chapter 3

# Modeling a bipedal robot in the sagittal plane

In this chapter, a specific bipedal robot is considered that is constrained to the sagittal plane. A model of this robot is created by first formulating the multibody dynamics of the robot. Then, a state-reset law is introduced to model the collisions between the swing foot and the ground. Lastly, the hybrid systems framework is used to combine both parts into one complete model that describes the non-smooth dynamics of the robot. The resulting model is used for the control design and the subsequent analysis in the following chapters.

## 3.1 A fully actuated planar bipedal robot

Consider the planar bipedal robot that is depicted in Figure 3.1. The robot consists of 5 rigid bodies that are interconnected by revolute joints, whereby the torso body is mechanically constrained so that it cannot rotate. This allows the robot to be fully actuated in a single support mode, as will become clear later on. A list with the names of the rigid bodies and joints is presented in Table 3.1. The abbreviations listed in this table are also used in the figure. Notice that the robot has point feet, which are represented by the points Clf,lsh and Crf,rsh in the figure. If a foot is in contact with the ground, then it is modeled as a frictionless revolute joint. By the way, if it is said that a foot is in contact with the ground, then by default this means non-sliding contact, unless explicitly stated otherwise. For convenience let us define the set of rigid bodies $\mathcal{B} = \{\mathrm{tr}, \mathrm{lth}, \mathrm{lsh}, \mathrm{rth}, \mathrm{rsh}\}$ and the set of joints $\mathcal{J} = \{\mathrm{lh}, \mathrm{lk}, \mathrm{lf}, \mathrm{rh}, \mathrm{rk}, \mathrm{rf}\}$. The location of the center of mass of a body is denoted by CM$j$, where $j \in \mathcal{B}$. The location of a joint on a rigid body is denoted by C$i$,$j$, where $i \in \mathcal{J}$ and $j \in \mathcal{B}$. Each body has two joints, which are assumed to be located at an equal distance from the body's center of mass.

In Figure 3.1 the right foot is in contact with the ground. The overall dynamics of the robot is different depending on which foot is in contact with the ground. It is useful to distinguish between these so-called modes of the robot. Therefore, consider the set of modes that is presented in Table 3.2. The word "pinned" is used here to emphasize that a foot that is in contact with the ground, is modeled as a revolute joint and not merely as a contact point. A revolute joint is a bilateral constraint between two bodies, whereas a contact point is a unilateral constraint between two bodies. Let $\sigma$ denote the current mode of the robot, and consider that $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$ during walking. Hereby, it is assumed that during walking the stance foot lifts off the ground at the same time that the swing foot collides with the ground. The stance foot can lift off the ground either by itself as a direct result of the collision between the swing foot and the ground, or with the help of the controller that forces the stance foot to lift off the ground immediately after the swing foot has collided with the ground.

It is assumed that the state of the robot $\boldsymbol{x}$ can be measured or at least accurately estimated. This includes the absolute position and velocity of the robot relative to an inertial frame. In addition, it is assumed that the surface geometry of the ground is known. In this way the elevation of the swing foot can be determined relative to the ground. Here, a flat and level surface will be used.

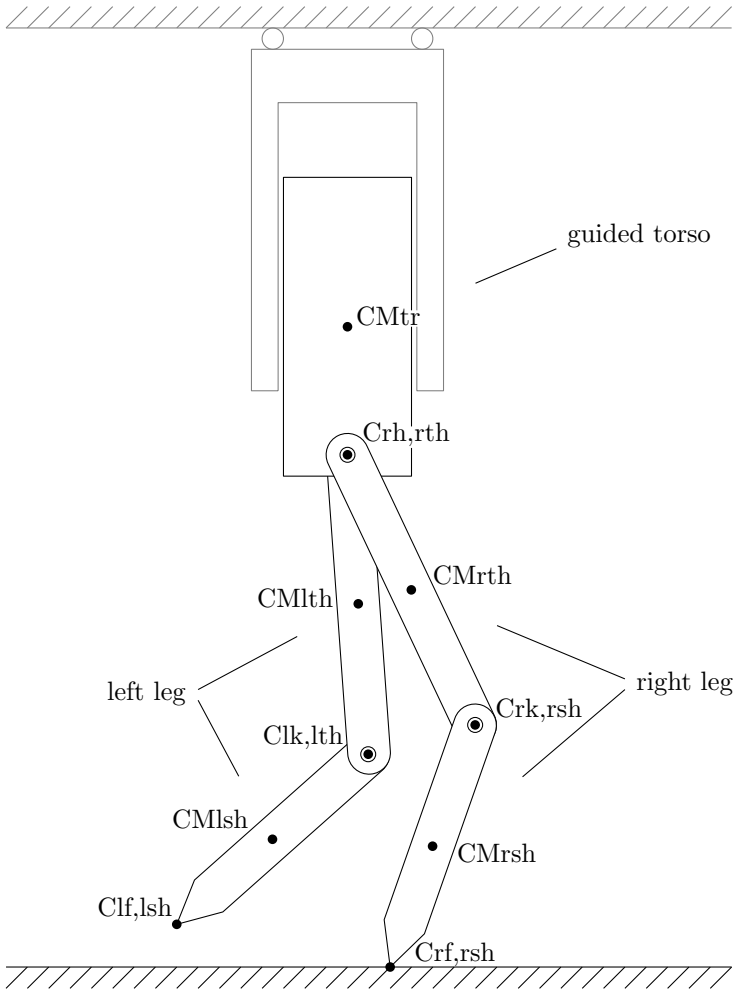



Figure 3.1: An illustration of the planar bipedal robot.

| body | abbreviation |
|---|---|
| torso | tr |
| left thigh | lth |
| left shank | lsh |
| right thigh | rth |
| right shank | rsh |

| joint | abbreviation |
|---|---|
| left hip | lh |
| left knee | lk |
| left foot | lf |
| right hip | rh |
| right knee | rk |
| right foot | rf |

Table 3.1: The names of the rigid bodies and joints.

Consider that both hip joints and both knee joints are actuated. Let the torque that is applied to each joint be denoted by $\tau_i$, where $i \in \mathcal{J} \setminus \{\mathrm{lf}, \mathrm{rf}\} = \{\mathrm{lh}, \mathrm{lk}, \mathrm{rh}, \mathrm{rk}\}$. For example, the actuator at the left hip joint applies a torque $\tau_{\mathrm{lh}}$ to the left thigh and a reaction torque to the torso. Similarly, the actuator at the right hip joint applies a torque $\tau_{\mathrm{rh}}$ to the right thigh and a reaction torque to the torso. Notice that these reaction torques have to be counteracted by a kinematic constraint on the orientation of the torso in order to keep the torso upright at all times.

The applied torques are collected in the vector of applied torques

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{\mathrm{lh}} & \tau_{\mathrm{lk}} & \tau_{\mathrm{rh}} & \tau_{\mathrm{rk}} \end{bmatrix}^\top. \tag{3.1}$$

Naturally, there is a limit to the amount of torque that an actuator can apply. Therefore, consider that the torque $\tau_i$ saturates at a given maximum torque $L_i > 0$, such that $|\tau_i| \leq L_i$. To express this in a useful way, let us define the saturation function

$$\mathrm{sat}_\kappa(\nu) = \begin{cases} \nu, & |\nu| \leq \kappa \\ \kappa \,\mathrm{sign}(\nu), & |\nu| > \kappa, \end{cases} \tag{3.2}$$

where the constant $\kappa > 0$ and the signum function

$$\mathrm{sign}(\nu) = \begin{cases} -1, & \nu < 0 \\ 0, & \nu = 0 \\ 1, & \nu > 0. \end{cases} \tag{3.3}$$

| mode | abbreviation | description |
|---|---|---|
| left foot pinned | lfp | the left foot is in contact with the ground |
| right foot pinned | rfp | the right foot is in contact with the ground |
| both feet pinned | bfp | both feet are in contact with the ground |
| neither foot pinned | nfp | neither foot is in contact with the ground |

Table 3.2: The different modes that the bipedal robot can be in.

| body | length [m] | mass [kg] | moment of inertia, w.r.t. the center of mass [kgm$^2$] |
|---|---|---|---|
| torso | $l_{\mathrm{tr}}$ 0.4 | $m_{\mathrm{tr}}$ 10 | $J_{\mathrm{tr}}$ 0.25 |
| (left or right) thigh | $l_{\mathrm{th}}$ 0.4 | $m_{\mathrm{th}}$ 3.2 | $J_{\mathrm{th}}$ 0.016 |
| (left or right) shank | $l_{\mathrm{sh}}$ 0.3 | $m_{\mathrm{sh}}$ 0.9 | $J_{\mathrm{sh}}$ 0.007 |

| joint | maximum torque [Nm] | damping coefficient [Nms] |
|---|---|---|
| (left or right) hip | $L_{\mathrm{h}}$ 80 | $d_{\mathrm{h}}$ 5 |
| (left or right) knee | $L_{\mathrm{k}}$ 80 | $d_{\mathrm{k}}$ 5 |

Table 3.3: The parameters of the planar bipedal robot.

Given the vectors $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_N)$ and $\boldsymbol{\kappa} = (\kappa_1, \ldots, \kappa_N) > \mathbf{0}$, let us also define the vector-valued saturation function $\mathbf{sat}_{\boldsymbol{\kappa}}(\boldsymbol{\nu}) = (\mathrm{sat}_{\kappa_1}(\nu_1), \ldots, \mathrm{sat}_{\kappa_N}(\nu_N))$. Now, observe that the vector of applied torques saturates according to

$$\boldsymbol{\tau} = \mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u}), \tag{3.4}$$

where $\boldsymbol{L} = (L_{\mathrm{h}}, L_{\mathrm{k}}, L_{\mathrm{h}}, L_{\mathrm{k}})$ and $\boldsymbol{u}$ is the vector of input torques, or simply the input vector.

In addition, it is assumed that friction is present in all actuated joints. This friction is modeled as linear viscous damping. The damping coefficients are given in Table 3.3 together with the other parameters of the robot. The link lengths and inertial properties of the links are based on the humanoid robot TUlip of the Eindhoven University of Technology.

To physically constrain the motion of the robot to the sagittal plane, one can construct a guiding frame for the torso as sketched in Figure 3.1, and let the robot walk on a treadmill. The guiding frame also removes the rotational degree of freedom from the torso. However, a more common approach of constraining the motion of the robot to the sagittal plane is to use a lateral support bar. This bar is connected at one end to the torso in a way that the bar is pointing out of the sagittal plane, while the other end of the bar is pinned to the ground so that the robot walks in circles. This effectively maps the plane of motion onto the face of a cylinder. The effect it has on the dynamics of the robot in the sagittal plane can be neglected for bipedal robots with point feet, especially if the feet are free to roll on the ground in the lateral direction, as shown in Figure 2.3a for example. To constrain the orientation of the torso, one has to remove one additional rotational degree of freedom at the joint between the torso and the lateral support bar. In any case, the orientation of the torso must be constrained so that the torso remains upright at all times. This ensures that the robot is always fully actuated for the modes $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$, as is formally shown in the next section.

## 3.2 The multibody dynamics of the robot

The multibody dynamics is formulated using $n$ generalized coordinates, whereby the position and orientation of each rigid body is described with respect to one inertial frame. Thus, all generalized coordinates are absolute coordinates. This is illustrated in Figure 3.2, where $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are the unit vectors of the inertial frame, $\boldsymbol{r}_{\mathrm{CM}j} = x_j \boldsymbol{e}_1 + y_j \boldsymbol{e}_2$ is the position vector of the center of mass of body $j \in \mathcal{B}$, $\boldsymbol{r}_{\mathrm{swf}} = x_{\mathrm{swf}} \boldsymbol{e}_1 + y_{\mathrm{swf}} \boldsymbol{e}_2$ is the position vector of the swing foot, and $\boldsymbol{r}_{\mathrm{stf}} = x_{\mathrm{stf}} \boldsymbol{e}_1 + y_{\mathrm{stf}} \boldsymbol{e}_2$ is the position vector of the stance foot. Consider that the vector of generalized coordinates is given by

$$\boldsymbol{q} = \begin{bmatrix} x_{\mathrm{tr}} & y_{\mathrm{tr}} \,\vdots\, x_{\mathrm{lth}} & y_{\mathrm{lth}} & \theta_{\mathrm{lth}} \,\vdots\, x_{\mathrm{lsh}} & y_{\mathrm{lsh}} & \theta_{\mathrm{lsh}} \,\vdots\, x_{\mathrm{rth}} & y_{\mathrm{rth}} & \theta_{\mathrm{rth}} \,\vdots\, x_{\mathrm{rsh}} & y_{\mathrm{rsh}} & \theta_{\mathrm{rsh}} \end{bmatrix}^{\top}, \tag{3.5}$$

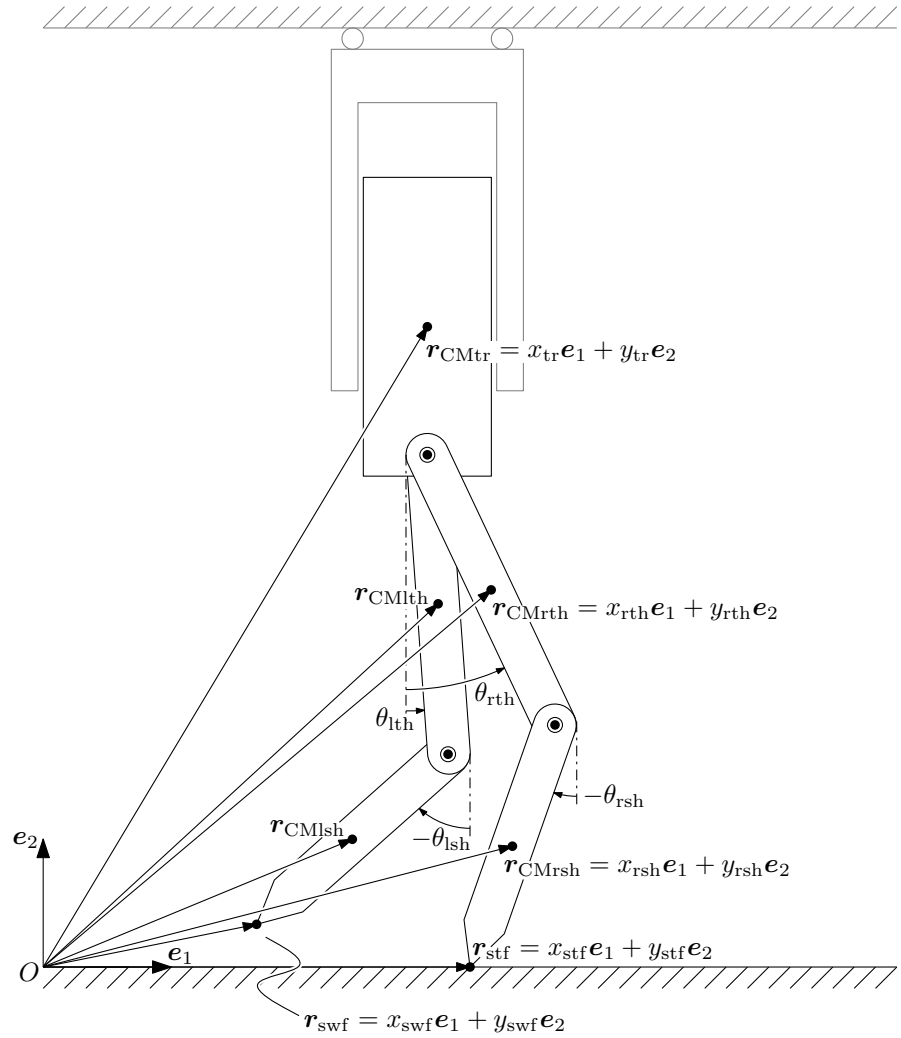



Figure 3.2: The absolute coordinates of the multibody system.

and that the state vector is given by $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$. The position coordinates of the swing foot $(x_{\mathrm{swf}}, y_{\mathrm{swf}})$ as well as the position coordinates of the stance foot $(x_{\mathrm{stf}}, y_{\mathrm{stf}})$ can be expressed in terms of the generalized coordinates $\boldsymbol{q}$. Hereby, let the position of the swing foot $(x_{\mathrm{swf}}, y_{\mathrm{swf}}) = (x_{\mathrm{swf},\sigma}(\boldsymbol{q}), y_{\mathrm{swf},\sigma}(\boldsymbol{q}))$, with

$$x_{\mathrm{swf},\sigma}(\boldsymbol{q}) = \begin{cases} x_{\mathrm{rsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ x_{\mathrm{lsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases} \tag{3.6}$$

$$y_{\mathrm{swf},\sigma}(\boldsymbol{q}) = \begin{cases} y_{\mathrm{rsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ y_{\mathrm{lsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases} \tag{3.7}$$

and the position of the stance foot $(x_{\mathrm{stf}}, y_{\mathrm{stf}}) = (x_{\mathrm{stf},\sigma}(\boldsymbol{q}), y_{\mathrm{stf},\sigma}(\boldsymbol{q}))$, with

$$x_{\mathrm{stf},\sigma}(\boldsymbol{q}) = \begin{cases} x_{\mathrm{lsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{lfp} \\ x_{\mathrm{rsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{rfp}, \end{cases} \tag{3.8}$$

$$y_{\mathrm{stf},\sigma}(\boldsymbol{q}) = \begin{cases} y_{\mathrm{lsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{lfp} \\ y_{\mathrm{rsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{rfp}, \end{cases} \tag{3.9}$$

where $l_{\mathrm{sh}}$ is the length of the shank.

The equations of motion of the multibody system are derived using the well-known Newton-Euler equations. First, consider the free body diagram shown in Figure 3.3, where $g_{\mathrm{a}}$ denotes the gravitational

acceleration, and $\lambda_{i1}, \lambda_{i2}$ with $i \in \mathcal{J}$ denote the kinematic constraint forces of the constraints that are associated with the revolute joints. The other symbols that are used in the figure have been introduced in the previous section. The Newton-Euler equations are used together with the free body diagram to obtain the following $n = 14$ equations of motion for the mode $\sigma =$ rfp:

$$\text{torso:} \quad \begin{cases} m_{\mathrm{tr}}\ddot{x}_{\mathrm{tr}} = -\lambda_{\mathrm{lh1}} - \lambda_{\mathrm{rh1}} \\ m_{\mathrm{tr}}\ddot{y}_{\mathrm{tr}} = -m_{\mathrm{tr}}g_{\mathrm{a}} - \lambda_{\mathrm{lh2}} - \lambda_{\mathrm{rh2}}, \end{cases} \tag{3.10}$$

$$\text{left thigh:} \quad \begin{cases} m_{\mathrm{th}}\ddot{x}_{\mathrm{lth}} = \lambda_{\mathrm{lh1}} - \lambda_{\mathrm{lk1}} \\ m_{\mathrm{th}}\ddot{y}_{\mathrm{lth}} = -m_{\mathrm{th}}g_{\mathrm{a}} + \lambda_{\mathrm{lh2}} - \lambda_{\mathrm{lk2}} \\ J_{\mathrm{th}}\ddot{\theta}_{\mathrm{lth}} = \tau_{\mathrm{lh}} - \tau_{\mathrm{lk}} - d_{\mathrm{h}}\dot{\theta}_{\mathrm{lth}} + d_{\mathrm{k}}\left(\dot{\theta}_{\mathrm{lsh}} - \dot{\theta}_{\mathrm{lth}}\right) \\ \qquad\qquad - (\lambda_{\mathrm{lh1}} + \lambda_{\mathrm{lk1}})\,\tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{lth}} - (\lambda_{\mathrm{lh2}} + \lambda_{\mathrm{lk2}})\,\tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{lth}}, \end{cases} \tag{3.11}$$

$$\text{left shank:} \quad \begin{cases} m_{\mathrm{sh}}\ddot{x}_{\mathrm{lsh}} = \lambda_{\mathrm{lk1}} \\ m_{\mathrm{sh}}\ddot{y}_{\mathrm{lsh}} = -m_{\mathrm{sh}}g_{\mathrm{a}} + \lambda_{\mathrm{lk2}} \\ J_{\mathrm{sh}}\ddot{\theta}_{\mathrm{lsh}} = \tau_{\mathrm{lk}} - d_{\mathrm{k}}\left(\dot{\theta}_{\mathrm{lsh}} - \dot{\theta}_{\mathrm{lth}}\right) - \lambda_{\mathrm{lk1}}\tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{lsh}} - \lambda_{\mathrm{lk2}}\tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{lsh}}, \end{cases} \tag{3.12}$$

$$\text{right thigh:} \quad \begin{cases} m_{\mathrm{th}}\ddot{x}_{\mathrm{rth}} = \lambda_{\mathrm{rh1}} - \lambda_{\mathrm{rk1}} \\ m_{\mathrm{th}}\ddot{y}_{\mathrm{rth}} = -m_{\mathrm{th}}g_{\mathrm{a}} + \lambda_{\mathrm{rh2}} - \lambda_{\mathrm{rk2}} \\ J_{\mathrm{th}}\ddot{\theta}_{\mathrm{rth}} = \tau_{\mathrm{rh}} - \tau_{\mathrm{rk}} - d_{\mathrm{h}}\dot{\theta}_{\mathrm{rth}} + d_{\mathrm{k}}\left(\dot{\theta}_{\mathrm{rsh}} - \dot{\theta}_{\mathrm{rth}}\right) \\ \qquad\qquad - (\lambda_{\mathrm{rh1}} + \lambda_{\mathrm{rk1}})\,\tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{rth}} - (\lambda_{\mathrm{rh2}} + \lambda_{\mathrm{rk2}})\,\tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{rth}}, \end{cases} \tag{3.13}$$

$$\text{right shank:} \quad \begin{cases} m_{\mathrm{sh}}\ddot{x}_{\mathrm{rsh}} = \lambda_{\mathrm{rk1}} + \lambda_{\mathrm{rf1}} \\ m_{\mathrm{sh}}\ddot{y}_{\mathrm{rsh}} = -m_{\mathrm{sh}}g_{\mathrm{a}} + \lambda_{\mathrm{rk2}} + \lambda_{\mathrm{rf2}} \\ J_{\mathrm{sh}}\ddot{\theta}_{\mathrm{rsh}} = \tau_{\mathrm{rk}} - d_{\mathrm{k}}\left(\dot{\theta}_{\mathrm{rsh}} - \dot{\theta}_{\mathrm{rth}}\right) - \lambda_{\mathrm{rk1}}\tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}} - \lambda_{\mathrm{rk2}}\tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}} \\ \qquad\qquad + \lambda_{\mathrm{rf1}}\tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}} + \lambda_{\mathrm{rf2}}\tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}}. \end{cases} \tag{3.14}$$

The constraints that are associated with the revolute joints are described by $s = 10$ additional equations. These so-called kinematic constraint equations are given here for the mode $\sigma =$ rfp as well:

$$\text{left hip:} \quad \begin{cases} x_{\mathrm{tr}} = x_{\mathrm{lth}} - \tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{lth}} \\ y_{\mathrm{tr}} - \tfrac{1}{2}l_{\mathrm{tr}} = y_{\mathrm{lth}} + \tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{lth}}, \end{cases} \tag{3.15}$$

$$\text{left knee:} \quad \begin{cases} x_{\mathrm{lth}} + \tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{lth}} = x_{\mathrm{lsh}} - \tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{lsh}} \\ y_{\mathrm{lth}} - \tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{lth}} = y_{\mathrm{lsh}} + \tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{lsh}}, \end{cases} \tag{3.16}$$

$$\text{right hip:} \quad \begin{cases} x_{\mathrm{tr}} = x_{\mathrm{rth}} - \tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{rth}} \\ y_{\mathrm{tr}} - \tfrac{1}{2}l_{\mathrm{tr}} = y_{\mathrm{rth}} + \tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{rth}}, \end{cases} \tag{3.17}$$

$$\text{right knee:} \quad \begin{cases} x_{\mathrm{rth}} + \tfrac{1}{2}l_{\mathrm{th}}\sin\theta_{\mathrm{rth}} = x_{\mathrm{rsh}} - \tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}} \\ y_{\mathrm{rth}} - \tfrac{1}{2}l_{\mathrm{th}}\cos\theta_{\mathrm{rth}} = y_{\mathrm{rsh}} + \tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}}, \end{cases} \tag{3.18}$$

$$\text{right foot:} \quad \begin{cases} x_{\mathrm{rsh}} + \tfrac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}} = x_{\mathrm{stf}} \\ y_{\mathrm{rsh}} - \tfrac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}} = y_{\mathrm{stf}}. \end{cases} \tag{3.19}$$

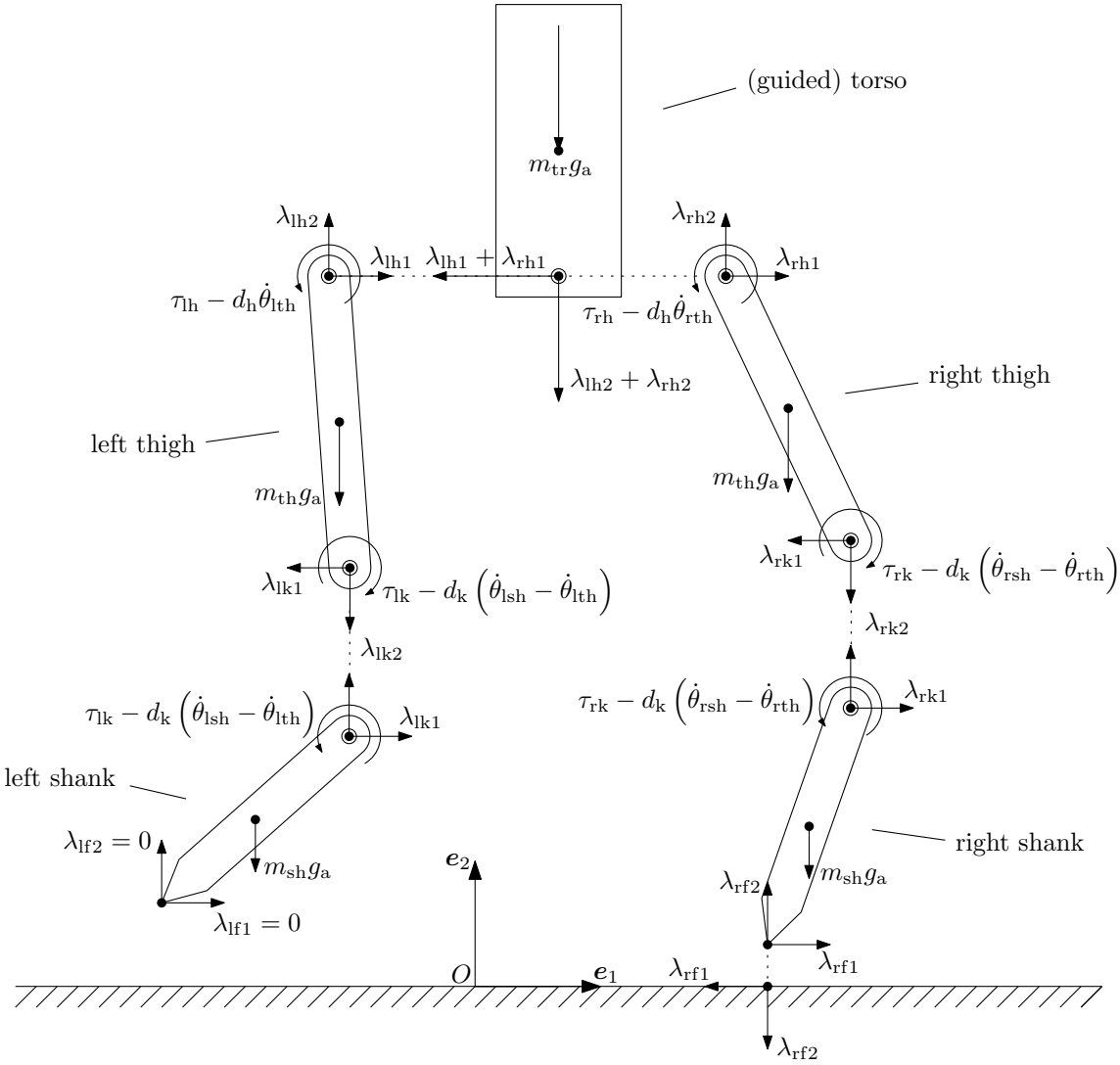



Figure 3.3: The free body diagram for the mode $\sigma = \mathrm{rfp}$.

The equations of motion and the kinematic constraint equations can also be derived for the mode $\sigma = \mathrm{lfp}$ in a similar way. At this point it becomes useful to write both sets of equations in matrix form. However, in order to do so, it is necessary to define the vector of kinematic constraint forces

$$\boldsymbol{\lambda}_\sigma = \begin{bmatrix} \boldsymbol{\lambda} \\ F_{\mathrm{f},\sigma} \\ F_{\mathrm{n},\sigma} \end{bmatrix}, \tag{3.20}$$

where

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_{\mathrm{lh1}} & \lambda_{\mathrm{lh2}} & \lambda_{\mathrm{lk1}} & \lambda_{\mathrm{lk2}} & \lambda_{\mathrm{rh1}} & \lambda_{\mathrm{rh2}} & \lambda_{\mathrm{rk1}} & \lambda_{\mathrm{rk2}} \end{bmatrix}^\top,$$

the friction contact force

$$F_{\mathrm{f},\sigma} = \begin{cases} \lambda_{\mathrm{lf1}}, & \sigma = \mathrm{lfp} \\ \lambda_{\mathrm{rf1}}, & \sigma = \mathrm{rfp}, \end{cases} \tag{3.21}$$

and the normal contact force

$$F_{\mathrm{n},\sigma} = \begin{cases} \lambda_{\mathrm{lf2}}, & \sigma = \mathrm{lfp} \\ \lambda_{\mathrm{rf2}}, & \sigma = \mathrm{rfp}. \end{cases} \tag{3.22}$$

Now, the equations of motion can be written in matrix form as

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{H}(\dot{\boldsymbol{q}}) = \boldsymbol{S}\boldsymbol{\tau} + \boldsymbol{W}_\sigma(\boldsymbol{q})\boldsymbol{\lambda}_\sigma, \tag{3.23}$$

for $\sigma \in \{\text{lfp}, \text{rfp}\}$, where $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ is the inertia matrix, $\boldsymbol{H}\colon \mathbb{R}^n \to \mathbb{R}^n$ contains the gravitational terms and the damping terms, $\boldsymbol{S} \in \mathbb{R}^{n\times p}$ represents the generalized force directions of the applied torques $\boldsymbol{\tau}$, and $\boldsymbol{W}_\sigma\colon \mathbb{R}^n \to \mathbb{R}^{n\times s}$ is the constraint Jacobian that also represents the generalized force directions of the kinematic constraint forces $\boldsymbol{\lambda}_\sigma$. The number of generalized coordinates $n = \dim(\boldsymbol{q}) = 14$, the number of actuators $p = \dim(\boldsymbol{\tau}) = 4$, and the number of kinematic constraints $s = \dim(\boldsymbol{\lambda}_\sigma) = 10$. The kinematic constraint equations are expressed in vector form as $\boldsymbol{h}_\sigma(\boldsymbol{q}) = \boldsymbol{0}$ for $\sigma \in \{\text{lfp}, \text{rfp}\}$, where $\boldsymbol{h}_\sigma\colon \mathbb{R}^n \to \mathbb{R}^s$. Hereby, it holds that

$$\dot{\boldsymbol{h}}_\sigma = \frac{\partial \boldsymbol{h}_\sigma(\boldsymbol{q})}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} = \boldsymbol{W}_\sigma^\top(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0}, \tag{3.24}$$

$$\ddot{\boldsymbol{h}}_\sigma = \boldsymbol{W}_\sigma^\top(\boldsymbol{q})\ddot{\boldsymbol{q}} + \frac{\partial \boldsymbol{W}_\sigma^\top(\boldsymbol{q})\dot{\boldsymbol{q}}}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} \coloneqq \boldsymbol{W}_\sigma^\top(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{0}. \tag{3.25}$$

The multibody dynamics of the robot is described by (3.23) and (3.25), which can be combined into a single matrix equation

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{W}_\sigma(\boldsymbol{q}) \\ \boldsymbol{W}_\sigma^\top(\boldsymbol{q}) & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ \boldsymbol{\lambda}_\sigma \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}\boldsymbol{\tau} - \boldsymbol{H}(\dot{\boldsymbol{q}}) \\ -\boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix}. \tag{3.26}$$

The matrices and vectors $\boldsymbol{M}$, $\boldsymbol{H}(\dot{\boldsymbol{q}})$, $\boldsymbol{S}$, $\boldsymbol{W}_\sigma(\boldsymbol{q})$, and $\boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}})$ are written out in Appendix A. The differential algebraic equation (3.26) describes the multibody dynamics for the modes $\sigma \in \{\text{lfp}, \text{rfp}\}$, where it is assumed that the stance foot is constrained to the ground as a revolute joint. However, this is not an accurate representation of the unilateral contact constraint that actually exists between the stance foot and the ground. Therefore, let us use Coulomb's model of static friction to check the validity of the contact constraint at any given moment in time. Coulomb's model is given by

$$|F_{\text{f},\sigma}| \leq \mu F_{\text{n},\sigma}, \tag{3.27}$$

where $F_{\text{f},\sigma}$ and $F_{\text{n},\sigma}$ are the contact forces, and $\mu > 0$ is the coefficient of static friction. As long as $F_{\text{f},\sigma}$ and $F_{\text{n},\sigma}$ satisfy (3.27), then the system (3.26) accurately describes the multibody dynamics for $\sigma \in \{\text{lfp}, \text{rfp}\}$. If (3.27) is violated, then the results cannot be trusted to be physically accurate. One reason is that, in reality, the stance foot is only unilaterally constrained to the ground, and not bilaterally as with a revolute joint. Note that (3.27) can be violated in two particular ways: $|F_{\text{f},\sigma}| > \mu F_{\text{n},\sigma} > 0$ , i.e. sliding contact, and $F_{\text{n},\sigma} < 0$, i.e. loss of contact. The normal contact force (3.22) can become negative because it is a kinematic constraint force of a revolute joint, which forms a bilateral constraint between the stance foot and the ground. By the way, it is assumed that the coefficient of static friction $\mu = 1$.

With regard to solving (3.26) for $\ddot{\boldsymbol{q}}$ and $\boldsymbol{\lambda}_\sigma$, observe that

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{W}_\sigma(\boldsymbol{q}) \\ \boldsymbol{W}_\sigma^\top(\boldsymbol{q}) & \boldsymbol{0} \end{bmatrix}^{-1} = \\ \begin{bmatrix} \boldsymbol{M}^{-1} - \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1}\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1} & \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1} \\ -\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1}\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1} & \left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1} \end{bmatrix}, \tag{3.28}$$

where the argument of $\boldsymbol{W}_\sigma(\boldsymbol{q})$ has been omitted for the sake of brevity and readability. Note that the inertia matrix $\boldsymbol{M}$ is diagonal and positive definite. It follows that its inverse $\boldsymbol{M}^{-1}$ is diagonal and positive definite as well. In addition, the constraint Jacobian $\boldsymbol{W}_\sigma(\boldsymbol{q})$ has full column rank for all $\boldsymbol{q}$ and for both modes $\sigma \in \{\text{lfp}, \text{rfp}\}$. Given the positive definite matrix $\boldsymbol{M}^{-1}$ and the full rank matrix $\boldsymbol{W}_\sigma(\boldsymbol{q})$, it can be shown that the matrix $\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma$ is positive definite, see for example [23, Observation 7.1.8]. As a result, the matrix inverse (3.28) exists for all $\boldsymbol{q}$ and for both modes $\sigma$. For convenience, let us define

$$\begin{bmatrix} \boldsymbol{A}_{\sigma,11}(\boldsymbol{q}) & \boldsymbol{A}_{\sigma,12}(\boldsymbol{q}) \\ \boldsymbol{A}_{\sigma,21}(\boldsymbol{q}) & \boldsymbol{A}_{\sigma,22}(\boldsymbol{q}) \end{bmatrix} = \\ \begin{bmatrix} \boldsymbol{M}^{-1} - \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1}\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1} & \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1} \\ -\left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1}\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1} & \left(\boldsymbol{W}_\sigma^\top\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma\right)^{-1} \end{bmatrix}. \tag{3.29}$$

Finally, the equations of motion can be written explicitly as

$$\ddot{\boldsymbol{q}} = \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\left(\boldsymbol{S}\boldsymbol{\tau} - \boldsymbol{H}(\dot{\boldsymbol{q}})\right) - \boldsymbol{A}_{\sigma,12}(\boldsymbol{q})\boldsymbol{w}_{\sigma}(\boldsymbol{q},\dot{\boldsymbol{q}}), \tag{3.30}$$

which is now independent of the kinematic constraint forces $\boldsymbol{\lambda}_{\sigma}$.

Let $N_{\mathrm{c}}$ denote the total number of constraints on the robot and let $N_{\mathrm{b}}$ denote the number of rigid bodies. According to the rule of Grübler [58], the number of degrees of freedom $N_{\mathrm{dof}} = 3N_{\mathrm{b}} - N_{\mathrm{c}}$, for a planar multibody system. It is clear that the number of rigid bodies $N_{\mathrm{b}} = 5$. The total number of constraints $N_{\mathrm{c}} = s + 1$, because there are $s = 10$ kinematic constraint equations and there is one additional kinematic constraint on the orientation of the torso. The latter has been taken into account already by the absence of a coordinate $\theta_{\mathrm{tr}}$ in the multibody dynamics. Consequently, the number of degrees of freedom $N_{\mathrm{dof}} = 3 \cdot 5 - (10 + 1) = 4$ for $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$, which is equal to the number of actuators $p = 4$. Hence, the robot is fully actuated in a single support mode, i.e. for $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$. This statement is true regardless of the state of the robot $\boldsymbol{x}$. Thus, the bipedal robot is always fully actuated during walking as long as the contact forces satisfy Coulomb's model of static friction. The control of this relatively simple bipedal robot is useful, because it can give insights in how to control other, more complex (planar) bipedal robots that are also meant to be fully actuated, but which can become underactuated as well. An example of such a robot would be a planar bipedal robot with actual feet and actuators at the ankle joints, so that it would not need a constraint on the orientation of the torso in order to be fully actuated. This bipedal robot would also be fully actuated in a single support mode as long as the stance foot remains at rest and in full contact with the ground.

The number of degrees of freedom is also equal to the minimum number of generalized coordinates that can completely describe the kinematics of the robot. Such coordinates are referred to as minimal coordinates. Here, the vector of minimal coordinates is given by

$$\boldsymbol{q}_{\mathrm{min}} = \begin{bmatrix} \theta_{\mathrm{lth}} & \theta_{\mathrm{lsh}} & \theta_{\mathrm{rth}} & \theta_{\mathrm{rsh}} \end{bmatrix}^{\top}. \tag{3.31}$$

In some instances it is convenient to use $\boldsymbol{q}_{\mathrm{min}}$ instead of $\boldsymbol{q}$. However, note that if $\boldsymbol{q}_{\mathrm{min}}$ is used, then in addition to $\boldsymbol{q}_{\mathrm{min}}$ a pair of position coordinates is required to describe the position of the robot with respect to the inertial frame. One can use the coordinates of one of the marked points on the robot, such as CM$j$ or C$i$,$j$, where $i \in \mathcal{J}$ and $j \in \mathcal{B}$, as shown in Figure 3.1. Then the complete vector $\boldsymbol{q}$ can be determined from $\boldsymbol{q}_{\mathrm{min}}$ together with the position coordinates of the chosen point on the robot.

## 3.3 A state-reset law for the collisions with the ground

Every time that the swing foot is placed on the ground there is a collision between that foot and the ground. The impacts from these collisions cannot be neglected for bipedal robots, as opposed to legged robots with more than two legs [25]. Still, the time scale of a collision is several orders of magnitude smaller than the time scale of the multibody dynamics of the robot. Therefore, a collision is regarded as an instantaneous event. For a given solution $\boldsymbol{x}(t) = (\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))$, this implies that the configuration of the robot $\boldsymbol{q}(t^-) = \boldsymbol{q}(t^+)$ during a collision, where $t^-$ and $t^+$ are the moments just before and just after the collision, respectively. In this section, let $\boldsymbol{q}(t^-) = \boldsymbol{q}$ and $\boldsymbol{q}(t^+) = \boldsymbol{q}^+$. Similarly, let $\dot{\boldsymbol{q}}(t^-) = \dot{\boldsymbol{q}}$ and $\dot{\boldsymbol{q}}(t^+) = \dot{\boldsymbol{q}}^+$.

It is assumed that collisions are perfectly inelastic, which means that the two colliding bodies stay in contact with each other after the collision. The impact of a perfectly inelastic collision is described by Newton's impact law, where the restitution coefficient is equal to zero [60]. The kinematic constraint equations on the velocity level (3.24) can be used to impose Newton's impact law with a zero restitution for the bipedal robot that is discussed here, where the mode $\sigma$ is used to select the appropriate subset of kinematic constraint equations for a given collision between the swing foot and the ground. Hereby, notice that the swing foot becomes the stance foot at the moment just after the collision, where it must hold that

$$\boldsymbol{W}_{\sigma}^{\top}(\boldsymbol{q})\dot{\boldsymbol{q}}^+ = \boldsymbol{0}, \tag{3.32}$$

where $\boldsymbol{q} = \boldsymbol{q}^+$ and the mode $\sigma$ is specified by the foot that has hit the ground. If the left foot has hit the ground, then this foot is now the stance foot and one must use $\sigma = \mathrm{lfp}$ in (3.32). Conversely,

if the right foot has hit the ground, then one must use $\sigma = \text{rfp}$. At the moment of impact, the other foot, i.e. the former stance foot, is assumed to lift off the ground without interaction. In other words, the vertical component of this foot's velocity is assumed to be positive after impact. This ensures deterministic transitions between the two single support modes $\{\text{lfp}, \text{rfp}\}$. If the vertical velocity of the former stance foot happens to be negative, then it is reset to zero together with the foot that hit the ground. This corresponds to the mode $\sigma = \text{bfp}$ in (3.32), where the constraint Jacobian $\boldsymbol{W}_\sigma(\boldsymbol{q})$ contains all 12 kinematic constraint equations from all 6 joints. The actual mode transition between the two single support modes is still deterministic, because in this case the controller will make sure that the former stance foot lifts off the ground after the collision.

The momentum of the robot is conserved during collisions, which can be expressed as

$$\boldsymbol{M}\dot{\boldsymbol{q}}^+ - \boldsymbol{M}\dot{\boldsymbol{q}} = \boldsymbol{W}_\sigma(\boldsymbol{q})\boldsymbol{\Lambda}_\sigma, \tag{3.33}$$

where $\boldsymbol{\Lambda}_\sigma = \int_{t^-}^{t^+} \boldsymbol{\lambda}_\sigma(t)\,\mathrm{d}t$ [64]. The conservation of momentum (3.33) is combined with the impact equation (3.32) to obtain the matrix equation

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{W}_\sigma(\boldsymbol{q}) \\ \boldsymbol{W}_\sigma^\top(\boldsymbol{q}) & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{q}}^+ \\ \boldsymbol{\Lambda}_\sigma \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}\dot{\boldsymbol{q}} \\ \boldsymbol{0} \end{bmatrix}. \tag{3.34}$$

Observe that (3.34) can be solved for $\dot{\boldsymbol{q}}^+$ and $\boldsymbol{\Lambda}_\sigma$ using (3.28). As a result, the reset law for the generalized velocities $\dot{\boldsymbol{q}}$ can be written as

$$\dot{\boldsymbol{q}}^+ = \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{M}\dot{\boldsymbol{q}}, \tag{3.35}$$

where $\boldsymbol{A}_{\sigma,11}(\boldsymbol{q})$ is defined in (3.29).

## 3.4 The hybrid dynamical system

The bipedal robot is a non-smooth continuous-time dynamical system, where the non-smoothness is manifested as discontinuities, or jumps, in the state $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$. These discontinuities are caused by collisions between the swing foot and the ground that require a reset of the generalized velocities $\dot{\boldsymbol{q}}$. There are several mathematical frameworks that can model non-smooth dynamical systems, see for an overview [20, 32]. Here, the so-called hybrid systems framework is used [14]. In the past, this framework has been used already to model bipedal robots, such as in [60, 64]. Hybrid systems are characterized by the interaction between continuous-time dynamics and discrete events. Obviously, the discrete events are the collisions between the swing foot and the ground.

After a collision the state $\boldsymbol{x}$ must be reset and the mode $\sigma$ must be updated. The new state $\boldsymbol{x}$ and the updated mode $\sigma$ are denoted by $\boldsymbol{x}^+$ and $\sigma^+$, respectively, where $\boldsymbol{x}^+ = (\boldsymbol{q}^+, \dot{\boldsymbol{q}}^+)$. In addition, the control input $\boldsymbol{u}$ is subjected to a torque saturation, such that the vector of applied torques $\boldsymbol{\tau} = \mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u})$. Finally, consider that the non-smooth dynamics of the bipedal robot for the modes $\sigma \in \{\text{lfp}, \text{rfp}\}$ is described by the hybrid dynamical system

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}_\sigma(\boldsymbol{x}, \boldsymbol{u}), & \boldsymbol{x} \in \mathcal{C}_\sigma \\ \left(\boldsymbol{x}^+, \sigma^+\right) = \left(\boldsymbol{g}_\sigma(\boldsymbol{x}), \{\text{lfp}, \text{rfp}\} \setminus \sigma\right), & \boldsymbol{x} \in \mathcal{D}_\sigma, \end{cases} \tag{3.36}$$

where

$$\begin{aligned} \boldsymbol{f}_\sigma(\boldsymbol{x}, \boldsymbol{u}) &= \begin{bmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\left(\boldsymbol{S}\,\mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u}) - \boldsymbol{H}(\dot{\boldsymbol{q}})\right) - \boldsymbol{A}_{\sigma,12}(\boldsymbol{q})\boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix}, \\ \boldsymbol{g}_\sigma(\boldsymbol{x}) &= \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{M}\dot{\boldsymbol{q}} \end{bmatrix}, \end{aligned}$$

$$\begin{aligned} \mathcal{C}_\sigma &= \left\{\boldsymbol{x} \in \mathbb{R}^{2n} : y_{\text{swf},\sigma}(\boldsymbol{q}) \geq 0\right\}, \\ \mathcal{D}_\sigma &= \left\{\boldsymbol{x} \in \mathbb{R}^{2n} : y_{\text{swf},\sigma}(\boldsymbol{q}) = 0, \dot{y}_{\text{swf},\sigma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) < 0\right\}. \end{aligned}$$

Recall from earlier in this chapter that the ground is assumed to be flat and level. Consequently, the altitude of the ground can be set to zero relative to the inertial frame, so that it vanishes in the set rules of $\mathcal{C}_\sigma$ and $\mathcal{D}_\sigma$. Lastly, note that if the contact constraint (3.27) is violated during the continuous-time evolution of the state $\boldsymbol{x}$, then the solution $\boldsymbol{x}(t)$ is considered invalid and must be rejected.

## 3.5 Auxiliary quantities

In this section, several quantities are introduced that will be useful for the control of the bipedal robot, as discussed in the next chapter. First, recall that the vector of generalized coordinates is given by

$$\boldsymbol{q} = \begin{bmatrix} x_{\mathrm{tr}} & y_{\mathrm{tr}} & x_{\mathrm{lth}} & y_{\mathrm{lth}} & \theta_{\mathrm{lth}} & x_{\mathrm{lsh}} & y_{\mathrm{lsh}} & \theta_{\mathrm{lsh}} & x_{\mathrm{rth}} & y_{\mathrm{rth}} & \theta_{\mathrm{rth}} & x_{\mathrm{rsh}} & y_{\mathrm{rsh}} & \theta_{\mathrm{rsh}} \end{bmatrix}^{\top}.$$

Also recall that $l_{\mathrm{tr}}$ is the length of the torso, $l_{\mathrm{th}}$ is the length of a thigh, and $l_{\mathrm{sh}}$ is the length of a shank. Now, let us define the stance leg angle

$$\theta_{\mathrm{stl}}(\boldsymbol{q}) = -\arctan\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf},\sigma}(\boldsymbol{q})}{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}} - y_{\mathrm{stf},\sigma}(\boldsymbol{q})}\right), \tag{3.37}$$

where $\left(x_{\mathrm{tr}}, y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\right)$ is the position of the hips, and $(x_{\mathrm{stf},\sigma}(\boldsymbol{q}), y_{\mathrm{stf},\sigma}(\boldsymbol{q}))$ is the position of the stance foot. The stance leg angle is the angle between a vertical line and the line that goes through both the stance foot and the hips. If the stance leg angle $\theta_{\mathrm{stl}}(\boldsymbol{q}) = 0$, then the hips are exactly above the stance foot. Note that $y_{\mathrm{stf},\sigma}(\boldsymbol{q}) = 0$, because the ground is flat and level and its altitude is set to zero.

In addition, let us define the maximum step size

$$\Delta_{\max}(y_{\mathrm{tr}}) = 2\sqrt{(l_{\mathrm{th}} + l_{\mathrm{sh}})^2 - \left(y_{\mathrm{tr}} - \tfrac{1}{2}l_{\mathrm{tr}}\right)^2}, \tag{3.38}$$

where $(l_{\mathrm{th}} + l_{\mathrm{sh}})$ is the length of a fully extended leg, and $\left(y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\right)$ is the elevation of the hips. The maximum step size is the maximum distance between both feet for a given elevation of the hips, whereby it is assumed that both feet are in contact with the ground. If the elevation of the hips satisfies

$$0 < y_{\mathrm{tr}} - \tfrac{1}{2}l_{\mathrm{tr}} < l_{\mathrm{th}} + l_{\mathrm{sh}}, \tag{3.39}$$

then the maximum step size satisfies $0 < \Delta_{\max}(y_{\mathrm{tr}}) < 2\,(l_{\mathrm{th}} + l_{\mathrm{sh}})$.

Lastly, consider that the bipedal robot is standing on one foot and that its posture is fixed. As a consequence, the robot will pivot around the stance foot as if the entire robot is an inverted pendulum. Also, assume that the center of mass (COM) of the robot is located at the hips. The COM accelerates towards the ground with a gravitational acceleration $g_{\mathrm{a}}$, as depicted in Figure 3.4. However, the motion of the COM is constrained to a circular path here, provided that the stance foot stays in contact with the ground without slipping. Along this circular path, the gravitational acceleration vector $-g_{\mathrm{a}}\boldsymbol{e}_2$ can be decomposed into a tangential component and a normal component. The tangential component is the component that is of interest here, because it represents the acceleration of the COM along the circular path that makes the inverted pendulum tip over. The magnitude of the tangential component is equal to $g_{\mathrm{a}}\sin(\theta_{\mathrm{stl}}(\boldsymbol{q}))$. The tangential component itself can be described in the inertial frame as the vector $\bar{\boldsymbol{a}} = \bar{a}_1\boldsymbol{e}_1 + \bar{a}_2\boldsymbol{e}_2$, with

$$\begin{aligned} \bar{a}_1 &= -g_{\mathrm{a}}\sin(\theta_{\mathrm{stl}}(\boldsymbol{q}))\cos(\theta_{\mathrm{stl}}(\boldsymbol{q})), \\ \bar{a}_2 &= -g_{\mathrm{a}}\sin(\theta_{\mathrm{stl}}(\boldsymbol{q}))\sin(\theta_{\mathrm{stl}}(\boldsymbol{q})). \end{aligned}$$

Hereby, observe that given the identities $\sin(\arctan(\nu)) = \nu/\sqrt{1+\nu^2}$ and $\cos(\arctan(\nu)) = 1/\sqrt{1+\nu^2}$, it follows that

$$\begin{aligned} \sin(\theta_{\mathrm{stl}}(\boldsymbol{q})) &= -\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{\sqrt{(x_{\mathrm{tr}} - x_{\mathrm{stf}})^2 + \left(y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\right)^2}}, \\ \cos(\theta_{\mathrm{stl}}(\boldsymbol{q})) &= \frac{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}{\sqrt{(x_{\mathrm{tr}} - x_{\mathrm{stf}})^2 + \left(y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\right)^2}}, \end{aligned}$$

where it is assumed that the position of the stance foot $(x_{\mathrm{stf},\sigma}(\boldsymbol{q}), y_{\mathrm{stf},\sigma}(\boldsymbol{q})) = (x_{\mathrm{stf}}, 0)$. Furthermore, let us define the functions

$$\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}) = g_{\mathrm{a}}\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)\left(\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)^2 + 1\right)^{-1}, \tag{3.40}$$

$$\bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}) = -g_{\mathrm{a}}\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)^2\left(\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)^2 + 1\right)^{-1}. \tag{3.41}$$

These two functions equate to the foregoing trigonometric expressions of $\bar{a}_1$ and $\bar{a}_2$.

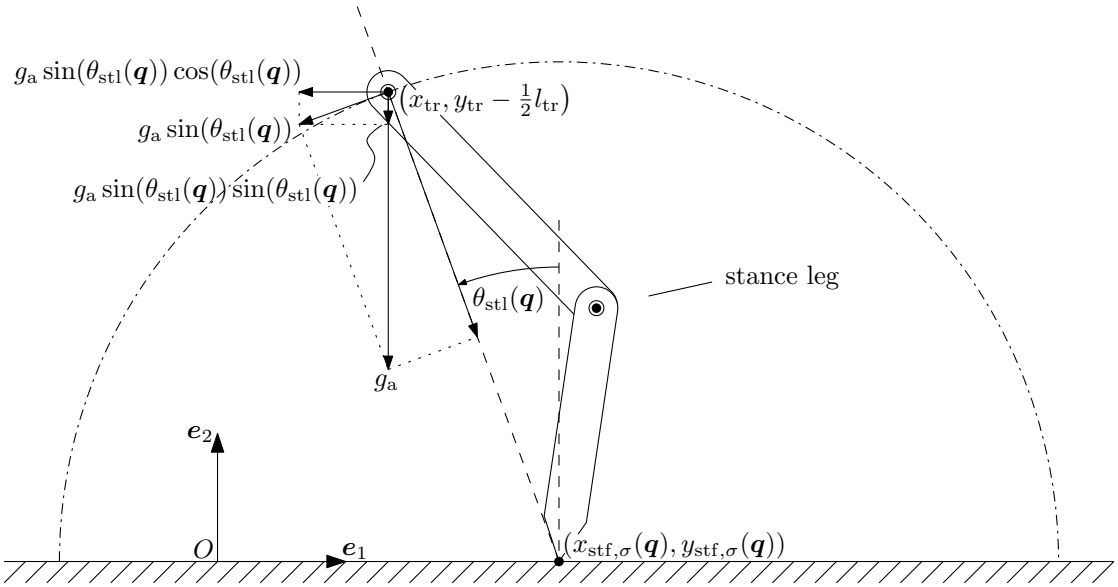



Figure 3.4: The gravitational acceleration vector $-g_\mathrm{a}\boldsymbol{e}_2$ is attached to the center of mass of the bipedal robot, which is assumed to be located at the hips here. This vector can be decomposed into a tangential and a normal component if the posture of the robot is fixed so that it behaves like an inverted pendulum, whereby the center of mass rotates around the stance foot in a circle. Only the stance leg of the robot is shown in this illustration.

## 3.6 Summary

In this chapter, a 5-link planar bipedal robot is introduced that consists of a torso and two legs, whereby each leg consists of a thigh and a shank. Furthermore, the robot has point feet that are modeled as frictionless revolute joints during the time that they are in contact with the ground. Except for the point feet, all joints of the robot are actuated, where it is assumed that the applied torques saturate at a given torque limit. The motion of the robot is constrained to the sagittal plane, which can be realized using a lateral support bar. This bar can also be used to constrain the rotational degree of freedom of the torso in the sagittal plane, so that the robot is fully actuated in a single support mode.

The multibody dynamics of the bipedal robot is formulated using a redundant set of absolute coordinates. Hereby, the equations of motion are derived using the Newton-Euler equations. In addition, Coulomb's model of static friction is used to check the validity of the bilateral constraint that exists in the form of a revolute joint between the stance foot and the ground. Collisions between the swing foot and the ground are considered perfectly inelastic. The impacts from these collisions are modeled with Newton's impact law, where the restitution coefficient is equal to zero. By using this impact law in combination with the law of conservation of momentum, a state-reset law is derived for the generalized velocities of the robot. Finally, the hybrid systems framework is used to combine the multibody dynamics with the state-reset law into one complete model of the robot.

In preparation for the next chapter, the stance leg angle and the maximum step size are defined in terms of the generalized coordinates of the robot. The stance leg angle is used in the derivation of the tangential component of the gravitational acceleration vector. In this derivation, it is assumed that the posture of the robot is fixed, so that it behaves like an inverted pendulum. The maximum step size and the tangential component of the gravitational acceleration vector will be used for the control design in the next chapter.

# Chapter 4

# State feedback control for dynamic walking

In this chapter, a state feedback law is designed to obtain dynamic walking to a desired location with the planar bipedal robot that is introduced in the previous chapter. The desired walking gait is described in part by the desired position of the swing foot, which will need to be tracked by means of feedback control. For this reason, part of the output equation of the robotic system is defined so that it represents the error in the position of the swing foot. The other part of the system's output represents the error in the position of the torso, which allows the robot to be controlled to the desired location. The multivariable system is then input-output linearized. Subsequently, bounded controllers are used for the stabilization of the origin of the output error dynamics. Lastly, the bounds of these controllers are maximized subject to the torque limits and the contact constraint, as given by Coulomb's model of static friction.

## 4.1 The desired position of the swing foot

Let the pair of variables $\left(x^{\mathrm{d}}_{\mathrm{swf}}, y^{\mathrm{d}}_{\mathrm{swf}}\right)$ denote the desired position of the swing foot. This desired position will be specified as a function of the state of the robot $\boldsymbol{x}$ in order to obtain a dynamic walking gait. An important aspect of bipedal walking is that the swing foot and the stance foot of the robot are always on opposite sides of the torso/hips. Therefore, consider that the swing foot and the stance foot are at an equal horizontal distance from the torso, so that $x_{\mathrm{swf}} - x_{\mathrm{tr}} = x_{\mathrm{tr}} - x_{\mathrm{stf}}$, or equivalently

$$x_{\mathrm{swf}} = x_{\mathrm{tr}} + (x_{\mathrm{tr}} - x_{\mathrm{stf}}), \tag{4.1}$$

where $x_{\mathrm{swf}}$ is the location of the swing foot, $x_{\mathrm{tr}}$ is the location of the torso, and $x_{\mathrm{stf}}$ is the location of the stance foot. The relation (4.1) will be called the equidistant rule of bipedal walking. Also, consider that the swing foot is placed on the ground as soon as the horizontal distance between the torso and the stance foot $|x_{\mathrm{tr}} - x_{\mathrm{stf}}|$ exceeds a given threshold value.

Let us use the Capture Point [39] as the desired foot placement location. Recall from the literature study that the Capture Point is the point on the ground where a bipedal robot needs to step in order to come to a standstill above the stance foot. The Capture Point is defined so that, after the foot placement, the robot passively comes to a standstill as if the robot is an inverted pendulum. It is useful to step on the Capture Point in order to reject disturbances and to keep the robot balanced with the least amount of effort, as discussed in the literature study.

The Capture Point is derived using the equation of motion of the linear inverted pendulum model. The derivation is given here based on the current bipedal robot. Hereto, consider the system

$$\begin{aligned} m_{\mathrm{tr}}\ddot{x}_{\mathrm{tr}} &= m_{\mathrm{tr}}\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y^{\mathrm{d}}_{\mathrm{tr}}) \\ &= m_{\mathrm{tr}} g_{\mathrm{a}} \left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y^{\mathrm{d}}_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)\left(\left(\frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y^{\mathrm{d}}_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}}\right)^2 + 1\right)^{-1}, \end{aligned} \tag{4.2}$$

where the elevation of the torso $y_{\mathrm{tr}} = y_{\mathrm{tr}}^{\mathrm{d}} > \frac{1}{2} l_{\mathrm{tr}}$, and $l_{\mathrm{tr}}$ is the length of the torso. The constant $y_{\mathrm{tr}}^{\mathrm{d}}$ represents the desired elevation of the torso. Thus, the constant $y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2} l_{\mathrm{tr}} > 0$ represents the desired elevation of the hips. In addition, the location of the stance foot $x_{\mathrm{stf}}$ is piecewise constant during walking. Now, let us linearize the right-hand side of (4.2) at $x_{\mathrm{tr}} = x_{\mathrm{stf}}$. The result is the linear inverted pendulum equation

$$\ddot{x}_{\mathrm{tr}} = c\,(x_{\mathrm{tr}} - x_{\mathrm{stf}}), \tag{4.3}$$

with

$$c = \frac{g_{\mathrm{a}}}{y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2} l_{\mathrm{tr}}} > 0.$$

The implicit solution to this second-order linear differential equation satisfies

$$\dot{x}_{\mathrm{tr}}^2 - c\,(x_{\mathrm{tr}} - x_{\mathrm{stf}})^2 = \chi, \tag{4.4}$$

whereby $(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$ represents the state of the system, and $\chi$ is a constant. For $\chi = 0$, the equation (4.4) simplifies to $\dot{x}_{\mathrm{tr}} = \pm\sqrt{c}\,(x_{\mathrm{tr}} - x_{\mathrm{stf}})$, which corresponds to two lines that intersect at the point $(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = (x_{\mathrm{stf}}, 0)$ in the two-dimensional state space of the system. The point of intersection of these two lines is also the equilibrium point of (4.3). Notably, it is the point where the robot is at rest above the stance foot. Moreover, the equilibrium point is a saddle point, and the two lines of $\{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : \dot{x}_{\mathrm{tr}} = \pm\sqrt{c}\,(x_{\mathrm{tr}} - x_{\mathrm{stf}})\}$ are the stable and unstable manifolds of the saddle point. The Capture Point is defined as

$$x_{\mathrm{CP}}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = x_{\mathrm{tr}} + c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}, \tag{4.5}$$

which is the location of the stance foot $x_{\mathrm{stf}}$ on the stable manifold $\{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : \dot{x}_{\mathrm{tr}} = -\sqrt{c}\,(x_{\mathrm{tr}} - x_{\mathrm{stf}})\}$.

A potential problem with the use of the Capture Point as the desired foot placement location is that it goes out of reach for large enough $|\dot{x}_{\mathrm{tr}}|$. Recall from Section 3.5 that the maximum step size is given by

$$\Delta_{\max}(y_{\mathrm{tr}}) = 2\sqrt{(l_{\mathrm{th}} + l_{\mathrm{sh}})^2 - \left(y_{\mathrm{tr}} - \tfrac{1}{2} l_{\mathrm{tr}}\right)^2},$$

which is based on the configuration of the robot where both legs are fully extended and both feet are in contact with the ground. Let us denote the desired maximum step size as $\Delta_{\max}(y_{\mathrm{tr}}^{\mathrm{d}}) = \Delta_{\max}^{\mathrm{d}}$. Given that the elevation of the torso $y_{\mathrm{tr}} = y_{\mathrm{tr}}^{\mathrm{d}}$, it follows that the horizontal distance between the location of the torso $x_{\mathrm{tr}}$ and the location of either foot is at most $\frac{1}{2}\Delta_{\max}^{\mathrm{d}}$. Therefore, let us introduce the "confined" Capture Point

$$\bar{x}_{\mathrm{CP}}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = x_{\mathrm{tr}} + \mathrm{sat}_{\gamma}(c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}), \tag{4.6}$$

where the maximum relative distance

$$\gamma = \frac{1}{2} \beta \Delta_{\max}^{\mathrm{d}}, \tag{4.7}$$

with $0 < \beta < 1$. It follows that the horizontal distance between the location of the torso $x_{\mathrm{tr}}$ and the confined Capture Point $\bar{x}_{\mathrm{CP}}$ satisfies $|x_{\mathrm{tr}} - \bar{x}_{\mathrm{CP}}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})| \leq \gamma < \frac{1}{2}\Delta_{\max}^{\mathrm{d}}$ for all $(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$, which means that the desired foot placement location will no longer go out of reach of the robot.

Let the desired location of the swing foot $x_{\mathrm{swf}}^{\mathrm{d}}$ be a blend between the confined Capture Point (4.6) and a bounded version of the equidistant rule (4.1). Hereby, the sign of the equidistant rule will be preserved, because it is important that the swing foot and the stance foot are always on opposite sides of the torso. In particular, consider that the desired location of the swing foot

$$x_{\mathrm{swf}}^{\mathrm{d}} = x_{\mathrm{tr}} + \eta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}), \tag{4.8}$$

with

$$\eta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = \gamma^{-1}\,\mathrm{sat}_{\gamma}(x_{\mathrm{tr}} - x_{\mathrm{stf}})\,\mathrm{sign}(\dot{x}_{\mathrm{tr}})\,\mathrm{sat}_{\gamma}(c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}). \tag{4.9}$$

The function (4.9) is sketched in Figure 4.1a. Notice that the desired location of the swing foot is equal to the confined Capture Point in the region

$$\{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : |x_{\mathrm{tr}} - x_{\mathrm{stf}}| \geq \gamma\} \cap \{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : (x_{\mathrm{tr}} - x_{\mathrm{stf}})\,\dot{x}_{\mathrm{tr}} \geq 0\}. \tag{4.10}$$

It is desired that the foot placement occurs at the boundary of the region (4.10), so that the swing foot is placed on the confined Capture Point as soon as possible during walking, whereby it is assumed that the

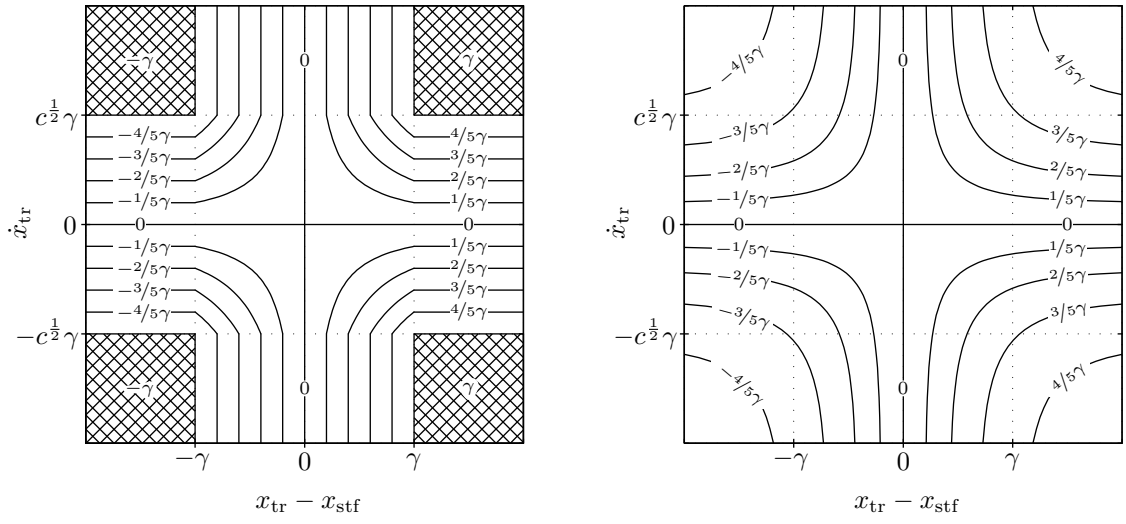



Figure 4.1: (a) A level curve sketch of (4.9) with the levels $\left\{-\gamma, -\frac{4}{5}\gamma, -\frac{3}{5}\gamma, \ldots, \frac{3}{5}\gamma, \frac{4}{5}\gamma, \gamma\right\}$. (b) A level curve sketch of (4.13) with the same levels as in (a), and with $a \gg 1$.

swing foot tracks (4.8). The aforementioned boundary can be given by the set $\{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : \zeta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = 0\}$, where

$$\begin{aligned}\zeta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = (\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}} + |\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}}|) \\ \cdot(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}} + |\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}}|).\end{aligned} \tag{4.11}$$

The function (4.11) is sketched in Figure 4.2a. Notice that in the region $\{(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) : (x_{\mathrm{tr}} - x_{\mathrm{stf}})\, \dot{x}_{\mathrm{tr}} > 0\}$ it is desired to place the swing foot on the ground when $|x_{\mathrm{tr}} - x_{\mathrm{stf}}| = \gamma$. Here, the constant $\gamma$ represents the same threshold value that is mentioned at the beginning of this section. Now, by using this observation in combination with the fact that $|\eta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})| \leq \gamma$, it follows that the desired distance between both feet satisfies $\left|x^{\mathrm{d}}_{\mathrm{swf}} - x_{\mathrm{stf}}\right| \leq 2\gamma < \Delta^{\mathrm{d}}_{\mathrm{max}}$.

To actually place the swing foot on the ground, consider that the desired elevation of the swing foot

$$y^{\mathrm{d}}_{\mathrm{swf}} = \bar{y}_{\mathrm{swf}} \tanh(\phi \zeta(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})), \tag{4.12}$$

where $\bar{y}_{\mathrm{swf}} > 0$ is the maximum foot clearance, and $\phi > 0$ constitutes the angle of approach for the foot placement. The hyperbolic tangent function $\tanh(\cdot)$ is a sigmoid function that is defined as $\tanh(\nu) = \frac{e^{\nu} - e^{-\nu}}{e^{\nu} + e^{-\nu}}$.

The desired position of the swing foot $\left(x^{\mathrm{d}}_{\mathrm{swf}}, y^{\mathrm{d}}_{\mathrm{swf}}\right)$ is used for feedback control. This requires that the partial derivatives of (4.9) and (4.11) are continuous, which will become clear in Section 4.2. Therefore, consider the function

$$\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = \gamma \tanh(\gamma^{-1}\,(x_{\mathrm{tr}} - x_{\mathrm{stf}})) \tanh(a\dot{x}_{\mathrm{tr}}) \tanh(\gamma^{-1} c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}) \tag{4.13}$$

as a substitute for (4.9), whereby the non-differentiable function $\mathrm{sat}_{\kappa}(\nu)$ is replaced by the differentiable approximation $\kappa \tanh(\kappa^{-1}\nu)$. Likewise, the function $\mathrm{sign}(\nu)$ is replaced by $\tanh(a\nu)$, with $a \gg 1$. In addition, consider the function

$$\begin{aligned}\tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = \left(\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}} + \sqrt{(\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}})^2 + \delta}\right) \\ \cdot\left(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}} + \sqrt{(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}})^2 + \delta}\right)\end{aligned} \tag{4.14}$$

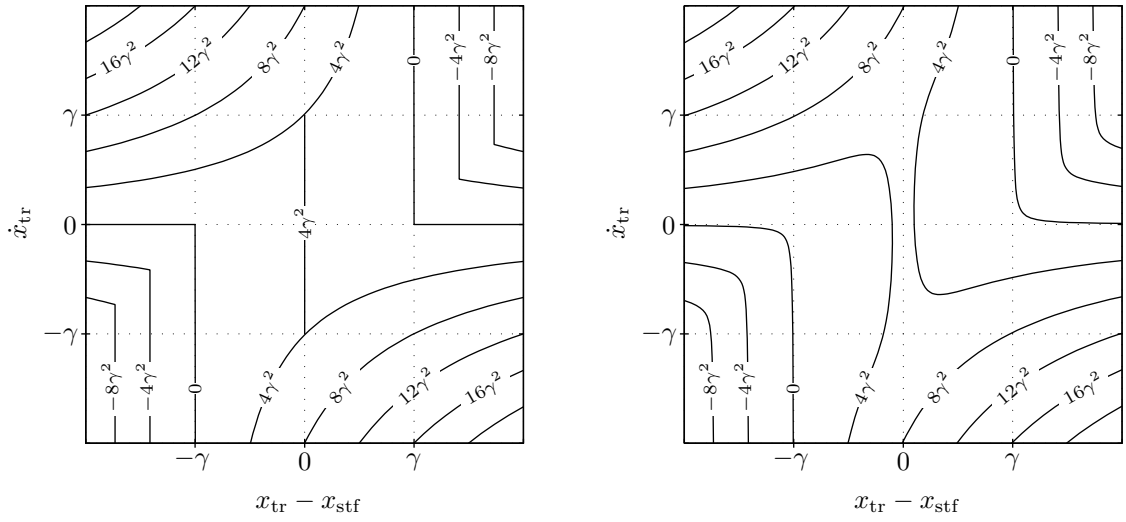



Figure 4.2: (a) A level curve sketch of (4.11) with the levels $\{-8\gamma^2, -4\gamma^2, 0, \ldots, 12\gamma^2, 16\gamma^2, 20\gamma^2\}$. (b) A level curve sketch of (4.14) with the same levels as in (a), and with $\frac{1}{\delta} \gg 1$.

as a substitute for (4.11), whereby the absolute value $|\nu|$ is replaced by $\sqrt{\nu^2 + \delta}$, with $\frac{1}{\delta} \gg 1$. The functions (4.13) and (4.14) are visualized in Figure 4.1b and Figure 4.2b, respectively. After substitution the desired position of the swing foot becomes

$$x_{\mathrm{swf}}^{\mathrm{d}} = x_{\mathrm{tr}} + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}), \tag{4.15}$$

$$y_{\mathrm{swf}}^{\mathrm{d}} = \bar{y}_{\mathrm{swf}} \tanh(\phi \tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})). \tag{4.16}$$

Note that due to the use of (4.13) in (4.15), the foot placement location will differ slightly from the confined Capture Point, especially for relatively large $|\dot{x}_{\mathrm{tr}}|$. In particular, the swing foot will be placed closer to the torso on the ground for relatively large $|\dot{x}_{\mathrm{tr}}|$. As a result, the robot will take slightly smaller steps than what is prescribed by the confined Capture Point.

## 4.2 Input-output linearization of the continuous-time dynamics

First, consider the linear system

$$\dot{\rho} = -\omega_0 (\rho - x_{\mathrm{tr}}), \tag{4.17a}$$

$$\psi = -\omega_0 (\rho - x_{\mathrm{tr}}), \tag{4.17b}$$

where $\rho$ is the state variable, $\psi$ is the output variable, and $\omega_0$ is a positive constant. The location of the torso $x_{\mathrm{tr}}$ is used as the input variable. The system (4.17) is adopted from [5], where it is part of a dynamic output feedback control law that is proposed for the global regulation of robots using only position measurements. The input-output behavior of (4.17) is described by the transfer function

$$G(s) = \frac{\omega_0 s}{s + \omega_0}, \tag{4.18}$$

where $s$ is the complex variable that is associated with the Laplace transform. Observe that if $s \to 0$, then $G(s) \to s$. In classical control theory, (4.18) is called a tame differentiator, because it is a differentiator with the addition of a pole at $s = -\omega_0$. Evidently, the behavior of (4.17) is such that the output $\psi$ represents a filtered version of the velocity of the torso $\dot{x}_{\mathrm{tr}}$. This behavior will turn out to be useful for input-output decoupling, where $\psi$ will be used as a substitute for $\dot{x}_{\mathrm{tr}}$.

Notice that the output of the tame differentiator (4.17b) is a continuous function of time, because $\rho$ and $x_{\mathrm{tr}}$ are also continuous functions of time. On the other hand, the velocity of the torso $\dot{x}_{\mathrm{tr}}$ is not a

continuous function of time; it must be reset during walking at every collision between the swing foot and the ground. To compensate for these discontinuities in $\dot{x}_{\mathrm{tr}}$, consider that the state variable of the tame differentiator $\rho$ is reset at the same time as the velocity of the torso $\dot{x}_{\mathrm{tr}}$, so that

$$\psi^- - \dot{x}_{\mathrm{tr}}^- = \psi^+ - \dot{x}_{\mathrm{tr}}^+, \tag{4.19}$$

where the superscript minus sign denotes the moment just before impact, and the superscript plus sign denotes the moment just after impact. The reset map (4.19) ensures that the output of the tame differentiator $\psi$ copies the instantaneous jumps of the velocity of the torso $\dot{x}_{\mathrm{tr}}$. As a result, the difference between $\psi$ and $\dot{x}_{\mathrm{tr}}$ will be continuous. Furthermore, observe that (4.19) can be written as

$$\omega_0\rho^- + \dot{x}_{\mathrm{tr}}^- = \omega_0\rho^+ + \dot{x}_{\mathrm{tr}}^+,$$

where the relation (4.17b) is used together with the fact that $x_{\mathrm{tr}}^- = x_{\mathrm{tr}}^+$. The instantaneous jumps of $\dot{x}_{\mathrm{tr}}$ are described by the reset law for the vector of generalized velocities (3.35). Hereby, let $\boldsymbol{p}_1$ be a constant vector so that $\boldsymbol{p}_1^\top \dot{\boldsymbol{q}} = \dot{x}_{\mathrm{tr}}$. Now, if the superscript minus signs are omitted, then the reset law for $\rho$ that satisfies (4.19) can be expressed as

$$\rho^+ = \rho + \omega_0^{-1}\boldsymbol{p}_1^\top \left(\boldsymbol{I} - \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{M}\right)\dot{\boldsymbol{q}}, \tag{4.20}$$

where $\dot{x}_{\mathrm{tr}}^+ = \boldsymbol{p}_1^\top \dot{\boldsymbol{q}}^+ = \boldsymbol{p}_1^\top \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{M}\dot{\boldsymbol{q}}$ according to (3.35), and $\boldsymbol{I}$ is the identity matrix of appropriate dimensions.

Recall that for $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$, the state equation of the hybrid dynamical system (3.36) is given by

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\left(\boldsymbol{S}\,\mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u}) - \boldsymbol{H}(\dot{\boldsymbol{q}})\right) - \boldsymbol{A}_{\sigma,12}(\boldsymbol{q})\boldsymbol{w}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}}) \end{bmatrix}, \tag{4.21}$$

where $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$ with

$$\boldsymbol{q} = \begin{bmatrix} x_{\mathrm{tr}} & y_{\mathrm{tr}} & x_{\mathrm{lth}} & y_{\mathrm{lth}} & \theta_{\mathrm{lth}} & x_{\mathrm{lsh}} & y_{\mathrm{lsh}} & \theta_{\mathrm{lsh}} & x_{\mathrm{rth}} & y_{\mathrm{rth}} & \theta_{\mathrm{rth}} & x_{\mathrm{rsh}} & y_{\mathrm{rsh}} & \theta_{\mathrm{rsh}} \end{bmatrix}^\top,$$

and

$$\begin{aligned} \boldsymbol{A}_{\sigma,11}(\boldsymbol{q}) &= \boldsymbol{M}^{-1} - \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma(\boldsymbol{q})\left(\boldsymbol{W}_\sigma^\top(\boldsymbol{q})\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma(\boldsymbol{q})\right)^{-1}\boldsymbol{W}_\sigma^\top(\boldsymbol{q})\boldsymbol{M}^{-1}, \\ \boldsymbol{A}_{\sigma,12}(\boldsymbol{q}) &= \boldsymbol{M}^{-1}\boldsymbol{W}_\sigma(\boldsymbol{q})\left(\boldsymbol{W}_\sigma^\top(\boldsymbol{q})\boldsymbol{M}^{-1}\boldsymbol{W}_\sigma(\boldsymbol{q})\right)^{-1}. \end{aligned}$$

Furthermore, the vector $\boldsymbol{u}$ represents the input of the system (4.21). Now, consider that this system has the output vector

$$\boldsymbol{y}_\sigma = \begin{bmatrix} x_{\mathrm{tr}}, & y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}, & x_{\mathrm{swf},\sigma}(\boldsymbol{q}) - x_{\mathrm{swf}}^{\mathrm{d}}, & y_{\mathrm{swf},\sigma}(\boldsymbol{q}) - y_{\mathrm{swf}}^{\mathrm{d}} \end{bmatrix}^\top, \tag{4.22}$$

where

$$\begin{aligned} x_{\mathrm{swf}}^{\mathrm{d}} &= x_{\mathrm{tr}} + \tilde{\eta}(x_{\mathrm{tr}}, \psi), \\ y_{\mathrm{swf}}^{\mathrm{d}} &= \bar{y}_{\mathrm{swf}} \tanh(\phi\tilde{\zeta}(x_{\mathrm{tr}}, \psi)), \end{aligned}$$

in which $\psi$ is replacing $\dot{x}_{\mathrm{tr}}$ as compared to (4.15)-(4.16). Also recall from Section 3.2 that the position of the swing foot is given by

$$\begin{aligned} x_{\mathrm{swf},\sigma}(\boldsymbol{q}) &= \begin{cases} x_{\mathrm{rsh}} + \frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ x_{\mathrm{lsh}} + \frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases} \\ y_{\mathrm{swf},\sigma}(\boldsymbol{q}) &= \begin{cases} y_{\mathrm{rsh}} - \frac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ y_{\mathrm{lsh}} - \frac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}. \end{cases} \end{aligned}$$

Notice that the output vector (4.22) is only a function of the vector of generalized coordinates $\boldsymbol{q}$ and the state of the tame differentiator $\rho$, because the output of the tame differentiator $\psi = -\omega_0(\rho - x_{\mathrm{tr}})$, whereby $x_{\mathrm{tr}} = \boldsymbol{p}_1^\top \boldsymbol{q}$. To clarify the dependence on $\rho$, it is useful to define the functions

$$\breve{\eta}(x_{\mathrm{tr}}, \rho) = \tilde{\eta}(x_{\mathrm{tr}}, -\omega_0(\rho - x_{\mathrm{tr}})), \tag{4.23}$$

$$\breve{\zeta}(x_{\mathrm{tr}}, \rho) = \tilde{\zeta}(x_{\mathrm{tr}}, -\omega_0(\rho - x_{\mathrm{tr}})), \tag{4.24}$$

and then to define the functions

$$x^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}},\rho) = x_{\mathrm{tr}} + \breve{\eta}(x_{\mathrm{tr}},\rho), \tag{4.25}$$

$$y^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}},\rho) = \bar{y}_{\mathrm{swf}} \tanh(\phi\breve{\zeta}(x_{\mathrm{tr}},\rho)), \tag{4.26}$$

so that the desired position of the swing foot finally becomes $\big(x^{\mathrm{d}}_{\mathrm{swf}}, y^{\mathrm{d}}_{\mathrm{swf}}\big) = \big(x^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}},\rho), y^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}},\rho)\big)$.

The multivariable system (4.21)-(4.22) is square, which means that $\dim(\boldsymbol{u}) = \dim(\boldsymbol{y}_\sigma)$. Note that the input vector $\boldsymbol{u}$ will contain the control actions for the four actuators of the robot, whereby the vector of applied torques $\boldsymbol{\tau} = (\tau_{\mathrm{lh}}, \tau_{\mathrm{lk}}, \tau_{\mathrm{rh}}, \tau_{\mathrm{rk}})$ saturates according to the relation $\boldsymbol{\tau} = \mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u})$, where $\boldsymbol{L} = (L_{\mathrm{h}}, L_{\mathrm{k}}, L_{\mathrm{h}}, L_{\mathrm{k}})$. Also notice that the first two elements of the output vector $(y_{\sigma,1}, y_{\sigma,2})$ represent the position of the hips, while the last two elements $(y_{\sigma,3}, y_{\sigma,4})$ represent the error in the position of the swing foot.

The first and second derivative of the output vector $\boldsymbol{y}_\sigma$ with respect to time are denoted by $\dot{\boldsymbol{y}}_\sigma$ and $\ddot{\boldsymbol{y}}_\sigma$, respectively. The explicit expressions for $\dot{\boldsymbol{y}}_\sigma$ and $\ddot{\boldsymbol{y}}_\sigma$ are given in Appendix B. These explicit expressions reveal that the second derivative $\ddot{\boldsymbol{y}}_\sigma$ is linear in $\ddot{\boldsymbol{q}}$, so that for appropriately defined functions $\boldsymbol{C}_\sigma(\boldsymbol{q},\rho)$ and $\boldsymbol{b}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\rho)$, the second derivative can be written as

$$\begin{aligned}\ddot{\boldsymbol{y}}_\sigma &= \boldsymbol{C}_\sigma(\boldsymbol{q},\rho)\ddot{\boldsymbol{q}} + \boldsymbol{b}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\rho) \\ &= \boldsymbol{C}_\sigma(\boldsymbol{q},\rho)\boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{S}\,\mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u}) - \boldsymbol{C}_\sigma(\boldsymbol{q},\rho)\left(\boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{H}(\dot{\boldsymbol{q}}) + \boldsymbol{A}_{\sigma,12}(\boldsymbol{q})\boldsymbol{w}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}})\right) + \boldsymbol{b}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\rho). \end{aligned} \tag{4.27}$$

Given the first derivative of the output vector $\dot{\boldsymbol{y}}_\sigma = \dot{\boldsymbol{y}}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\rho)$ and the second derivative of the output vector (4.27), it follows that each individual output $y_{\sigma,i}$ has relative degree $r_i \geq 2$, for $i = 1,\ldots,4$. Next, let us define the square matrix

$$\boldsymbol{D}_\sigma(\boldsymbol{q},\rho) = \boldsymbol{C}_\sigma(\boldsymbol{q},\rho)\boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{S}, \tag{4.28}$$

which is called the decoupling matrix. It can be shown with the help of a computer algebra system that

$$\det(\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)) = \frac{l^2_{\mathrm{th}} l^2_{\mathrm{sh}} \sin(\theta_{\mathrm{lsh}} - \theta_{\mathrm{lth}}) \sin(\theta_{\mathrm{rsh}} - \theta_{\mathrm{rth}})}{d_\sigma(\boldsymbol{q},\rho)}, \tag{4.29}$$

where $d_\sigma(\boldsymbol{q},\rho)$ represents the denominator of the determinant of $\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)$. If $\theta_{\mathrm{lsh}} - \theta_{\mathrm{lth}} \neq k\pi$ and $\theta_{\mathrm{rsh}} - \theta_{\mathrm{rth}} \neq k\pi$, where $k$ is an integer, then $\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)$ is non-singular. This holds for both modes $\sigma \in \{\mathrm{lfp}, \mathrm{rfp}\}$. Evidently, each row of $\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)$ is non-zero, at least for all configurations $\boldsymbol{q}$ for which $\det(\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)) \neq 0$. As a result, each individual output $y_{\sigma,i}$ has relative degree $r_i = 2$. Configurations for which $\det(\boldsymbol{D}_\sigma(\boldsymbol{q},\rho)) = 0$ are called singular configurations. The singular configurations that are relevant for walking are the configurations where at least one leg is fully extended. Note that if the velocity of the torso $\dot{x}_{\mathrm{tr}}$ would have been used instead of the output of the tame differentiator $\psi$, then the last two outputs $y_{\sigma,3}, y_{\sigma,4}$ would have had relative degree $r_3, r_4 = 1$, and the decoupling matrix would have been singular for all states.

Finally, consider the dynamic state feedback control law

$$\boldsymbol{u} = \boldsymbol{D}^{-1}_\sigma(\boldsymbol{q},\rho)\left(\boldsymbol{C}_\sigma(\boldsymbol{q},\rho)\left(\boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{H}(\dot{\boldsymbol{q}}) + \boldsymbol{A}_{\sigma,12}(\boldsymbol{q})\boldsymbol{w}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}})\right) - \boldsymbol{b}_\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\rho) + \boldsymbol{v}\right), \tag{4.30}$$

with

$$\begin{cases} \dot{\rho} = -\omega_0\,(\rho - x_{\mathrm{tr}}), & \boldsymbol{x} \in \mathcal{C}_\sigma \\ \rho^+ = \rho + \omega_0^{-1}\boldsymbol{p}_1^\top\,(\boldsymbol{I} - \boldsymbol{A}_{\sigma,11}(\boldsymbol{q})\boldsymbol{M})\,\dot{\boldsymbol{q}}, & \boldsymbol{x} \in \mathcal{D}_\sigma, \end{cases} \tag{4.31}$$

where $\boldsymbol{v}$ is the new control input, and $\mathcal{C}_\sigma, \mathcal{D}_\sigma$ are the sets that are part of the hybrid system (3.36). Since the control law is dynamic, the state of the feedback linearized system is now given by $(\boldsymbol{x},\rho) = (\boldsymbol{q},\dot{\boldsymbol{q}},\rho)$. Notice that if the control law (4.30) satisfies $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$, then $\mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u}) = \boldsymbol{u}$ and the output equation (4.27) reduces to

$$\ddot{\boldsymbol{y}}_\sigma = \boldsymbol{v}. \tag{4.32}$$

## 4.3 A control design for walking to a desired location

Recall from the previous section that the first two outputs $(y_{\sigma,1}, y_{\sigma,2}) = \big(x_{\mathrm{tr}}, y_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\big)$ represent the current position of the hips. Naturally, the desired position of the hips then becomes $\big(x^{\mathrm{d}}_{\mathrm{tr}}, y^{\mathrm{d}}_{\mathrm{tr}} - \frac{1}{2}l_{\mathrm{tr}}\big)$,

where $x_{\mathrm{tr}}^{\mathrm{d}}$ is the desired location of the center of mass of the torso, $y_{\mathrm{tr}}^{\mathrm{d}}$ is the desired elevation of the center of mass of the torso, and $l_{\mathrm{tr}}$ is the length of the torso. Let us use the desired position of the hips to define the desired output vector

$$\boldsymbol{y}_{\sigma}^{\mathrm{d}} = \begin{bmatrix} x_{\mathrm{tr}}^{\mathrm{d}} & y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2} l_{\mathrm{tr}} & 0 & 0 \end{bmatrix}^{\top}, \tag{4.33}$$

which will serve as the constant setpoint for the feedback controller. The setpoint (4.33) specifies the desired location for the robot to walk towards, as well as the desired walking height with which the robot must walk to this desired location. Obviously, one is interested in the output error $\boldsymbol{e}_{\sigma} = \boldsymbol{y}_{\sigma} - \boldsymbol{y}_{\sigma}^{\mathrm{d}}$, whereby

$$e_{\sigma,1} = x_{\mathrm{tr}} - x_{\mathrm{tr}}^{\mathrm{d}}, \tag{4.34a}$$

$$e_{\sigma,2} = y_{\mathrm{tr}} - y_{\mathrm{tr}}^{\mathrm{d}}, \tag{4.34b}$$

$$e_{\sigma,3} = x_{\mathrm{swf},\sigma}(\boldsymbol{q}) - x_{\mathrm{swf}}^{\mathrm{d}}(x_{\mathrm{tr}}, \rho), \tag{4.34c}$$

$$e_{\sigma,4} = y_{\mathrm{swf},\sigma}(\boldsymbol{q}) - y_{\mathrm{swf}}^{\mathrm{d}}(x_{\mathrm{tr}}, \rho). \tag{4.34d}$$

Conditionally, it holds that

$$\ddot{\boldsymbol{e}}_{\sigma} = \boldsymbol{v}, \tag{4.35}$$

because $\ddot{\boldsymbol{e}}_{\sigma} = \ddot{\boldsymbol{y}}_{\sigma}$, and if the control law (4.30) satisfies the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$, then $\ddot{\boldsymbol{y}}_{\sigma} = \boldsymbol{v}$, as shown in the previous section. Now, the goal is to design a control law for $\boldsymbol{v}$ that satisfies the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$ and stabilizes the origin of the output error dynamics

$$(\boldsymbol{e}_{\sigma}, \dot{\boldsymbol{e}}_{\sigma}) = (\boldsymbol{0}, \boldsymbol{0}). \tag{4.36}$$

Hereby, note that there is no zero dynamics if the stance foot is stationary on the ground, because then a constant position of the hips and a constant position of the swing foot will determine the entire state of the robot up to an inversion of the knee joints. This is indeed the case at the origin of the output error dynamics. Furthermore, if the origin (4.36) is stabilized by a suitable control law for $\boldsymbol{v}$, then it will become an equilibrium point of the output error dynamics (4.35), where $\boldsymbol{v}$ is equal to this control law. The origin (4.36) can then be a globally asymptotically stable equilibrium point of the output error dynamics. However, due to the hybrid dynamics of the robotic system, the stabilization of (4.36) only proves that the corresponding equilibrium point(s) of the closed-loop system are locally asymptotically stable, with a conservative estimate of the region of attraction that is confined to be within zero walking steps of the equilibrium point(s). The true region of attraction cannot be determined by the feedback linearization and the subsequent stabilization of the origin of the output error dynamics, because these operations are performed only on the continuous-time dynamics of the hybrid dynamical system.

In addition, observe that there will be an infinite number of equilibrium points in the state space of the closed-loop system that will correspond to the stabilized origin of the output error dynamics. This is a result of the fact that no error is defined for the position of the stance foot $(x_{\mathrm{stf}}, y_{\mathrm{stf}})$. Namely, the output error (4.34) only refers to the position of the torso/hips and the position of the swing foot. Consequently, the piecewise constant position of the stance foot is not controlled directly during walking, and there is also no unique relation between the position of the stance foot and the state of the output error dynamics $(\boldsymbol{e}_{\sigma}, \dot{\boldsymbol{e}}_{\sigma})$. This includes the origin (4.36) of course, which is meant to be stabilized. Thus, the stance foot can be at any feasible position if the state of the output error dynamics is at the stabilized origin. Moreover, there will be an equilibrium point in the state space of the closed-loop system for every feasible position of the stance foot for which it holds that (4.36). All these equilibrium points form a connected set in the state space, which is referred to as the equilibrium set of the closed-loop system.

Consider the bounded Proportional-Derivative (PD) control law

$$v_i = -\operatorname{sat}_{\varepsilon_i}(k_{\mathrm{P}i} e_{\sigma,i} + k_{\mathrm{D}i} \dot{e}_{\sigma,i}), \tag{4.37}$$

where $v_i$ is the $i$-th component of $\boldsymbol{v}$, and $\varepsilon_i, k_{\mathrm{P}i}, k_{\mathrm{D}i} > 0$ are positive constants. These PD controllers are bounded, which makes it possible to meet the requirements of the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$ in certain cases, as is discussed in Section 4.4. Now, let us turn our attention to the first output error (4.34a), for which the output error rate $\dot{e}_{\sigma,1} = \dot{x}_{\mathrm{tr}}$. The horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}$ can be regarded

as the walking speed of the robot. It is important to be able to modulate this speed. Unfortunately, the bounded PD controller (4.37) does not take into account the rate at which the error $e_{\sigma,1}$ approaches zero. Therefore, let us use a different control law for $v_1$ that does take the convergence rate into account. Hereto, consider the bounded control law

$$v_1 = -\operatorname{sat}_{\varepsilon_1}(k_1 v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1}) + k_1 \dot{e}_{\sigma,1}), \tag{4.38}$$

with the constants $\varepsilon_1, k_1, v_{\mathrm{tr}}^{\mathrm{d}}, \alpha > 0$. The positive constant $v_{\mathrm{tr}}^{\mathrm{d}}$ represents the desired walking speed. The control law (4.38) resembles a bounded PD controller, where the proportional term is replaced by the sigmoidal term $v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1})$. The convergence rate is specified by the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$, but only for values of $e_{\sigma,1}$ for which the sigmoidal term levels off. This is illustrated in Example 4.1, where a comparison is made between the closed-loop behavior of the double integrator $\ddot{e}_{\sigma,1} = v_1$ with (4.37) and with (4.38).

**Example 4.1.** Consider the double integrator $\ddot{e}_{\sigma,1} = v_1$ with the bounded PD controller (4.37), where $i = 1$, so that

$$\ddot{e}_{\sigma,1} = -\operatorname{sat}_{\varepsilon_1}(k_{\mathrm{P}1} e_{\sigma,1} + k_{\mathrm{D}1} \dot{e}_{\sigma,1}). \tag{4.39}$$

The phase portrait of (4.39) is shown in Figure 4.3a, where $\varepsilon_1 = 2$, $k_{\mathrm{P}1} = 2$, and $k_{\mathrm{D}1} = 1$. In addition, consider the double integrator $\ddot{e}_{\sigma,1} = v_1$ with the bounded controller (4.38), so that

$$\ddot{e}_{\sigma,1} = -\operatorname{sat}_{\varepsilon_1}(k_1 v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1}) + k_1 \dot{e}_{\sigma,1}). \tag{4.40}$$

The phase portrait of (4.40) is shown in Figure 4.3b, where $\varepsilon_1 = 2$, $k_1 = 1$, $v_{\mathrm{tr}}^{\mathrm{d}} = 2$, and $\alpha = 1$. Hereby, observe that in the two-dimensional state space $(e_{\sigma,1}, \dot{e}_{\sigma,1}) \in \mathbb{R}^2$, near the line $\{(e_{\sigma,1}, \dot{e}_{\sigma,1}) : e_{\sigma,1} = 0\}$, the hyperbolic tangent function $\tanh(\alpha e_{\sigma,1}) \approx \alpha e_{\sigma,1}$. In the same way, it is also true that far away from the line $\{(e_{\sigma,1}, \dot{e}_{\sigma,1}) : e_{\sigma,1} = 0\}$, the hyperbolic tangent function $\tanh(\alpha e_{\sigma,1}) \approx \operatorname{sign}(e_{\sigma,1})$. Thus, near the line $\{(e_{\sigma,1}, \dot{e}_{\sigma,1}) : e_{\sigma,1} = 0\}$, the system (4.40) can be approximated by

$$\ddot{e}_{\sigma,1} = -\operatorname{sat}_{\varepsilon_1}(k_1 v_{\mathrm{tr}}^{\mathrm{d}} \alpha e_{\sigma,1} + k_1 \dot{e}_{\sigma,1}). \tag{4.41}$$

Notice that this system behaves just like (4.39), where the control parameters are chosen such that $k_1 v_{\mathrm{tr}}^{\mathrm{d}} \alpha = k_{\mathrm{P}1}$ and $k_1 = k_{\mathrm{D}1}$. In addition, far away from the line $\{(e_{\sigma,1}, \dot{e}_{\sigma,1}) : e_{\sigma,1} = 0\}$, the system (4.40) can be approximated by

$$\ddot{e}_{\sigma,1} = \begin{cases} -\operatorname{sat}_{\varepsilon_1}(k_1 \dot{e}_{\sigma,1} - k_1 v_{\mathrm{tr}}^{\mathrm{d}}), & e_{\sigma,1} \ll 0 \\ -\operatorname{sat}_{\varepsilon_1}(k_1 \dot{e}_{\sigma,1} + k_1 v_{\mathrm{tr}}^{\mathrm{d}}), & e_{\sigma,1} \gg 0. \end{cases} \tag{4.42}$$

In both cases of (4.42), the dynamics of the system is described by a first-order differential equation of the variable $\dot{e}_{\sigma,1}$. Furthermore, in each case the system has a single asymptotically stable equilibrium point, namely

$$\dot{e}_{\sigma,1} = \begin{cases} +v_{\mathrm{tr}}^{\mathrm{d}}, & e_{\sigma,1} \ll 0 \\ -v_{\mathrm{tr}}^{\mathrm{d}}, & e_{\sigma,1} \gg 0. \end{cases} \tag{4.43}$$

It follows that the behavior of (4.40) is such that the velocity of the torso $\dot{x}_{\mathrm{tr}} = \dot{e}_{\sigma,1}$ tends to (4.43) if $\left|x_{\mathrm{tr}} - x_{\mathrm{tr}}^{\mathrm{d}}\right| = |e_{\sigma,1}| \gg 0$, and otherwise the behavior of (4.40) is similar to (4.39). △

So far, the control law for $\boldsymbol{v} = (v_1, v_2, v_3, v_4)$ reads as

$$v_1 = -\operatorname{sat}_{\varepsilon_1}(k_1 v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1}) + k_1 \dot{e}_{\sigma,1}), \tag{4.44}$$

$$v_i = -\operatorname{sat}_{\varepsilon_i}(k_{\mathrm{P}i} e_{\sigma,i} + k_{\mathrm{D}i} \dot{e}_{\sigma,i}), \tag{4.45}$$

for $i = 2, 3, 4$. Now, consider that the accelerations $\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ and $\bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ from Section 3.5 are added to the control law, such that

$$v_1 = -\operatorname{sat}_{\varepsilon_1}(k_1 v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1}) + k_1 \dot{e}_{\sigma,1}) + k_0 \frac{\min(|\dot{x}_{\mathrm{tr}}|, v_{\mathrm{tr}}^{\mathrm{d}})}{v_{\mathrm{tr}}^{\mathrm{d}}} \bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}^{\mathrm{d}}), \tag{4.46}$$

$$v_2 = -\operatorname{sat}_{\varepsilon_2}(k_{\mathrm{P}2} e_{\sigma,2} + k_{\mathrm{D}2} \dot{e}_{\sigma,2}) + k_0 \frac{\min(|\dot{x}_{\mathrm{tr}}|, v_{\mathrm{tr}}^{\mathrm{d}})}{v_{\mathrm{tr}}^{\mathrm{d}}} \bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}^{\mathrm{d}}), \tag{4.47}$$

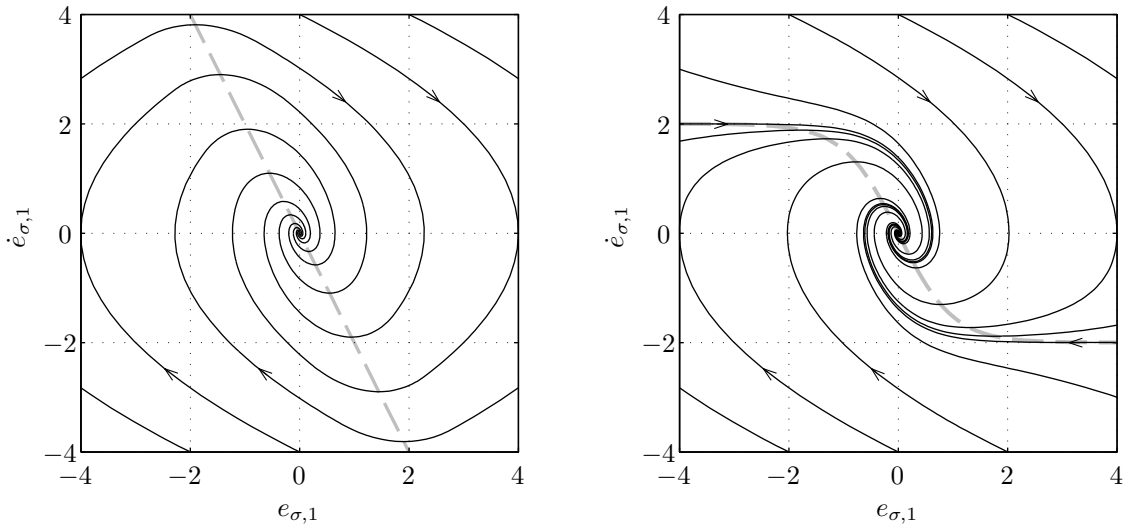



Figure 4.3: (a) The phase portrait of (4.39) with $\varepsilon_1 = 2$, $k_{\mathrm{P}1} = 2$, and $k_{\mathrm{D}1} = 1$, which includes the gray dashed nullcline $\ddot{e}_{\sigma,1} = 0$. (b) The phase portrait of (4.40) with $\varepsilon_1 = 2$, $k_1 = 1$, $v_{\mathrm{tr}}^{\mathrm{d}} = 2$, and $\alpha = 1$, which also includes the gray dashed nullcline $\ddot{e}_{\sigma,1} = 0$. Both phase portraits belong to Example 4.1.

where the gain $k_0 \geq 0$ and the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} > 0$. Note that the location of the torso $x_{\mathrm{tr}} = e_{\sigma,1} + x_{\mathrm{tr}}^{\mathrm{d}}$, and the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}} = \dot{e}_{\sigma,1}$, where the constant $x_{\mathrm{tr}}^{\mathrm{d}}$ represents the desired location of the torso. In Section 3.5, it is shown that the accelerations $\bar{a}_1, \bar{a}_2$ are given by

$$\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}) = g_{\mathrm{a}} \left( \frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2} l_{\mathrm{tr}}} \right) \left( \left( \frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2} l_{\mathrm{tr}}} \right)^2 + 1 \right)^{-1},$$

$$\bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}) = -g_{\mathrm{a}} \left( \frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2} l_{\mathrm{tr}}} \right)^2 \left( \left( \frac{x_{\mathrm{tr}} - x_{\mathrm{stf}}}{y_{\mathrm{tr}} - \frac{1}{2} l_{\mathrm{tr}}} \right)^2 + 1 \right)^{-1}.$$

In the control law above, the last function argument of both $\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ and $\bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ is set to the desired elevation of the torso $y_{\mathrm{tr}}^{\mathrm{d}}$, which is a constant. Also recall from Section 3.5 that the accelerations $\bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ and $\bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}})$ are the components of the vector $\bar{\boldsymbol{a}} = \bar{a}_1 \boldsymbol{e}_1 + \bar{a}_2 \boldsymbol{e}_2$, where $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are the unit vectors of the inertial frame. The vector $\bar{\boldsymbol{a}}$ represents the gravitational acceleration of the COM of an inverted pendulum in the tangential direction along the circular path that the COM follows. The accelerations $\bar{a}_1, \bar{a}_2$ are added in parallel to the first two bounded (PD) controllers of the control law in order to incorporate inverted pendulum-like motion into the walking gait. The hope is that this will reduce energy consumption, because the added gravitational acceleration terms will force the robot to walk more in a manner that is in accordance with the natural tendencies of the robot. These tendencies of the robot are to tip over under the influence of gravity, just like an inverted pendulum. In the next chapter, in Section 5.4, a comparison is made in terms of energy consumption between the control design with $k_0 = 0$ and with $k_0 > 0$.

The multiplication of the accelerations $\bar{a}_1, \bar{a}_2$ by the term $\min(|\dot{x}_{\mathrm{tr}}|, v_{\mathrm{tr}}^{\mathrm{d}})/v_{\mathrm{tr}}^{\mathrm{d}}$ ensures that no new equilibrium points are created, while the original accelerations are retained for $|\dot{x}_{\mathrm{tr}}| \geq v_{\mathrm{tr}}^{\mathrm{d}}$. The gain $k_0 \geq 0$ enables both the amplification as well as the straightforward elimination of the effect of the added accelerations to the behavior of the closed-loop system. Notice that it is allowed to add these accelerations to the control law, because the first two control inputs $v_1, v_2$ are accelerations as well. This follows from the relation $(v_1, v_2) = (\ddot{e}_{\sigma,1}, \ddot{e}_{\sigma,2}) = (\ddot{x}_{\mathrm{tr}}, \ddot{y}_{\mathrm{tr}})$. Lastly, note that with the addition of the accelerations $\bar{a}_1, \bar{a}_2$, the first two double integrators of (4.35) are now uni-directionally coupled, because the bounded PD controller for the second double integrator $\ddot{e}_{\sigma,2} = v_2$ now uses the location of the torso $x_{\mathrm{tr}} = e_{\sigma,1} + x_{\mathrm{tr}}^{\mathrm{d}}$ and the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}} = \dot{e}_{\sigma,1}$, where $(e_{\sigma,1}, \dot{e}_{\sigma,1})$ is the state of the first double integrator $\ddot{e}_{\sigma,1} = v_1$.

## 4.4 Maximizing the controller bounds

In the previous section, a bounded control law is proposed for the new control input $\boldsymbol{v}$, where the controller bounds are denoted by $\varepsilon_1, \ldots, \varepsilon_4$. Specifically, recall that the control law is given by (4.44)-(4.45) as well as (4.46)-(4.47). Now, for a given set of states of the closed-loop system $\mathcal{S}$, let us formulate an optimization problem that maximizes the controller bounds $\varepsilon_1, \ldots, \varepsilon_4 > 0$, such that the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$ is satisfied for the states $(\boldsymbol{x}, \rho) \in \mathcal{S}$, and that the contact constraint $|F_{\mathrm{f},\sigma}| \leq \mu F_{\mathrm{n},\sigma}$ is satisfied for the states $(\boldsymbol{x}, \rho) \in \mathcal{S}$. The first constraint must be satisfied in order to achieve input-output linearization, and to ensure that the control law is executed properly. The second constraint must be satisfied in order to keep the stance foot in contact with the ground without slipping. The design variables of this optimization problem are given by the vector of controller bounds

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 & \varepsilon_2 & \varepsilon_3 & \varepsilon_4 \end{bmatrix}^\top. \tag{4.48}$$

The input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$, or equivalently $\pm\boldsymbol{u} - \boldsymbol{L} \leq \boldsymbol{0}$, is expressed in terms of $\boldsymbol{\varepsilon}$ and $(\boldsymbol{x}, \rho) = (\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho)$ by using the existing control law

$$\boldsymbol{u} = \boldsymbol{D}_\sigma^{-1}(\boldsymbol{q}, \rho) \left( \boldsymbol{C}_\sigma(\boldsymbol{q}, \rho) \left( \boldsymbol{A}_{\sigma,11}(\boldsymbol{q}) \boldsymbol{H}(\dot{\boldsymbol{q}}) + \boldsymbol{A}_{\sigma,12}(\boldsymbol{q}) \boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) \right) - \boldsymbol{b}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) + \boldsymbol{v} \right),$$

where

$$\boldsymbol{v} = -\,\mathbf{sat}_{\boldsymbol{\varepsilon}} \left( \begin{bmatrix} k_1 v_{\mathrm{tr}}^{\mathrm{d}} \tanh(\alpha e_{\sigma,1}) + k_1 \dot{e}_{\sigma,1} \\ k_{\mathrm{P}2} e_{\sigma,2} + k_{\mathrm{D}2} \dot{e}_{\sigma,2} \\ k_{\mathrm{P}3} e_{\sigma,3} + k_{\mathrm{D}3} \dot{e}_{\sigma,3} \\ k_{\mathrm{P}4} e_{\sigma,4} + k_{\mathrm{D}4} \dot{e}_{\sigma,4} \end{bmatrix} \right) + k_0 \frac{\min(|\dot{x}_{\mathrm{tr}}|, v_{\mathrm{tr}}^{\mathrm{d}})}{v_{\mathrm{tr}}^{\mathrm{d}}} \begin{bmatrix} \bar{a}_1(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}^{\mathrm{d}}) \\ \bar{a}_2(x_{\mathrm{tr}}, x_{\mathrm{stf}}, y_{\mathrm{tr}}^{\mathrm{d}}) \\ 0 \\ 0 \end{bmatrix}.$$

Notice that only the vector-valued saturation function $\mathbf{sat}_{\boldsymbol{\varepsilon}}(\cdot)$ utilizes the vector of controller bounds (4.48). For the optimization, let us assume that $\psi = \dot{x}_{\mathrm{tr}}$, so that from (4.17b) it follows that $\rho = x_{\mathrm{tr}} - \omega_0^{-1} \dot{x}_{\mathrm{tr}}$. In addition, let us stick to the primary control design and set the gain $k_0 = 0$. Now, observe that at a given state $(\boldsymbol{x}, \rho)$ all four components of the input vector $\boldsymbol{u}$ are at an extremum if the vector-valued saturation function $\mathbf{sat}_{\boldsymbol{\varepsilon}}(\cdot)$ is saturated in all of its four components. This can happen in $2^4 = 16$ different ways. To illustrate this, observe that the image of the vector-valued saturation function is given by

$$\mathbf{sat}_{\boldsymbol{\varepsilon}}(\mathbb{R}^4) = [-\varepsilon_1, \varepsilon_1] \times [-\varepsilon_2, \varepsilon_2] \times [-\varepsilon_3, \varepsilon_3] \times [-\varepsilon_4, \varepsilon_4], \tag{4.49}$$

which is an orthotope in four dimensions centered around the origin. An orthotope is the generalization of a rectangle for higher dimensions. The 4-dimensional orthotope (4.49) has $2^4 = 16$ vertices. These 16 vertices correspond to the 16 possible cases of complete saturation of the vector-valued saturation function $\mathbf{sat}_{\boldsymbol{\varepsilon}}(\cdot)$. Let us consider all 16 vertices in order to check whether the input constraint is satisfied. It is sufficient to check only the vertices of the orthotope because of the linearity of the input constraint. Each vertex is assigned a unique label from the set $\{\mathrm{V1}, \mathrm{V2}, \mathrm{V3}, \ldots, \mathrm{V16}\}$. The position in $\mathbb{R}^4$ of the $i$-th vertex V$i$ is determined by the function

$$\boldsymbol{\xi}_{\mathrm{V}i}(\boldsymbol{\varepsilon}) = \begin{bmatrix} \varepsilon_1 \cdot (-1)^{\lfloor (i-1)/1 \rfloor} \\ \varepsilon_2 \cdot (-1)^{\lfloor (i-1)/2 \rfloor} \\ \varepsilon_3 \cdot (-1)^{\lfloor (i-1)/4 \rfloor} \\ \varepsilon_4 \cdot (-1)^{\lfloor (i-1)/8 \rfloor} \end{bmatrix}, \tag{4.50}$$

where the floor function $\lfloor \nu \rfloor \coloneqq \max\{\kappa \in \mathbb{Z} : \kappa \leq \nu\}$. For brevity, let us define the function

$$\boldsymbol{s}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) = \boldsymbol{C}_\sigma(\boldsymbol{q}, \rho) \left( \boldsymbol{A}_{\sigma,11}(\boldsymbol{q}) \boldsymbol{H}(\dot{\boldsymbol{q}}) + \boldsymbol{A}_{\sigma,12}(\boldsymbol{q}) \boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) \right) - \boldsymbol{b}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho), \tag{4.51}$$

so that the control law becomes $\boldsymbol{u} = \boldsymbol{D}_\sigma^{-1}(\boldsymbol{q}, \rho) \left( \boldsymbol{s}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) - \boldsymbol{\xi}_{\mathrm{V}i}(\boldsymbol{\varepsilon}) \right)$ at vertex V$i$. Now, the input constraint $\pm\boldsymbol{u} - \boldsymbol{L} \leq \boldsymbol{0}$ can be expressed at vertex V$i$ as

$$\pm\, \boldsymbol{D}_\sigma^{-1}(\boldsymbol{q}, \rho) \left( \boldsymbol{s}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) - \boldsymbol{\xi}_{\mathrm{V}i}(\boldsymbol{\varepsilon}) \right) - \boldsymbol{L} \leq \boldsymbol{0}, \tag{4.52}$$

for $i = 1, \ldots, 16$.

The contact constraint $|F_{\mathrm{f},\sigma}| \leq \mu F_{\mathrm{n},\sigma}$, or equivalently $\pm F_{\mathrm{f},\sigma} - \mu F_{\mathrm{n},\sigma} \leq 0$, is expressed in terms of $\boldsymbol{\varepsilon}$ and $(\boldsymbol{x}, \rho) = (\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho)$ using the fact that the friction contact force $F_{\mathrm{f},\sigma}$ and the normal contact force $F_{\mathrm{n},\sigma}$

are both kinematic constraint forces. Recall from Section 3.2 that the vector of kinematic constraint forces is given by

$$\boldsymbol{\lambda}_\sigma = \begin{bmatrix} \boldsymbol{\lambda}^\top & F_{\mathrm{f},\sigma} & F_{\mathrm{n},\sigma} \end{bmatrix}^\top,$$

where the friction contact force (3.21), and the normal contact force (3.22). According to the multibody dynamics (3.26), with (3.28) and (3.29), it holds that

$$\boldsymbol{\lambda}_\sigma = \boldsymbol{A}_{\sigma,21}(\boldsymbol{q}) \left( \boldsymbol{S}\boldsymbol{\tau} - \boldsymbol{H}(\dot{\boldsymbol{q}}) \right) - \boldsymbol{A}_{\sigma,22}(\boldsymbol{q}) \boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}), \tag{4.53}$$

where $\boldsymbol{\tau} = \mathbf{sat}_{\boldsymbol{L}}(\boldsymbol{u})$. Next, let us define the function

$$\boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon}) = \boldsymbol{A}_{\sigma,21}(\boldsymbol{q}) \left( \boldsymbol{S}\boldsymbol{D}_\sigma^{-1}(\boldsymbol{q}, \rho) \left( \boldsymbol{s}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) - \boldsymbol{\xi}_{\mathrm{V}i}(\boldsymbol{\varepsilon}) \right) - \boldsymbol{H}(\dot{\boldsymbol{q}}) \right) - \boldsymbol{A}_{\sigma,22}(\boldsymbol{q}) \boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}). \tag{4.54}$$

Observe that if the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$ is satisfied, then $\boldsymbol{\tau} = \boldsymbol{u}$ and thus it holds that $\boldsymbol{\lambda}_\sigma = \boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon})$ at vertex V$i$. Hereby, let $\boldsymbol{p}_\mathrm{f}$ and $\boldsymbol{p}_\mathrm{n}$ be constant vectors, such that $\boldsymbol{p}_\mathrm{f}^\top \boldsymbol{\lambda}_\sigma = F_{\mathrm{f},\sigma}$ and $\boldsymbol{p}_\mathrm{n}^\top \boldsymbol{\lambda}_\sigma = F_{\mathrm{n},\sigma}$. Now, the contact constraint $\pm F_{\mathrm{f},\sigma} - \mu F_{\mathrm{n},\sigma} \leq 0$ can be expressed at vertex V$i$ as

$$\pm\, \boldsymbol{p}_\mathrm{f}^\top \boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon}) - \mu \boldsymbol{p}_\mathrm{n}^\top \boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon}) \leq 0, \tag{4.55}$$

for $i = 1, \ldots, 16$.

Consider that the aforementioned set

$$\begin{aligned} \mathcal{S} = \big\{ (\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) \in \mathbb{R}^{2n+1} : \; & \rho = x_\mathrm{tr} - \omega_0^{-1} \dot{x}_\mathrm{tr}, x_\mathrm{tr} = 0, y_\mathrm{tr} = y_\mathrm{tr}^\mathrm{d}, \dot{x}_\mathrm{tr} = 0, \dot{y}_\mathrm{tr} = 0, \\ & x_{\mathrm{stf},\sigma}(\boldsymbol{q}) = 0, y_{\mathrm{stf},\sigma}(\boldsymbol{q}) = 0, \dot{x}_{\mathrm{stf},\sigma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0, \dot{y}_{\mathrm{stf},\sigma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0, \\ & x_{\mathrm{swf},\sigma}(\boldsymbol{q}) = 0, y_{\mathrm{swf},\sigma}(\boldsymbol{q}) = 0, \dot{x}_{\mathrm{swf},\sigma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0, \dot{y}_{\mathrm{swf},\sigma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0 \big\}. \end{aligned} \tag{4.56}$$

This set contains one state that will often be used as the initial state of the robot. The singleton set (4.56) is a simple yet natural choice for the initial state of the robot, whereby the robot is located at the origin. Actually, it does not matter where the robot is initially located. However, it makes sense to place the robot at the origin, and to let it walk to a desired location $x_\mathrm{tr}^\mathrm{d} \neq 0$.

The reason why an initial state is used to test whether the input constraint and the contact constraint are satisfied is because the initial state of the robot is likely to deviate the most from the desired walking gait, especially an initial state such as the one specified in (4.56). Therefore, it is reasonable to assume that the initial state will give the largest output error $\boldsymbol{e}_\sigma$ and the largest output error rate $\dot{\boldsymbol{e}}_\sigma$ of all states on a solution curve of the closed-loop system. In other words, if the input constraint and/or the contact constraint are violated during walking, then the initial state $(\boldsymbol{x}, \rho) \in \mathcal{S}$ will most likely be the state at which the violation occurred. Simulations of the closed-loop system confirm that if the controller bounds $\boldsymbol{\varepsilon}$ are set too large, then the contact constraint will indeed be violated immediately at the start of the simulation.

The optimization problem is formulated as follows:

$$\begin{aligned} \bar{\boldsymbol{\varepsilon}} = \underset{\boldsymbol{\varepsilon}}{\mathrm{argmax}} \quad & \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \varepsilon_4 \\ \text{subject to} \quad & \varepsilon_1, \ldots, \varepsilon_4 \geq 0 \\ & \pm\, \boldsymbol{D}_\sigma^{-1}(\boldsymbol{q}, \rho) \left( \boldsymbol{s}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) - \boldsymbol{\xi}_{\mathrm{V}i}(\boldsymbol{\varepsilon}) \right) - \boldsymbol{L} \leq \boldsymbol{0}, \; i = 1, \ldots, 16, \; (\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) \in \mathcal{S} \\ & \pm\, \boldsymbol{p}_\mathrm{f}^\top \boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon}) - \mu \boldsymbol{p}_\mathrm{n}^\top \boldsymbol{\lambda}_{\sigma,\mathrm{V}i}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho, \boldsymbol{\varepsilon}) \leq 0, \; i = 1, \ldots, 16, \; (\boldsymbol{q}, \dot{\boldsymbol{q}}, \rho) \in \mathcal{S} \\ & \varepsilon_1 = \varepsilon_2 = \tfrac{1}{2}\varepsilon_3 = \tfrac{1}{2}\varepsilon_4. \end{aligned} \tag{4.57}$$

Notice that all constraints of (4.57) are linear in $\boldsymbol{\varepsilon}$. The objective function is linear as well. Clearly, the optimization problem (4.57) is a linear programming problem. The set of points that satisfies all constraints of the optimization problem is called the feasible region. In linear programming, the feasible region is convex, and any local optimum is a global optimum.

It is desired that each individual controller bound $\varepsilon_1, \ldots, \varepsilon_4$ is maximized, and that none of the constraints $\varepsilon_1, \ldots, \varepsilon_4 \geq 0$ are active. This is accomplished within the rules of linear programming by the addition of the equality constraints

$$\varepsilon_1 = \varepsilon_2 = \tfrac{1}{2}\varepsilon_3 = \tfrac{1}{2}\varepsilon_4. \tag{4.58}$$

Recall that the controller bound $\varepsilon_1$ restricts the acceleration of the torso in the horizontal direction. Similarly, the controller bound $\varepsilon_3$ restricts the acceleration of the swing foot in the horizontal direction. During every stride, the swing foot has to travel approximately twice the distance that the torso has to travel in the same amount of time. To accommodate for this, a factor of two is introduced in the equality constraints (4.58), whereby the controller bounds of the torso and the swing foot are kept pairwise equal to each other.

The line segment that extends from $\boldsymbol{\varepsilon} = \mathbf{0}$ to the solution of (4.57) necessarily satisfies the equality constraints (4.58). This line segment also lies in the feasible region due to the convexity of the optimization problem. Thus, every point on the line segment, except $\boldsymbol{\varepsilon} = \mathbf{0}$, can be chosen to be the actual controller bounds, not just the maximal controller bounds $\bar{\boldsymbol{\varepsilon}}$ that are obtained by the optimization. Therefore, let the maximal controller bounds $\bar{\boldsymbol{\varepsilon}}$ only serve as an upper bound for the specification of the controller bounds, with $\mathbf{0} < \boldsymbol{\varepsilon} < \bar{\boldsymbol{\varepsilon}}$.

The optimization problem is formulated based on the assumption of complete saturation. There are $2^4 = 16$ different cases of complete saturation that are possible with the vector-valued saturation function $\mathbf{sat}_{\boldsymbol{\varepsilon}}(\cdot)$. All 16 cases are considered during the optimization, and each case represents a different set of inequality constraints in the optimization. However, only the most restrictive case of complete saturation will produce active inequality constraints. Thus, if this most restrictive case differs from the case of complete saturation that actually occurs at the initial state $\mathcal{S}$, then the optimization (4.57) will produce conservative results.

## 4.5 Summary

The desired position of the swing foot is specified as a function of the state to obtain a dynamic walking gait. This desired position consists of a desired location and a desired elevation of the swing foot. The desired location is defined such that the swing foot and the stance foot are always on opposite sides of the torso. The swing foot is placed on the ground at the Capture Point as soon as the horizontal distance between the torso and the stance foot exceeds a given threshold, but only if the torso is moving away from the stance foot. In addition, an event function is defined, which is equal to zero at the desired foot placement location. Moreover, this event function is positive in the region before the foot placement, and it is negative in the region after the foot placement. Therefore, the desired elevation of the swing foot is simply specified as a sigmoid function of the event function.

Then, input-output linearization is performed with an output vector of the same dimension as the input vector of the robotic system. This means that the multivariable system is square. The output vector is defined so that the first two outputs represent the position of the hips, and the last two outputs represent the error in the position of the swing foot. A tame differentiator of the location of the torso is used to replace the velocity of the torso in the output vector. This is done to ensure that the decoupling matrix is non-singular.

Dynamic walking to a desired location is accomplished by the stabilization of the origin of the output error dynamics, whereby the desired output vector only consists of the desired position of the torso/hips. Bounded controllers are used for the stabilization of the output errors. Hereby, a particular bounded controller is used for the stabilization of the first output error. This controller takes into account how fast the error of the first output approaches zero, which is important because it represents the walking speed of the robot. In addition, the previously derived gravitational acceleration terms are added to the control law with the intention to decrease the energy consumption of the overall control design.

Lastly, an optimization problem is formulated to determine the maximal controller bounds. This is done with the constraint that the applied torques do not saturate during walking, and that the stance foot stays in contact with the ground without slipping. The first constraint must be satisfied in order to achieve input-output linearization, and to ensure that the control law is executed properly. Both constraints are evaluated at a particular initial state of the robot that will often be used in simulations in the next chapter.

# Chapter 5

# Gait analysis and simulations of the closed-loop system

In this chapter, an approximation of the maximum achievable walking speed is derived analytically for the given control design. This maximum achievable walking speed is then used together with the maximal controller bounds for the specification of the control parameters. This is done in such a way that the robot does not slip nor fall over during walking. For a particular specification of the parameters, multiple simulations are performed to show the behavior of the closed-loop system. Hereby, several important aspects of the walking gait are analyzed in detail. Lastly, the energy efficiency of the walking gait is discussed, and it is shown that the energy efficiency changes according to the specification of some of the control parameters.

## 5.1 An analytical approximation of the maximum achievable walking speed

Intuitively, the boundedness of the control design suggests that the walking speed is limited as well. An analytical approximation of the maximum achievable walking speed can be derived based on the desired gait. In this derivation it is assumed that the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$ matches the average walking speed of the robot. The velocity of the torso $\dot{x}_{\mathrm{tr}}$ is considered as the actual walking speed of the robot.

Recall from the previous chapter that the desired gait is mostly defined by the desired position of the swing foot, which is given by

$$\begin{aligned} x_{\mathrm{swf}}^{\mathrm{d}} &= x_{\mathrm{tr}} + \tilde{\eta}(x_{\mathrm{tr}}, \psi), \\ y_{\mathrm{swf}}^{\mathrm{d}} &= \bar{y}_{\mathrm{swf}} \tanh(\phi \tilde{\zeta}(x_{\mathrm{tr}}, \psi)), \end{aligned}$$

where $\psi$ is the output of the tame differentiator, and $\bar{y}_{\mathrm{swf}}, \phi > 0$ are positive constants. For simplicity, it is assumed that $\psi = \dot{x}_{\mathrm{tr}} > 0$ here. The function $\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$ represents the desired location of the swing foot relative to the torso, and it is defined as

$$\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = \gamma \tanh(\gamma^{-1} (x_{\mathrm{tr}} - x_{\mathrm{stf}})) \tanh(a \dot{x}_{\mathrm{tr}}) \tanh(\gamma^{-1} c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}),$$

where $x_{\mathrm{stf}}$ is the location of the stance foot, and $\gamma, c > 0$, $a \gg 1$ are positive constants. Also note that the location of the stance foot is piecewise constant during walking. In addition, the function $\tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$ is defined as

$$\begin{aligned} \tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = &\left( \gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}} + \sqrt{(\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}})^2 + \delta} \right) \\ &\cdot \left( \gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \dot{x}_{\mathrm{tr}} + \sqrt{(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \dot{x}_{\mathrm{tr}})^2 + \delta} \right), \end{aligned}$$

where $\delta$ is a small non-negative number. If the function $\tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = 0$, then the desired elevation of the swing foot $y_{\mathrm{swf}}^{\mathrm{d}} = 0$, and thus the swing foot should be placed on the ground. Now, observe that if

the constant $\delta = 0$ and the velocity of the torso $\dot{x}_{\mathrm{tr}} > 0$, then the equation $\tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = 0$ reduces to $x_{\mathrm{tr}} - x_{\mathrm{stf}} = \gamma$. In other words, the swing foot should be placed on the ground as soon as the distance between the location of the torso $x_{\mathrm{tr}}$ and the location of the stance foot $x_{\mathrm{stf}}$ is equal to $\gamma$.

Next, consider the third output error $e_{\sigma,3} = x_{\mathrm{swf}} - x^{\mathrm{d}}_{\mathrm{swf}}$, which represents the error in the location of the swing foot. The control design for the third output error is such that

$$\ddot{e}_{\sigma,3} = -\operatorname{sat}_{\varepsilon_3}(k_{\mathrm{P}3} e_{\sigma,3} + k_{\mathrm{D}3} \dot{e}_{\sigma,3}),$$

where it is important to notice the boundedness of $\ddot{e}_{\sigma,3}$. Also, observe that if $|\dot{x}_{\mathrm{tr}}|$ is large enough and $|x_{\mathrm{tr}} - x_{\mathrm{stf}}|$ is small enough, then the function $\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) \approx x_{\mathrm{tr}} - x_{\mathrm{stf}}$, so that the desired location of the swing foot $x^{\mathrm{d}}_{\mathrm{swf}} \approx 2x_{\mathrm{tr}} - x_{\mathrm{stf}}$, and the desired velocity of the swing foot $\dot{x}^{\mathrm{d}}_{\mathrm{swf}} \approx 2\dot{x}_{\mathrm{tr}}$. It turns out that the desired velocity of the swing foot $\dot{x}^{\mathrm{d}}_{\mathrm{swf}}$ is more or less constant during steady state walking. This makes sense because the velocity of the torso $\dot{x}_{\mathrm{tr}} \approx \pm v^{\mathrm{d}}_{\mathrm{tr}}$ during steady state walking as well. Therefore, it is assumed that the desired acceleration of the swing foot $\ddot{x}^{\mathrm{d}}_{\mathrm{swf}} = 0$. The actual acceleration of the swing foot must satisfy $|\ddot{x}_{\mathrm{swf}}| \leq \varepsilon_3$ according to the control design mentioned above. One can imagine that if the robot is walking forward with a relatively high speed, then the acceleration of the swing foot will be $\ddot{x}_{\mathrm{swf}} = \varepsilon_3$ in order to keep up with the desired location of the swing foot $x^{\mathrm{d}}_{\mathrm{swf}}$ and to avoid falling over. Hereby, it is important to note that the swing foot is stationary at the start of every stride. This is because after every step, both feet exchange roles between being the swing foot and the stance foot. Another thing to note is that the torso is controlled independently of the swing foot, and the robot walks to the desired location with a speed that is close to the desired walking speed $v^{\mathrm{d}}_{\mathrm{tr}}$, as is discussed in Section 4.3. So, the swing foot has to really keep up with the torso in order to avoid falling over, because the torso will just keep on moving at the desired walking speed.

Let us assume the ideal case of steady state walking with $\dot{x}_{\mathrm{tr}} > 0$, whereby the swing foot is placed on the ground as soon as $x_{\mathrm{tr}} - x_{\mathrm{stf}} = \gamma$. Hereby, it is assumed that the location of the swing foot $x_{\mathrm{swf}} = x^{\mathrm{d}}_{\mathrm{swf}} = x_{\mathrm{tr}} + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$ at the moment of foot placement. So, at every foot placement it holds that $x_{\mathrm{tr}} = x_{\mathrm{stf}} + \gamma = x_{\mathrm{swf}} - \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$. Also notice that if $x_{\mathrm{tr}} - x_{\mathrm{stf}} = \gamma$ and $\dot{x}_{\mathrm{tr}} > 0$, then $0 < \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) < \gamma \tanh(1)$. Now, observe that after the foot placement, where both feet exchange roles between being the swing foot and the stance foot, it holds that $x_{\mathrm{tr}} = x_{\mathrm{swf}} + \gamma = x_{\mathrm{stf}} - \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$. The difference between the location of the torso at the end of a stride and the location of the torso at the start of a stride represents the distance traveled by the torso during one stride. In the ideal case, as described here, this distance is equal to

$$(x_{\mathrm{stf}} + \gamma) - (x_{\mathrm{stf}} - \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})) = \gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}). \tag{5.1}$$

The location of the stance foot $x_{\mathrm{stf}}$ can be used as a reference point, because it stays constant during every stride. Similarly, observe that the distance traveled by the swing foot during one stride is equal to

$$(x_{\mathrm{stf}} + \gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})) - (x_{\mathrm{stf}} - \gamma - \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})) = 2\,(\gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})). \tag{5.2}$$

In addition, if the duration of one stride of steady state walking is denoted by $t^*$, then the distance traveled by the torso during one stride can also be given by $v^{\mathrm{d}}_{\mathrm{tr}} t^*$, because the desired walking speed $v^{\mathrm{d}}_{\mathrm{tr}}$ is assumed to match the average velocity of the torso. On the other hand, the distance traveled by the swing foot during one stride is at most $\frac{1}{2}\varepsilon_3 (t^*)^2$, because the acceleration of the swing foot is assumed to satisfy $|\ddot{x}_{\mathrm{swf}}| \leq \varepsilon_3$. Now, if the swing foot just manages to travel the distance that is required to keep up with the torso, then it holds that

$$v^{\mathrm{d}}_{\mathrm{tr}} t^* = \gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}), \tag{5.3}$$

$$\tfrac{1}{2}\varepsilon_3 (t^*)^2 = 2\,(\gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})). \tag{5.4}$$

This system of equations is solved for $v^{\mathrm{d}}_{\mathrm{tr}}$ by the elimination of $t^*$. The solution is given by

$$v^{\mathrm{d}}_{\mathrm{tr}} = \sqrt{\frac{1}{4}\varepsilon_3\,(\gamma + \tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}))}, \tag{5.5}$$

which represents the maximum achievable walking speed of the robot in relation to the desired walking speed $v^{\mathrm{d}}_{\mathrm{tr}} > 0$. Unfortunately, in the current form this result is not of much use, because it depends on the state of the torso $(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$. However, in the derivation above it is pointed out that the function

$\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) > 0$. Therefore, consider that the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$ is specified such that it satisfies the constraint

$$0 < v_{\mathrm{tr}}^{\mathrm{d}} \leq \sqrt{\tfrac{1}{4}\varepsilon_3 \gamma}. \tag{5.6}$$

The parameter constraint (5.6) can be used to avoid stumbling over, which is the kind of falling over that occurs if the swing foot fails to keep up with the torso. The right-hand side of (5.6) represents the analytical approximation of the maximum achievable walking speed.

## 5.2 The specification of the control parameters

Consider the specification of the control parameters that is presented in Table 5.1. This specification is used to obtain numerical results in the remainder of this chapter. Notice that the gain $k_0$ is assigned two values. Both values are used in simulations of the closed-loop system in order to compare the behavior of the closed-loop system with, and without the addition of the gravitational acceleration terms $\bar{a}_1, \bar{a}_2$ to the control law. The gain $k_0$ is in fact the gain for these accelerations, as discussed in Section 4.3. Also, notice that some control parameters are assigned a value that is constructed from an angular frequency. For example, the pole of the tame differentiator (4.18) is placed at $\omega_0 = 2\pi 20\,\mathrm{rad/s} = 20\,\mathrm{Hz}$. Similarly, the PD control gains $k_{\mathrm{P}i}$ and $k_{\mathrm{D}i}$ of the bounded PD controller (4.37) are specified such that the proportional gain $k_{\mathrm{P}i} = \omega_{\mathrm{n}i}^2$ and the derivative gain $k_{\mathrm{D}i} = 2\omega_{\mathrm{n}i}$, with the angular frequencies $\omega_{\mathrm{n}1}, \omega_{\mathrm{n}2} = 2\pi 1\,\mathrm{rad/s}$ and $\omega_{\mathrm{n}3}, \omega_{\mathrm{n}4} = 2\pi 3\,\mathrm{rad/s}$. The PD control gains are set small enough so that the effects of the added accelerations $\bar{a}_1, \bar{a}_2$ are visible in the simulations of the closed-loop system. Note that the above ratio of PD control gains results in a bounded PD controller that provides critical damping in the non-saturated case. For the modified PD controller (4.38), consider that the proportional gain $k_{\mathrm{P}1} = k_1 v_{\mathrm{tr}}^{\mathrm{d}} \alpha$ and the derivative gain $k_{\mathrm{D}1} = k_1$. The specification of the control parameters is such that $k_1 = 2\omega_{\mathrm{n}1}$ and $k_1 v_{\mathrm{tr}}^{\mathrm{d}} \alpha = \omega_{\mathrm{n}1}^2$. So, if the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} = 1\,\mathrm{m/s}$, then it is assumed that the controller (4.38) provides critical damping to the system $\ddot{e}_{\sigma,1} = v_1$. Currently, the system is considered overdamped, because the specified desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s} < 1\,\mathrm{m/s}$.

For completeness, the desired position of the torso $\left(x_{\mathrm{tr}}^{\mathrm{d}}, y_{\mathrm{tr}}^{\mathrm{d}}\right)$ is included in Table 5.1 as well. Note that this desired position completely specifies the desired output vector

$$\boldsymbol{y}_{\sigma}^{\mathrm{d}} = \begin{bmatrix} x_{\mathrm{tr}}^{\mathrm{d}} & y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2} l_{\mathrm{tr}} & 0 & 0 \end{bmatrix}^{\top},$$

where $l_{\mathrm{tr}}$ is the length of the torso. The desired output vector is used as the setpoint for the closed-loop system, where $x_{\mathrm{tr}}^{\mathrm{d}}$ and $y_{\mathrm{tr}}^{\mathrm{d}}$ can be chosen freely. However, it is useful to set the desired elevation of the torso $y_{\mathrm{tr}}^{\mathrm{d}} = 0.8\,\mathrm{m}$ already, because it allows for the calculation of some important auxiliary control parameters that depend on it. In particular, observe that the desired maximum step size is now given by

$$\Delta_{\max}^{\mathrm{d}} = 2\sqrt{(l_{\mathrm{th}} + l_{\mathrm{sh}})^2 - \left(y_{\mathrm{tr}}^{\mathrm{d}} - \tfrac{1}{2} l_{\mathrm{tr}}\right)^2} = 0.7211\,\mathrm{m}, \tag{5.7}$$

from which it follows that the related distance

$$\gamma = \tfrac{1}{2}\beta \Delta_{\max}^{\mathrm{d}} = 0.1803\,\mathrm{m}. \tag{5.8}$$

To clarify, the distance $\gamma$ defines the maximum distance between the location of the torso $x_{\mathrm{tr}}$ and the confined Capture Point $\bar{x}_{\mathrm{CP}}$. Therefore, it also defines the furthest desired foot placement location relative to the location of the torso $x_{\mathrm{tr}}$. In addition, the distance $\gamma$ is also the threshold distance between the location of the torso $x_{\mathrm{tr}}$ and the location of the stance foot $x_{\mathrm{stf}}$ for which it is desired that the foot placement takes place. This means that the gait is designed with a maximum step size of $2\gamma = \beta \Delta_{\max}^{\mathrm{d}} < \Delta_{\max}^{\mathrm{d}}$. The desired maximum step size $\Delta_{\max}^{\mathrm{d}}$ can be interpreted as the largest feasible step size of the robot, which depends on the choice for the value of the desired elevation of the torso $y_{\mathrm{tr}}^{\mathrm{d}}$. The last parameter that depends on the desired elevation of the torso $y_{\mathrm{tr}}^{\mathrm{d}}$ is given by

$$c = \frac{g_{\mathrm{a}}}{y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2} l_{\mathrm{tr}}} = 16.35\,\mathrm{s}^{-2}, \tag{5.9}$$

which appears in the definition of the confined Capture Point

$$\bar{x}_{\mathrm{CP}}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = x_{\mathrm{tr}} + \mathrm{sat}_{\gamma}(c^{-\frac{1}{2}} \dot{x}_{\mathrm{tr}}).$$

| control parameter | associated constraints | assigned value | units | first equation reference |
|---|---|---|---|---|
| $\beta$ | $0 < \beta < 1$ | 0.5 | - | (4.7) |
| $\bar{y}_{\mathrm{swf}}$ | $\bar{y}_{\mathrm{swf}} > 0$ | 0.05 | m | (4.12) |
| $\phi$ | $\phi > 0$ | 10 | - | (4.12) |
| $a$ | $a \gg 1$ | 100 | - | (4.13) |
| $\delta$ | $\frac{1}{\delta} \gg 1$ | 0.001 | - | (4.14) |
| $\omega_0$ | $\omega_0 > 0$ | $2\pi 20$ | rad/s | (4.17) |
| $\varepsilon_3$ | $\varepsilon_1 = \varepsilon_2 = \frac{1}{2}\varepsilon_3 = \frac{1}{2}\varepsilon_4 > 0$ | 8 | m/s$^2$ | (4.37) |
| $k_{\mathrm{P2}}$ | $k_{\mathrm{P2}} > 0$ | $(2\pi 1)^2$ | s$^{-2}$ | (4.37) |
| $k_{\mathrm{D2}}$ | $k_{\mathrm{D2}} > 0$ | $2(2\pi 1)$ | s$^{-1}$ | (4.37) |
| $k_{\mathrm{P3}}, k_{\mathrm{P4}}$ | $k_{\mathrm{P3}}, k_{\mathrm{P4}} > 0$ | $(2\pi 3)^2$ | s$^{-2}$ | (4.37) |
| $k_{\mathrm{D3}}, k_{\mathrm{D4}}$ | $k_{\mathrm{D3}}, k_{\mathrm{D4}} > 0$ | $2(2\pi 3)$ | s$^{-1}$ | (4.37) |
| $k_1$ | $k_1 > 0$ | $2(2\pi 1)$ | s$^{-1}$ | (4.38) |
| $\alpha$ | $\alpha > 0$ | $0.5(2\pi 1)$ | m$^{-1}$ | (4.38) |
| $v_{\mathrm{tr}}^{\mathrm{d}}$ | $v_{\mathrm{tr}}^{\mathrm{d}} > 0$ | 0.5 | m/s | (4.38) |
| $k_0$ | $k_0 \geq 0$ | $\{0, 1\}$ | - | (4.46)-(4.47) |
| $x_{\mathrm{tr}}^{\mathrm{d}}$ | $x_{\mathrm{tr}}^{\mathrm{d}} \in \mathbb{R}$ | n/a | m | (4.33) |
| $y_{\mathrm{tr}}^{\mathrm{d}}$ | $0 < y_{\mathrm{tr}}^{\mathrm{d}} - \frac{1}{2}l_{\mathrm{tr}} < l_{\mathrm{th}} + l_{\mathrm{sh}}$ | 0.8 | m | (4.33) |

Table 5.1: The specified control parameters of the closed-loop system, including the setpoint.

For more background on the parameters $\Delta_{\max}^{\mathrm{d}}$, $\gamma$, and $c$, see Section 4.1, where they are introduced. Lastly, note that the controller bounds $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4)$ are proportional to each other due to the imposed equality constraints $\varepsilon_1 = \varepsilon_2 = \frac{1}{2}\varepsilon_3 = \frac{1}{2}\varepsilon_4$. This means that the specification of the particular controller bound $\varepsilon_3 = 8\,\mathrm{m/s^2}$ automatically provides the specification of all the controller bounds.

Let us focus in particular on the specification of the controller bound $\varepsilon_3 > 0$, and on the specification of the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} > 0$. The values that are assigned to these parameters are of great significance, because together they determine whether or not the robot will fall over during walking. That is to say that for some values of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right) > \mathbf{0}$, the robot will eventually fall over, while for other values of the parameter pair, the robot will stay balanced and keep walking in a controlled manner. In fact, there are two non-trivial parameter constraints that are essential for the correct specification of $\varepsilon_3$ and $v_{\mathrm{tr}}^{\mathrm{d}}$. These two parameter constraints are not mentioned in Table 5.1. However, they have been discussed in depth in earlier Sections. One of these two parameter constraints is discussed in Section 5.1, where an analytical approximation of the maximum achievable walking speed is derived. This analytical approximation turns out to be $\sqrt{\frac{1}{4}\varepsilon_3\gamma}$, and it is used as an upper bound for the specification of the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$. As a result, the following relation between $\varepsilon_3$ and $v_{\mathrm{tr}}^{\mathrm{d}}$ is obtained:

$$v_{\mathrm{tr}}^{\mathrm{d}} \leq \sqrt{\tfrac{1}{4}\varepsilon_3\gamma} \quad \Leftrightarrow \quad \varepsilon_3 \geq 4\gamma^{-1}(v_{\mathrm{tr}}^{\mathrm{d}})^2. \tag{5.10}$$

The other parameter constraint is discussed in Section 4.4, where an optimization problem is formulated with the objective to maximize the sum of the controller bounds $\sum_{i=1}^{4}\varepsilon_i$ subject to the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$, the contact constraint $|F_{\mathrm{f},\sigma}| \leq \mu F_{\mathrm{n},\sigma}$, and the equality constraints $\varepsilon_1 = \varepsilon_2 = \frac{1}{2}\varepsilon_3 = \frac{1}{2}\varepsilon_4$. The input constraint and the contact constraint are both evaluated at the state $(\boldsymbol{x}, \rho) \in \mathcal{S}$, which is also the default initial state for the simulations of the closed-loop system. In Section 4.4 it is also explained why it is sufficient to evaluate these constraints of the optimization problem only at the initial state. The optimization is performed with the parameter values that are given in Table 5.1, which produces a maximum and a corresponding maximizer $\bar{\boldsymbol{\varepsilon}} = (\bar{\varepsilon}_1, \bar{\varepsilon}_2, \bar{\varepsilon}_3, \bar{\varepsilon}_4)$ that is given by

$$2\bar{\varepsilon}_1 = 2\bar{\varepsilon}_2 = \bar{\varepsilon}_3 = \bar{\varepsilon}_4 = 9.1508\,\mathrm{m/s^2}. \tag{5.11}$$

The so-called maximal controller bounds $\bar{\boldsymbol{\varepsilon}}$ are used as an upper bound for the specification of the controller bounds $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4)$, with in particular

$$\varepsilon_3 < \bar{\varepsilon}_3. \tag{5.12}$$

Clearly, both the specified controller bound $\varepsilon_3 = 8\,\mathrm{m/s^2}$ and the specified desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$ satisfy all parameter constraints, including the two additional parameter constraints that are discussed above.

Meeting the requirements of the two additional parameter constraints helps to ensure that the robot walks in a controlled manner. In order to actually achieve "controlled walking", the robot must not stumble nor slip during walking. Hereby, note that the robot will stumble if the swing foot fails to keep up with the torso during walking. This occurs if the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$ is too large and/or the controller bound $\varepsilon_3$ is too small, according to (5.10). In addition, the robot will slip if the contact constraint is violated but the normal contact force $F_{\mathrm{n},\sigma}$ remains positive. This can occur if the controller bound $\varepsilon_3$ is set too large.

Now, consider the region in the two-dimensional space of the parameter pair $\big(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\big) \in \mathbb{R}_+^2$, for which the robot exhibits controlled walking. Let us refer to this region as the region of controlled walking. An approximation of this region is obtained by using the parameter constraints that are imposed on $\varepsilon_3$ and $v_{\mathrm{tr}}^{\mathrm{d}}$. Indeed, together these parameter constraints enclose a region in the two-dimensional space $\big(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\big) \in \mathbb{R}_+^2$. The interior of this region is given by

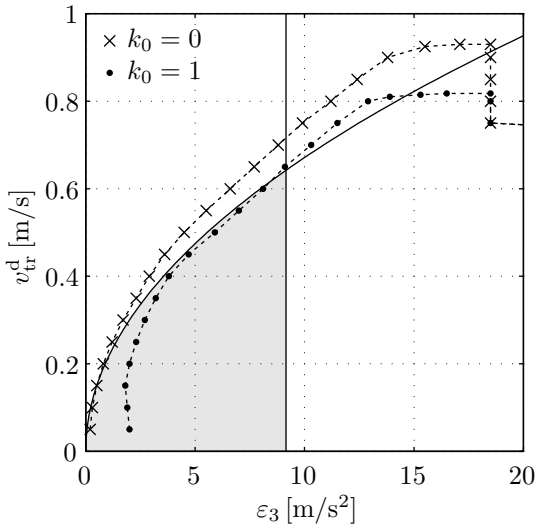
$$\mathcal{W} = \big\{ \big(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\big) \in \mathbb{R}^2 : 4\gamma^{-1}(v_{\mathrm{tr}}^{\mathrm{d}})^2 < \varepsilon_3 < \bar{\varepsilon}_3,\, v_{\mathrm{tr}}^{\mathrm{d}} > 0 \big\}. \tag{5.13}$$

Note that the set $\mathcal{W}$ represents only an approximation of the region of controlled walking. In addition, it is explained in Section 4.4 that the optimization problem (4.57) can produce conservative solutions for $\bar{\boldsymbol{\varepsilon}} = (\bar{\varepsilon}_1, \bar{\varepsilon}_2, \bar{\varepsilon}_3, \bar{\varepsilon}_4)$. As a result of this, the set (5.13) can give a conservative approximation of the region of controlled walking.

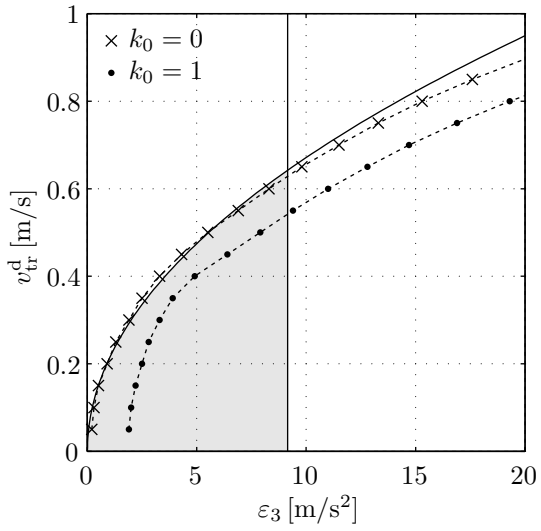
In Figure 5.1, the set $\mathcal{W}$ is visualized by the shaded region in the two-dimensional space of the parameter pair $\big(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\big) \in \mathbb{R}_+^2$. In the figure this analytical approximation of the region of controlled walking is compared to the true region of controlled walking. The comparison is made with respect to the walking direction, the PD control gains, and the two-valued gain $k_0 \in \{0, 1\}$. There are two sets of PD control gains that are considered: the set of PD control gains that is presented in Table 5.1, and a set of PD control gains for which the values are amplified, such that all PD controllers provide critical damping with a natural frequency $\omega_{\mathrm{n}} = 2\pi 5\,\mathrm{rad/s}$. This leads to a total number of $2^3 = 8$ different regions of controlled walking that are shown in Figure 5.1. Notice that (5.13) is only affected by the specification of the parameters $\gamma$ and $\bar{\varepsilon}_3$. Therefore, it is the same for all the cases that are shown in the figure. The boundary of the region of controlled walking is determined numerically by whether or not the robot falls over and/or violates the contact constraint $|F_{\mathrm{f},\sigma}| \leq \mu F_{\mathrm{n},\sigma}$ in simulations of the closed-loop system for different values of the parameter pair $\big(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\big)$, whereby the robot is instructed to walk a few meters from standstill to the desired location. A violation of the contact constraint often manifests itself in the form of sliding contact rather than loss of contact with the ground. The simulations are performed with the parameter values that are given in Table 5.1, except for the PD control gains, as mentioned above. In all these simulations, the initial state of the robot is set to (4.56), which is the same state that is used for the maximization of the controller bounds in order to obtain $\bar{\varepsilon}_3$, as described in Section 4.4. Thus, the results that are shown in Figure 5.1 are for one particular initial state only.

Notice that the shape of the region of controlled walking changes significantly as the PD control gains are increased. First of all, the region of controlled walking is smaller with the increased PD control gains. The difference between the regions for both values of $k_0 \in \{0, 1\}$ almost disappears as well, especially for $v_{\mathrm{tr}}^{\mathrm{d}} > 0.4\,\mathrm{m/s}$ and $\varepsilon_3 > 4\,\mathrm{m/s^2}$. Nevertheless, the region of controlled walking seems to be shifted downward slightly for $k_0 = 1$. Also notice that for the current specification of the control parameters, a desired walking speed of $v_{\mathrm{tr}}^{\mathrm{d}} = 1\,\mathrm{m/s}$ can only be achieved with the smallest PD control gains and only in the case of backward walking. Lastly, observe that $\mathcal{W}$ is indeed conservative with respect to $\bar{\varepsilon}_3$, as suggested earlier.

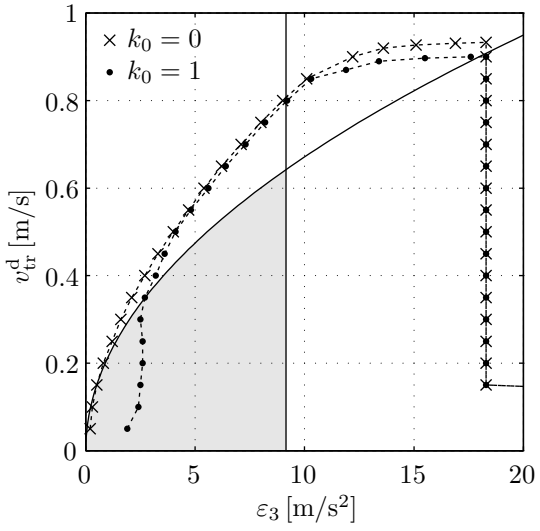
In Section 4.3, it is explained that the stabilized origin of the output error dynamics is an equilibrium point that corresponds to a connected set of infinitely many equilibrium points in the state space of the closed-loop system. This so-called equilibrium set is asymptotically stable regardless of the specification of the control parameters. Thus, it is asymptotically stable even for a specification of the control



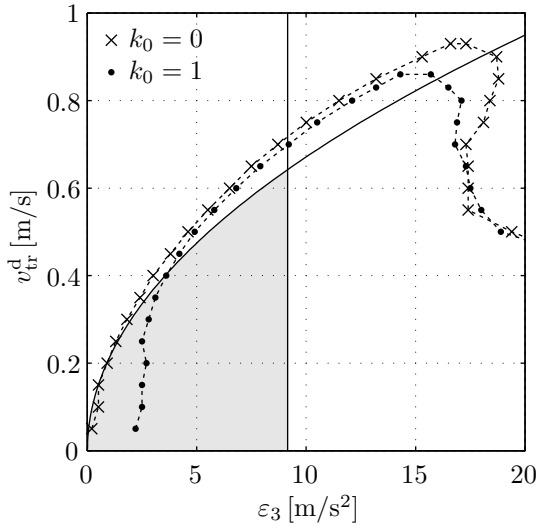

(a) Forward walking, with original PD gains.




(b) Backward walking, with original PD gains.




(c) Forward walking, with increased PD gains.




(d) Backward walking, with increased PD gains.

Figure 5.1: The region of controlled walking in the two-dimensional space of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right) \in \mathbb{R}_+^2$, whereby all other parameters have been set already according to the specification that is given in Table 5.1. The true region of controlled walking is determined numerically, and the results are shown for both forward walking and backward walking, as well as for two different sets of PD control gains. The analytically obtained approximation of the region of controlled walking $\mathcal{W}$ is shaded gray, while the true region of controlled walking is outlined by data points that are interconnected by dotted line segments. For reference, the boundaries of the set $\mathcal{W}$ are extended further out than the region itself. The numerical results are obtained by means of simulations of the closed-loop system, where the initial state $(\boldsymbol{x}(0), \rho(0)) \in \mathcal{S}$, and the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = \pm 3\,\mathrm{m}$, depending on the walking direction. The plotted data points represent points on the boundary of the region of controlled walking, i.e. the boundary between failed and successful attempts at walking to the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. The data points that are marked with a cross are obtained with simulations whereby $k_0 = 0$, and the data points that are marked with a dot are obtained with simulations whereby $k_0 = 1$. Besides the walking direction, it is also shown how the region of controlled walking changes as a result of increased PD control gains. The use of the original PD gains results in a region of controlled walking that is shown in the two sub-figures at the top. For comparison, the original PD gains are amplified such that all PD controllers provide critical damping with a natural frequency $\omega_{\mathrm{n}} = 2\pi 5\,\mathrm{rad/s}$. The use of these increased PD gains results in a region of controlled walking that is shown in the two sub-figures at the bottom.

parameters that is outside of the region of controlled walking, which means that the robot can fall over despite the unconditional stability of the equilibrium set. It turns out that the region of controlled walking actually relates to the size of the region of attraction of the equilibrium set. Moreover, controlled walking is achieved if and only if the initial state of the robot is in the region of attraction, which forces the robot to walk to the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. Note that the region of attraction refers to a region in the state space of the closed-loop system, while the region of controlled walking refers to a region in the two-dimensional space of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right) \in \mathbb{R}_+^2$.

In successful simulations of the closed-loop system, it is observed that the robot will start from a standstill at the initial state, it will then quickly reach a steady-state walking gait, and finally it will gently slow down to approach the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. In these successful simulations, the walking distance is irrelevant because of the ability of the robot to quickly reach a steady-state walking gait, which it can maintain indefinitely. Consequently, if the specification of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right)$ is inside the region of controlled walking, then the region of attraction of the equilibrium set will be unbounded in the sense that the robot can walk to the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$ from arbitrarily far away. Conversely, if the specification of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right)$ is not in the region of controlled walking, then the region of attraction of the equilibrium set will be bounded. Namely, for such a specification, the robot will slip or stumble after a finite number of steps. This usually happens within the first few steps of walking from the initial state. Therefore, the robot must start sufficiently close to the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$ in order to be able to get to the desired location without falling over. Clearly, the boundary of the region of controlled walking indicates a qualitative change in the size of the region of attraction of the equilibrium set. Moreover, a specification of the parameter pair $\left(\varepsilon_3, v_{\mathrm{tr}}^{\mathrm{d}}\right)$ inside the region of controlled walking ensures that the given initial state of the robot is in the region of attraction.

## 5.3 The behavior of the closed-loop system

Now that the parameters have been specified, it is finally possible to perform simulations of the closed-loop system. These simulations can be used to analyze the behavior of the closed-loop system, but only for the given specification of the control parameters. Keep in mind that the closed-loop behavior can change drastically depending on the specification of the parameters, as is highlighted in the previous section by the region of controlled walking. At this point, a simulation of the closed-loop system can be performed with a given initial state $(\boldsymbol{x}(0), \rho(0))$ together with an initial mode $\sigma(0)$, and for a given setpoint, which is represented by the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. For all simulations, the state of the tame differentiator (4.17) is initialized according to $\rho(0) = x_{\mathrm{tr}}(0) - \omega_0^{-1}\dot{x}_{\mathrm{tr}}(0)$, where $x_{\mathrm{tr}}(0)$ and $\dot{x}_{\mathrm{tr}}(0)$ are elements of the original state vector $\boldsymbol{x}(0)$. As a result, the initial output of the tame differentiator $\psi(0) = \dot{x}_{\mathrm{tr}}(0)$. Recall that the original state vector is given by $\boldsymbol{x}(t) = (\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))$, where $\boldsymbol{q}(t)$ is the vector of generalized coordinates and $\dot{\boldsymbol{q}}(t)$ is the vector of generalized velocities. This state vector $\boldsymbol{x}(t)$ has been extended to $(\boldsymbol{x}(t), \rho(t))$ as a result of the dynamic state feedback control law (4.30)-(4.31), whereby $\rho(t)$ is the additional state variable. In this section, only simulations of forward walking are performed, where the initial location of the torso $x_{\mathrm{tr}}(0)$ is situated to the left of the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$, i.e. $x_{\mathrm{tr}}(0) < x_{\mathrm{tr}}^{\mathrm{d}}$. Forward walking is the natural direction of walking, and it is also the easiest one to visualize. However, the robot can just as well walk backwards. As a matter of fact, the robot walks more gently in the case of backward walking, which is evident from the increased region of controlled walking, as shown in Figure 5.1. Moreover, the robot is more energy-efficient in the case of backward walking, see Section 5.4. The combination of both these facts leads us to believe that the robot's natural direction of walking is actually backward walking. Note that, due to the symmetries of the robot and the symmetries of the desired walking gait, backward walking is equivalent to forward walking if the orientation of the knee joints is reversed. This is actually common among species of birds.

The results from a number of different simulations of the closed-loop system are presented in Figure 5.2. Hereby, the first half of the simulations is shown in the sub-figures on the left-hand side, while the second half of the simulations is shown in the sub-figures on the right-hand side. In addition, the top two sub-figures show the results for the case that $k_0 = 0$, and the bottom two sub-figures show the results for the case that $k_0 = 1$. The simulations are presented in a way that resembles a phase portrait, where the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(t)$ is plotted against the location of the torso $x_{\mathrm{tr}}(t)$. Notice that the location of the torso $x_{\mathrm{tr}}(t)$ is measured relative to the location of the stance foot $x_{\mathrm{stf}}(t)$, which is

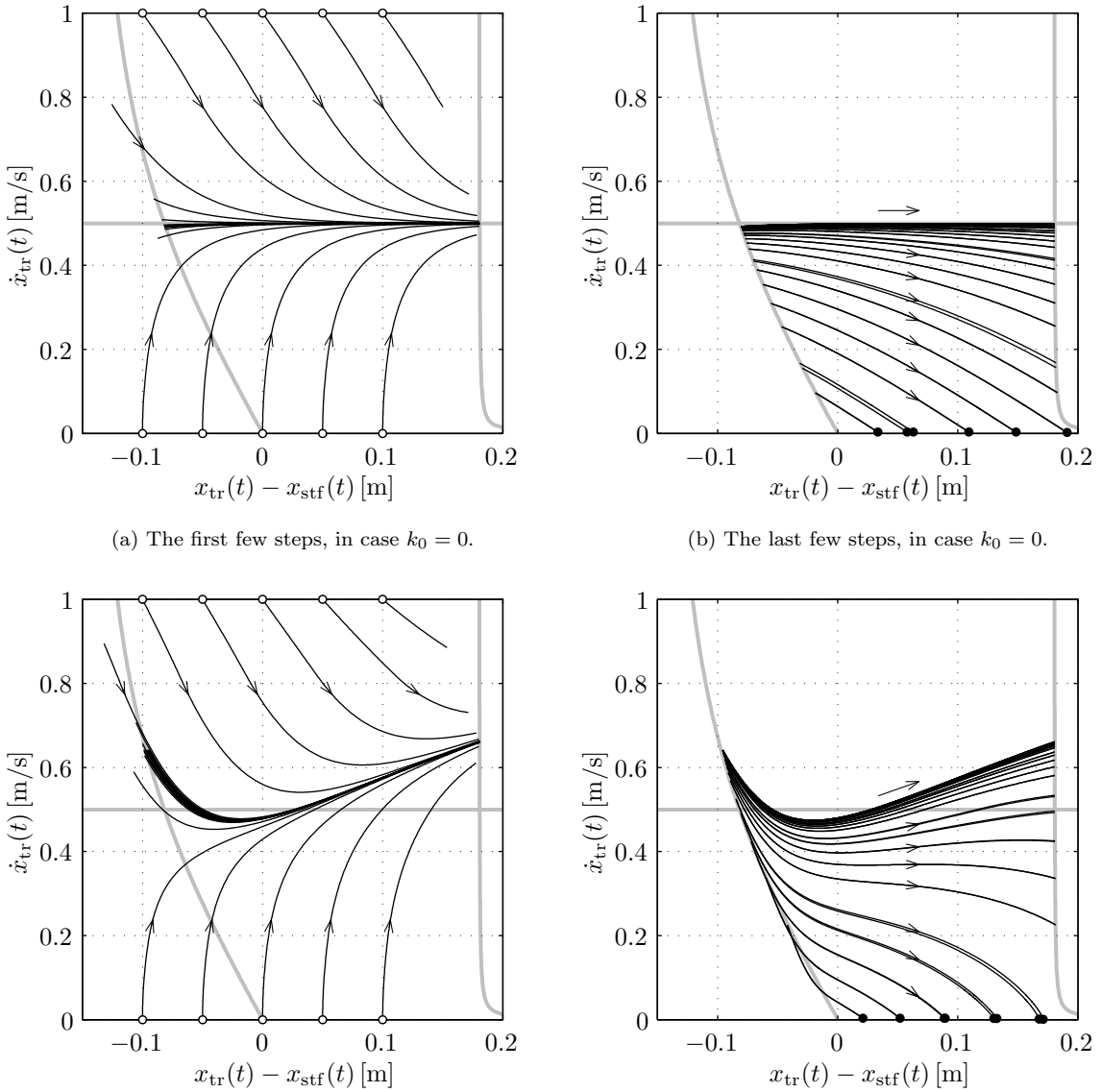
constant between walking steps. The dimension of the closed-loop system is simply too large to show everything in a single figure. Fortunately, the two state variables $x_{\mathrm{tr}}(t)$ and $\dot{x}_{\mathrm{tr}}(t)$ already provide a lot of important information regarding the walking behavior of the closed-loop system. In the latter part of this section, other important aspects of bipedal walking are presented for one of the simulations that is shown in Figure 5.2.

The initial states, which are indicated by the white dots in the sub-figures on the left, are specified such that the initial position of the torso is given by $(x_{\mathrm{tr}}(0), y_{\mathrm{tr}}(0)) = \left(0, y_{\mathrm{tr}}^{\mathrm{d}}\right)$ and the initial velocity of the torso is given by $(\dot{x}_{\mathrm{tr}}(0), \dot{y}_{\mathrm{tr}}(0)) \in \{(0,0), (1,0)\}$ m/s. For all initial states, the feet are placed on the ground on opposite sides of the torso with an equal distance to the torso. Recall that the distance between the feet is called the step size, which is denoted by $\Delta(t)$. For the simulations that are shown in Figure 5.2, the following three initial step sizes are used: $\Delta(0) \in \{0, 0.1, 0.2\}$ m. Note that the initial mode $\sigma(0) \in \{\mathrm{lfp}, \mathrm{rfp}\}$ determines which foot is actually used as the stance foot during the first stride of a simulation. The other foot will lift off the ground to become the swing foot. In all simulations, the initial velocity of both feet is set to zero. As a result, the robot is completely stationary at the start of a simulation, except for the simulations where the initial velocity of the torso $\dot{x}_{\mathrm{tr}}(0) = 1\,\mathrm{m/s}$. This initial non-zero velocity of the torso can be regarded as an initial disturbance, i.e. a push forward from standstill. The simulations show the response of the closed-loop system to these push disturbances. It is clear by looking at the figure that the robot can attenuate these particular disturbances within one or two walking steps.

The desired location of the torso $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$, while the initial location of the torso $x_{\mathrm{tr}}(0) = 0\,\mathrm{m}$. This may seem like a short distance to walk, but remember that the gait is designed for a maximum step size of $2\gamma = 0.3606\,\mathrm{m}$. Moreover, this maximum step size is only desired for a walking speed of $\dot{x}_{\mathrm{tr}} \gtrsim 1\,\mathrm{m/s}$. For a velocity of $\dot{x}_{\mathrm{tr}} = v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$, the desired step size is roughly $0.25\,\mathrm{m}$. Consequently, the robot will step 7 to 8 times within the walking distance of $2\,\mathrm{m}$. This is enough to observe both the transient walking behavior as well as the steady-state walking behavior of the closed-loop system. Notice that in the case of $k_0 = 0$, the steady-state walking speed is approximately equal to the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$. However, in the case of $k_0 = 1$, the steady-state walking speed forms a velocity profile of which the average velocity is slightly higher than the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$. To some extent, the velocity profile resembles the velocity profile of an unstable inverted pendulum, whereby the velocity increases as the pendulum leans further towards one side. Indeed, in the case of $k_0 = 1$, the robot tries to emulate the natural inverted pendulum-like walking motion that is observed in humans, which is thought to be a way to reduce the energy consumption of walking.

Overall, the walking gait of the closed-loop system conforms well to the desired walking gait, which is outlined by the gray curves in Figure 5.2, even though the PD control gains are set relatively low. The black dots in the sub-figures on the right indicate the endpoints of the simulations. Hereby, notice that the torso always comes out further than the stance foot as the robot approaches a standstill at the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. This is a result of the way that the gait is designed. Namely, the robot can stand still on the stance foot even if the stance foot is not directly underneath the torso. The robot is able to do this because it is fully actuated. Still, it requires a constant non-zero input torque from the actuators to maintain such a posture. Due to the asymptotic stability of the equilibrium set, a successful simulation of the closed-loop system is terminated prematurely once it holds that $\left|x_{\mathrm{tr}}(t) - x_{\mathrm{tr}}^{\mathrm{d}}\right| \leq 0.01\,\mathrm{m}$ and $\|\dot{\boldsymbol{q}}(t)\| \leq 0.01$. For unsuccessful simulations, the robot will either slip and thus violate the contact constraint $|F_{\mathrm{f},\sigma}(t)| \leq \mu F_{\mathrm{n},\sigma}(t)$, or the robot will stumble and thereby fully extend the stance leg. Lastly, notice in Figure 5.2 that there is indeed a set of equilibrium points, because all the black dots are endpoints of successful simulations, whereby the robot is at the very same desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. The difference is the final location of the stance foot. Also, notice that there seems to be fewer black dots than white dots. Apparently, the initial horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(0) \in \{0, 1\}$ m/s has not much effect on the final state of the robot, at least for the simulations that are shown in Figure 5.2.

Now, let us focus on a single simulation of the closed-loop system for $k_0 = 1$, where the initial state $(\boldsymbol{x}(0), \rho(0)) \in \mathcal{S}$ and the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$. The singleton set $\mathcal{S}$ is given by (4.56), which corresponds to the initial state of one of the simulations that is shown in Figure 5.2. Hereby, the initial location of the torso $x_{\mathrm{tr}}(0) = 0\,\mathrm{m}$, the initial horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(0) = 0\,\mathrm{m/s}$, and the initial step size $\Delta(0) = 0\,\mathrm{m}$. Consequently, the specified initial state of the tame differentiator $\rho(0) = x_{\mathrm{tr}}(0) - \omega_0^{-1}\dot{x}_{\mathrm{tr}}(0) = 0\,\mathrm{m}$, and the initial output of the tame differentiator $\psi(0) = \dot{x}_{\mathrm{tr}}(0) = 0\,\mathrm{m/s}$. Different aspects of this simulation are shown in multiple figures: from Figure 5.3 through to Figure 5.6.




(a) The first few steps, in case $k_0 = 0$.

(b) The last few steps, in case $k_0 = 0$.

(c) The first few steps, in case $k_0 = 1$.

(d) The last few steps, in case $k_0 = 1$.

Figure 5.2: Several simulations of the closed-loop system for different initial states. The solutions $(\boldsymbol{x}(t), \rho(t))$ are presented here in the form of phase portraits, where the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(t)$ is plotted against the location of the torso $x_{\mathrm{tr}}(t)$. These two state variables are chosen, because they provide a lot of information about the walking gait. Notice that the location of the torso $x_{\mathrm{tr}}(t)$ is measured relative to the location of the stance foot $x_{\mathrm{stf}}(t)$. This has the effect of stacking the trajectories of the walking steps on top of each other, and thereby giving the appearance of a phase portrait. Also, notice that the simulations are split in half. The first half is shown in the sub-figures on the left, while the second half is shown in the sub-figures on the right. In addition, the two sub-figures at the top show simulations for $k_0 = 0$, and the two sub-figures at the bottom show simulations for $k_0 = 1$. A simulation is terminated if the robot is within 1 cm of the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$ and close enough to a standstill, which is specified by $\|\dot{\boldsymbol{q}}(t)\| \leq 0.01$. The starting point of every simulation is marked by a white dot, and the endpoint is marked by a black dot. In each sub-figure, the three gray curves indicate the desired walking speed $\dot{x}_{\mathrm{tr}} = v_{\mathrm{tr}}^{\mathrm{d}}$, the desired foot placement $\tilde{\zeta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}}) = 0$, and the subsequent stance after the desired foot placement $x_{\mathrm{tr}} - x_{\mathrm{stf}} = -\tilde{\eta}(x_{\mathrm{tr}}, \dot{x}_{\mathrm{tr}})$.

In Figure 5.3 a visualization of the walking gait of the robot is given for the single simulation that is considered above. The plot at the top of the figure shows the velocity profile, just like in Figure 5.2, but this time fully unwrapped. Notice that the location of the torso $x_{\mathrm{tr}}(t)$ is a monotonically increasing function of time, which is due to the non-negativity of the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(t)$. The output of the tame differentiator $\psi(t)$ is included in the plot for reference. Recall that the tame differentiator is used in the feedback linearization to obtain a filtered version of the velocity of the torso $\dot{x}_{\mathrm{tr}}(t)$. In the figure, it is clear that $\psi(t)$ tracks $\dot{x}_{\mathrm{tr}}(t)$ quite well, which is mainly due to the relatively large value that is assigned to the parameter $\omega_0$ of the tame differentiator. Notice that the robot reaches steady-state walking here within just one walking step. On the other hand, it takes two steps and a relatively long time to go from steady-state walking to a standstill at the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$. The fast start is a result of the relatively large deviation between the initial state and the desired walking gait, which the controller quickly tries to eliminate as if it is a disturbance. The gentle slowdown near the end of the simulation is due to the proximity to the asymptotically stable equilibrium set, which is at the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$, in combination with the fact that the control parameters are specified such that the solution of the closed-loop system is overdamped. Indeed, there is no overshoot present in the solution that is shown in Figure 5.3, which is also true for the solutions that are shown in Figure 5.2.

The plot in the middle of Figure 5.3 depicts the walking gait of the robot. It includes the trajectories of all joints as they are traced out in space, whereby a full stick figure is drawn every 0.6 s to illustrate the posture of the robot at these moments in time. Dots are placed along the joint trajectories every 0.1 s to indicate the relative speed of the different parts of the robot during walking. Also, the left leg of the robot is colored gray to clarify that the robot is facing to the right and walking forward. Unfortunately, the visualization of the gait does not really capture the inverted pendulum-like walking behavior, despite its presence as shown by the oscillations of the horizontal velocity of the torso $\dot{x}_{\mathrm{tr}}(t)$. The related oscillations of the vertical velocity of the torso $\dot{y}_{\mathrm{tr}}(t)$ are just as hard to spot in the gait visualization itself.

The plot at the bottom of Figure 5.3 shows the location of the stance foot $x_{\mathrm{stf}}(t)$ in time. The location of the torso $x_{\mathrm{tr}}(t)$ is added as a dashed curve to this plot for reference. Incidentally, the trajectory of the torso shows what the trajectory of the robot actually looks like compared to the trajectory that is sketched in Figure 1.1. Notably, there is no overshoot here with respect to the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$, which means that the solution is indeed overdamped. The robot walks the prescribed distance of 2 m in under 6 s, at least according to the criteria for the termination of successful simulations, as discussed above. In actuality, the robot never reaches the desired location in finite time, because the equilibrium set is only asymptotically stable, and not finite-time stable, for example. Notice that the time period of 6 s includes both transient phases of walking in addition to the steady-state walking phase. From the plot it is clear that roughly 2 s of this time period is taken up by the last transient phase, in which the robot slows down to approach the desired location $x_{\mathrm{tr}}^{\mathrm{d}}$. Lastly, observe that the distance between two consecutive locations of the stance foot is on average slightly more than 0.25 m. This is consistent with the desired step size mentioned earlier.

For the same simulation of the closed-loop system, the position of the swing foot is given in Figure 5.4, whereby the desired position of the swing foot is shown in gray. Notice that there is no overshoot in the tracking of the desired position of the swing foot, which means that the solution is either critically damped or overdamped. In this case, the PD control gains are set to provide critical damping for the swing foot, as discussed in Section 5.2. Notice that the desired position of the swing foot differs from the actual position of the swing foot at the start of every stride. Also, notice that the desired position of the swing foot does not go to zero during the last stride, where the robot asymptotically approaches the desired location. If the robot is at rest, then the desired location of the swing foot $x_{\mathrm{swf}}^{\mathrm{d}}(t)$ is designed to coincide with the location of the torso $x_{\mathrm{tr}}(t)$. The torso ends up at the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$, which is 0.09 m to the right of the stance foot in this case. The desired elevation of the swing foot $y_{\mathrm{swf}}^{\mathrm{d}}(t)$ only goes to zero in case of a foot placement, which only takes place if it holds that $x_{\mathrm{tr}}(t) - x_{\mathrm{stf}}(t) \approx \gamma = 0.1803\,\mathrm{m}$. Thus, the location of the torso $x_{\mathrm{tr}}(t)$ is half way to a foot placement here. This is why the swing foot is kept well above the ground at the end of the simulation. By the way, the desired maximum foot clearance is given by $\bar{y}_{\mathrm{swf}} = 0.05\,\mathrm{m}$, which can be seen in the figure as well. Lastly, note that the swing foot is stationary on the ground at the start of every stride, because the swing foot and the stance foot change roles in between every stride.

In Figure 5.5, the torques are shown that are applied to every joint of the robot during the course of the same simulation as before. Mainly notice here that the applied torques stay within the torque limits

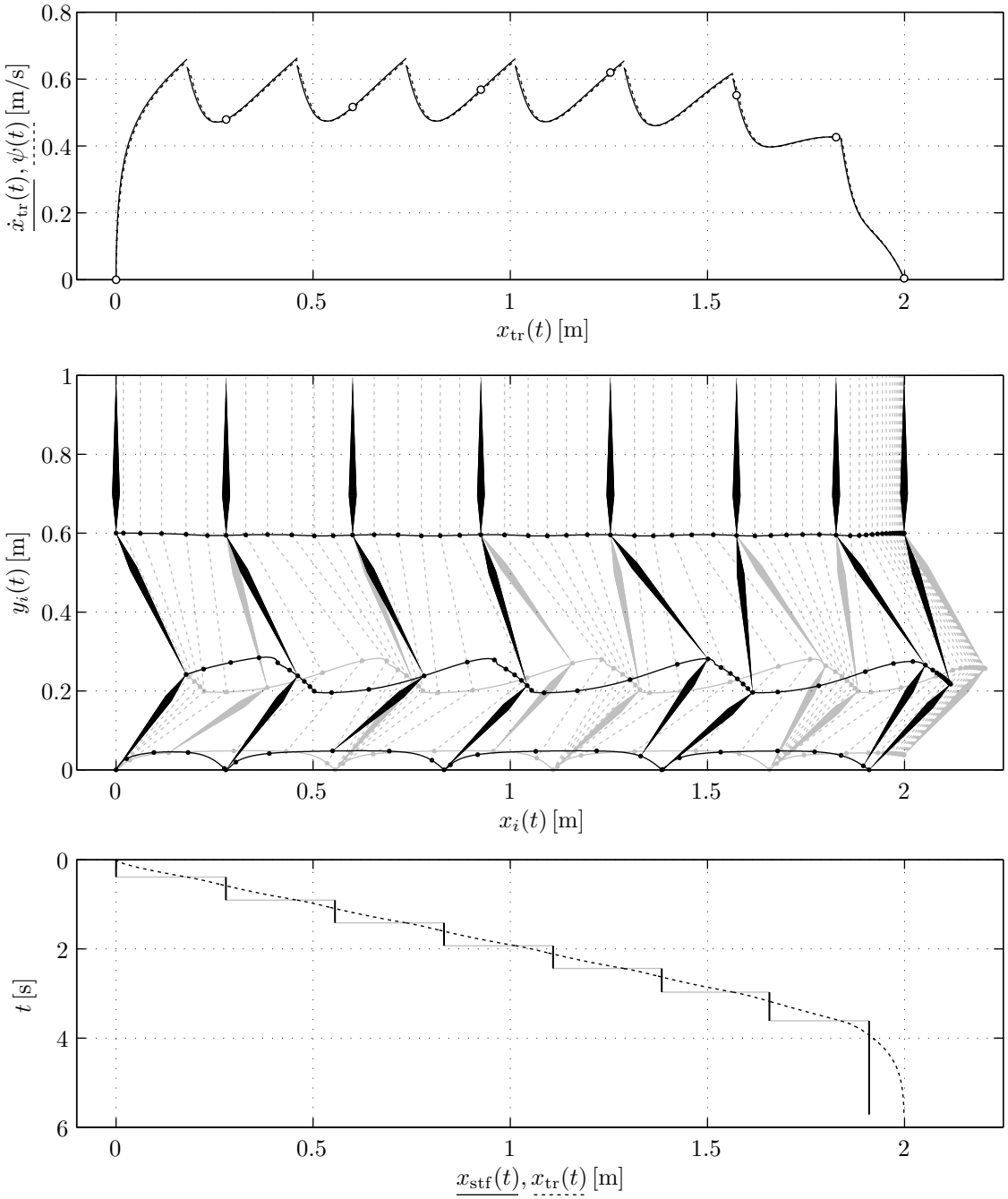



Figure 5.3: A visualization of the walking gait of the closed-loop system for $k_0 = 1$, where the initial state $(\boldsymbol{x}(0), \rho(0)) \in \mathcal{S}$ and the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$. The velocity profile is given in a plot above the gait visualization, and the location of the stance foot $x_{\mathrm{stf}}(t)$ is given as a function of time in a plot below the gait visualization. On the axes of the gait visualization itself, the coordinates $(x_i(t), y_i(t))$ denote the position coordinates of the joints of the robot, where $i \in \mathcal{J} = \{\mathrm{lh}, \mathrm{lk}, \mathrm{lf}, \mathrm{rh}, \mathrm{rk}, \mathrm{rf}\}$. The trajectories of the joints of the right leg are depicted in black, while the trajectories of the joints of the left leg are depicted in gray. Dots are placed along the joint trajectories every 0.1 s, and the gray dashed line-segments show the configuration of the left leg and the torso at those moments in time. A complete stick figure is drawn every 0.6 s, but only if it does not obstruct the view of the previous stick figure. This happens near the end of the simulation, where the robot gently slows down and asymptotically approaches a standstill at the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = 2\,\mathrm{m}$. So, only the final configuration of the robot is visualized with a stick figure near the end of the simulation. The location of each stick figure is also indicated by the white dots in the velocity profile above the gait visualization.

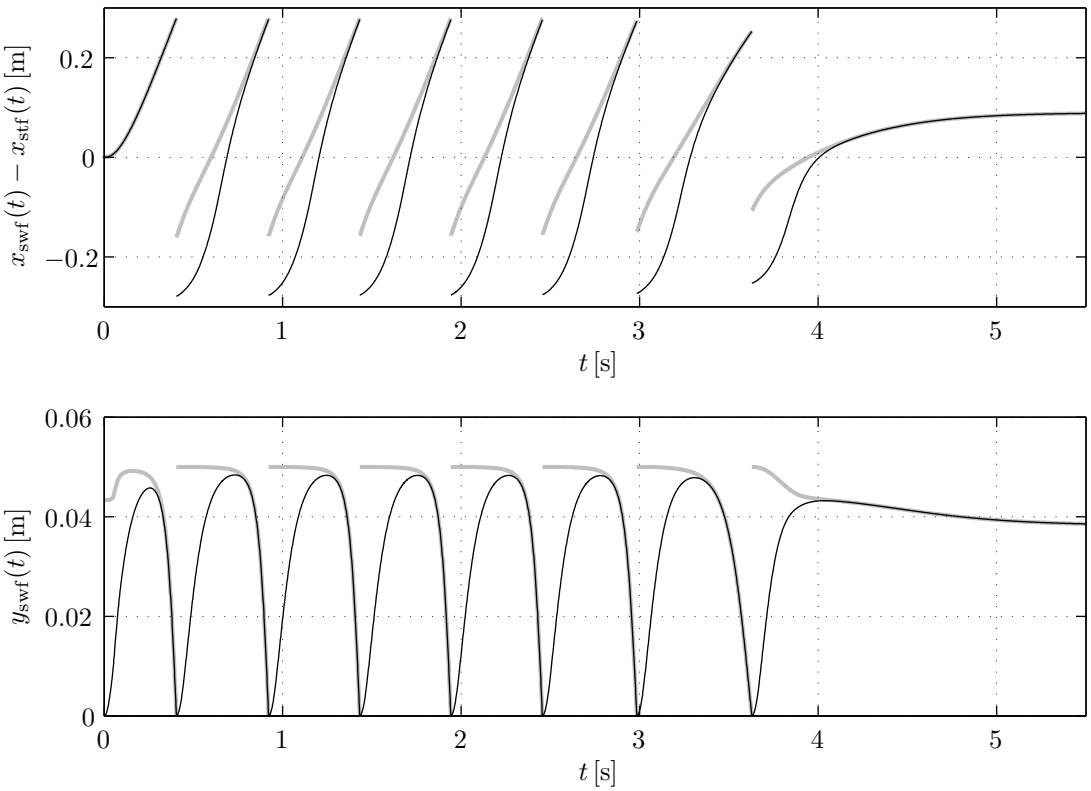



Figure 5.4: The position of the swing foot $(x_{\mathrm{swf}}(t), y_{\mathrm{swf}}(t))$ as a function of time. In the plot at the top, the location of the swing foot $x_{\mathrm{swf}}(t)$ is shown relative to the location of the stance foot $x_{\mathrm{stf}}(t)$. In the plot at the bottom, the elevation of the swing foot $y_{\mathrm{swf}}(t)$ is shown. For reference, the desired position of the swing foot $\big(x^{\mathrm{d}}_{\mathrm{swf}}(t), y^{\mathrm{d}}_{\mathrm{swf}}(t)\big)$ is added in gray.

of $L_{\mathrm{h}}, L_{\mathrm{k}} = 80\,\mathrm{Nm}$, as specified in Table 3.3. Recall from the previous chapter that the optimization of the controller bounds (4.57) is subject to the input constraint $-\boldsymbol{L} \leq \boldsymbol{u} \leq \boldsymbol{L}$, where $\boldsymbol{L} = (L_{\mathrm{h}}, L_{\mathrm{k}}, L_{\mathrm{h}}, L_{\mathrm{k}})$. Thus, it is the specification of the controller bounds $\boldsymbol{\varepsilon} < \bar{\boldsymbol{\varepsilon}}$ that actually ensures that the applied torques stay within the torque limits. In addition, notice in the figure that there are torque spikes at the start of every stride. These torque spikes enable the swing foot to get off the ground quickly, and they occur because of the error in the elevation of the swing foot at the start of every stride, as seen in Figure 5.4.

Lastly, Figure 5.6 shows the contact forces $(F_{\mathrm{f},\sigma}(t), F_{\mathrm{n},\sigma}(t))$ that exist between the stance foot and the ground. Clearly, the contact constraint $|F_{\mathrm{f},\sigma}(t)| \leq \mu F_{\mathrm{n},\sigma}(t)$ is not violated during walking. In fact, only the initial state comes close to violating the contact constraint, as indicated by the white dot in the figure. This observation validates the decision to evaluate the contact constraint and the input constraint only at the initial state $\mathcal{S}$ in the optimization of the controller bounds, as discussed in Section 4.4. The sign of the friction contact force $F_{\mathrm{f},\sigma}(t)$ dictates whether the robot pushes off against the ground to propel itself forward or to slow itself down. In Figure 5.6, the main darkened shape corresponds to the steady-state walking phase of the simulation, which contains all but the first and the last two strides. The trajectories that stick out to the left and right of this shape correspond to the transient phases, where the robot accelerates to start walking and decelerates to stop walking. Notice that these transient phases are accompanied by larger friction forces than the friction forces that are seen during the steady-state walking phase. Also, notice that the largest friction forces are achieved at the start of the simulation, but also at the start of every stride. This is evidently the moment where the robot pushes most forcefully against the ground, which is consistent with the observed spikes in the applied torques, as seen in Figure 5.5. Finally, observe that the friction force goes to zero at the end of the simulation, which is indicated by the black dot in the center of Figure 5.6. At this point, the horizontal velocity of the robot does not change anymore, which is good because the robot must be at rest once it reaches the desired location.

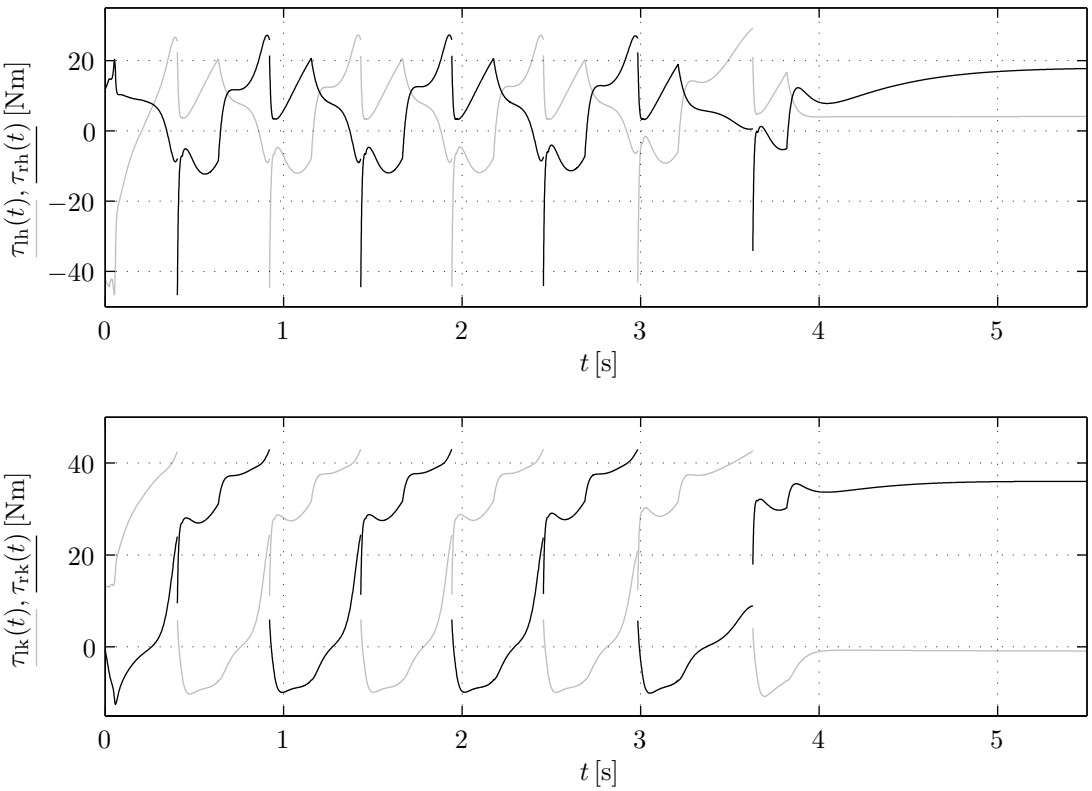



Figure 5.5: The applied torques $\boldsymbol{\tau}(t)$ as a function of time. The torques that are applied to the hip joints are shown in the plot at the top, and the torques that are applied to the knee joints are shown in the plot at the bottom. The color gray is used for applied torques to the left leg, and the color black is used for applied torques to the right leg.

## 5.4 The mechanical cost of transport

In this section, the energy efficiency of the walking gait of the closed-loop system is discussed. The main question here is whether the energy efficiency is improved by the addition of the accelerations $\bar{a}_1, \bar{a}_2$ to the control design. Hereby, recall that $k_0$ is the gain of these accelerations, as discussed in Section 4.3. So, for the binary specification of the gain $k_0 \in \{0, 1\}$, the accelerations $\bar{a}_1, \bar{a}_2$ can be switched on and off. If the gain $k_0 = 0$, then the accelerations do not play a role in the behavior of the closed-loop system. On the other hand, if the gain $k_0 = 1$, then the robot exhibits an inverted pendulum-like walking behavior that is shown in Section 5.3. In this section, the gain $k_0$ is also increased beyond $k_0 = 1$, which amplifies the inverted pendulum-like walking behavior.

The energy efficiency of the walking gait can be quantified by the so-called cost of transport [7]. The cost of transport (COT) is a dimensionless measure of the amount of energy that is used by the system for its displacement. It provides a metric to compare the energy efficiency of different modes of transport and different vehicles, regardless of the mass of the vehicle and the distance traveled. A bipedal robot with a low COT is considered more energy efficient than a bipedal robot with a high COT. Here, the cost of transport is calculated only based on the amount of mechanical work that is done by the system to achieve the displacement. This is referred to as the mechanical cost of transport. Note that the mechanical cost of transport does not take into account the losses of energy conversions. Therefore, the general cost of transport is always higher than the mechanical cost of transport, for a given robot.

The COT is determined by means of simulations of the closed-loop system, where the initial state $(\boldsymbol{x}(0), \rho(0)) \in \mathcal{S}$. Notice that the start-time of a simulation is set to $t = 0\,\mathrm{s}$. By convention, let the end-time of a simulation be denoted by $t = T > 0\,\mathrm{s}$. Observe that the total work done by the system is given by the integral $\int_0^T \dot{\boldsymbol{q}}^\top(t)\boldsymbol{S}\boldsymbol{\tau}(t)\,\mathrm{d}t$, where $\dot{\boldsymbol{q}}(t)$ is the vector of generalized velocities and $\boldsymbol{\tau}(t)$ is the vector of applied torques. The constant matrix $\boldsymbol{S}$, which is written out in Appendix A, represents the

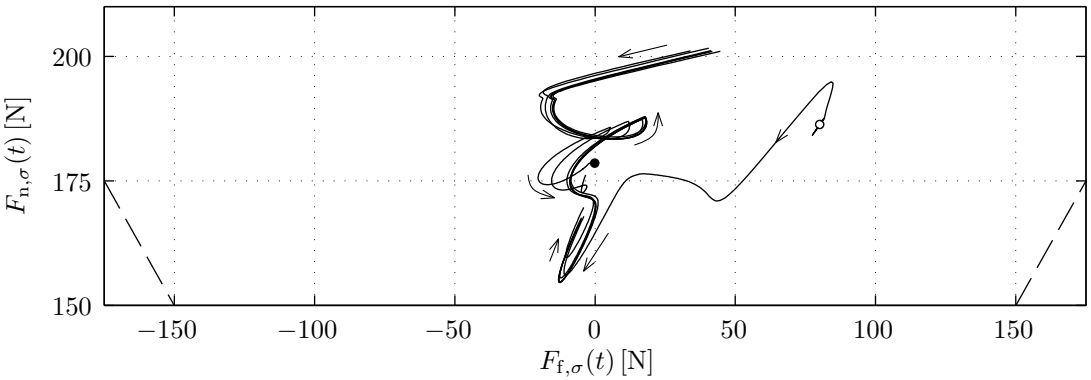



Figure 5.6: A plot of the normal contact force $F_{\mathrm{n},\sigma}(t)$ versus the friction contact force $F_{\mathrm{f},\sigma}(t)$. The arrows indicate the direction of flow, and the black dot in the center of the figure marks the endpoint of the simulation. The white dot on the right signifies the start of the simulation. Notice the two dashed lines at the edges of the figure. These lines represent the boundary of the contact constraint $|F_{\mathrm{f},\sigma}(t)| \leq \mu F_{\mathrm{n},\sigma}(t)$, where the coefficient of static friction $\mu = 1$.

mapping between the generalized coordinates of the robot and the applied torques. Consider that the mechanical cost of transport is defined as

$$E_{\mathrm{COT}} = \frac{\int_0^T \left|\dot{\boldsymbol{q}}^\top(t)\boldsymbol{S}\boldsymbol{\tau}(t)\right| \mathrm{d}t}{m_{\mathrm{robot}} g_{\mathrm{a}} \left|x_{\mathrm{tr}}(T) - x_{\mathrm{tr}}(0)\right|}, \tag{5.14}$$

where the mass of the robot $m_{\mathrm{robot}} = m_{\mathrm{tr}} + 2\,(m_{\mathrm{th}} + m_{\mathrm{sh}}) = 18.2\,\mathrm{kg}$, and the gravitational acceleration $g_{\mathrm{a}} = 9.81\,\mathrm{m/s^2}$. The distance $|x_{\mathrm{tr}}(T) - x_{\mathrm{tr}}(0)|$ represents the distance traveled by the robot during a simulation. Recall that for a successful simulation of the closed-loop system, the location of the torso $x_{\mathrm{tr}}(t) \to x_{\mathrm{tr}}^{\mathrm{d}}$ as time $t \to \infty$, where $x_{\mathrm{tr}}^{\mathrm{d}}$ is the desired location of the torso, which is also regarded as the desired location of the robot. At the moment that a successful simulation is terminated, the location of the torso will be $x_{\mathrm{tr}}(T) \approx x_{\mathrm{tr}}^{\mathrm{d}}$. In addition, notice that the mechanical cost of transport (5.14) is defined in terms of absolute power. This is done so that the numerator of (5.14) represents the cumulative amount of mechanical energy that is used by the robot over the course of the entire simulation. To clarity, the work that is done by the system is represented by the change in the mechanical energy of the system as a result of the applied torques, which can also be negative. Thus, the absolute value is used to account for both the positive and the negative changes in the mechanical energy of the system, which are both considered as energy spent for walking.

Table 5.2 shows the mechanical cost of transport for different values of the gain $k_0$, and for several different walking speeds $v_{\mathrm{tr}}^{\mathrm{d}}$. The actual COT itself is not that important here; the focus is on how the COT changes for different values of the control parameters. The COT is merely used as a way to compare the energy efficiency of the walking gait for different values of the control parameters. The desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$ is included in the analysis, because it is interesting to know how the COT changes as the robot walks the same distance at different speeds.

From Table 5.2a it is clear that in the case of forward walking, the energy efficiency of the walking gait generally improves with the addition of the accelerations $\bar{a}_1, \bar{a}_2$, whereby the gain $k_0 > 0$. This is true to a lesser extent in the case of backward walking. However, the COT does not actually decrease that much when going from $k_0 = 0$ to $k_0 > 0$. Thus, with the current control design, there is not much for the robot to gain in terms of energy efficiency by walking in an inverted pendulum-like manner. The walking gait of the robot is nearly just as energy efficient for $k_0 = 0$, especially at higher walking speeds.

The biggest difference in the COT is obtained between forward walking and backward walking, with even a factor of two in some cases. Note that backward walking is equivalent to forward walking if the orientation of the knee joints is reversed. Apparently, it is more energy efficient for the robot to walk with inverted knee joints. This configuration of the knee joints is also observed in different species of birds. Furthermore, some bipedal robots use this configuration as well, notably the robot SCHAFT

| | $k_0 = 0$ | $k_0 = 1$ | $k_0 = 2$ | $k_0 = 3$ |
|---|---|---|---|---|
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.1\,\mathrm{m/s}$ | 0.6901 | 0.4851 | 0.4238 | 0.4423 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.2\,\mathrm{m/s}$ | 0.6271 | 0.4824 | 0.4396 | 0.4759 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.3\,\mathrm{m/s}$ | 0.6117 | 0.5016 | 0.4773 | 0.5329 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.4\,\mathrm{m/s}$ | 0.6197 | 0.5435 | 0.5383 | 0.6036 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$ | 0.6491 | 0.6063 | 0.6316 | 0.7292 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.6\,\mathrm{m/s}$ | 0.6945 | 0.6568 | n/a | n/a |

(a) Forward walking.

| | $k_0 = 0$ | $k_0 = 1$ | $k_0 = 2$ | $k_0 = 3$ |
|---|---|---|---|---|
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.1\,\mathrm{m/s}$ | 0.3269 | 0.2886 | 0.3046 | 0.4956 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.2\,\mathrm{m/s}$ | 0.3196 | 0.2926 | 0.3331 | 0.5428 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.3\,\mathrm{m/s}$ | 0.3283 | 0.3141 | 0.3751 | 0.6074 |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.4\,\mathrm{m/s}$ | 0.3513 | 0.3603 | 0.4680 | n/a |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.5\,\mathrm{m/s}$ | 0.3898 | 0.4705 | n/a | n/a |
| $v_{\mathrm{tr}}^{\mathrm{d}} = 0.6\,\mathrm{m/s}$ | n/a | n/a | n/a | n/a |

(b) Backward walking.

Table 5.2: The mechanical cost of transport for different values of the gain $k_0$ and for different values of the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$. The results are shown for both forward walking and backward walking. For some combinations of $k_0$ and $v_{\mathrm{tr}}^{\mathrm{d}}$, the robot is unable to walk to the desired location without falling over. These unsuccessful simulations are listed as n/a. The mechanical cost of transport is determined by means of simulations of the closed-loop system, where the initial state $(\boldsymbol{x}(0), \rho(0)) \in \mathcal{S}$ and the desired location $x_{\mathrm{tr}}^{\mathrm{d}} = \pm 4\,\mathrm{m}$, depending on the walking direction. The walking distance is increased to 4 m to obtain a better average of the mechanical cost of transport, as it is calculated over an increased number of strides. Moreover, the increased walking distance diminishes the contribution of the two transient phases of walking and ensures that the mechanical cost of transport is mainly based on the steady-state walking phase.

from a Japanese company of the same name, and the robot Cassie from the American company Agility Robotics. In addition to a lower COT, the robot also has a larger region of controlled walking in the case of backward walking, as shown in Figure 5.1. For example, it is possible to obtain a desired walking speed of $v_{\mathrm{tr}}^{\mathrm{d}} = 1\,\mathrm{m/s}$ only in the case of backward walking. Lastly, note that in general the COT increases if the desired walking speed $v_{\mathrm{tr}}^{\mathrm{d}}$ is increased.

## 5.5 Summary

The maximum achievable walking speed is defined as the maximum speed at which the robot can walk without stumbling and falling over. An analytical approximation of the maximum achievable walking speed is derived for the given control design. Naturally, this analytical approximation is used as a constraint for the specification of the desired walking speed of the robot. It is shown that the maximum achievable walking speed depends on the controller bounds. As a result, the specification of the controller bounds and the specification of the desired walking speed are both crucial for successful walking, because they both determine if the walking speed of the robot exceeds the maximum achievable walking speed. In addition, the controller bounds themselves are limited by the maximal controller bounds. It is shown that the analytical approximation of the maximum achievable walking speed and the maximal controller bounds together enclose a region in the parameter space. This region is an approximation of the so-called region of controlled walking that represents the set of parameter values for which the robot can walk in a controlled manner to the desired location without falling over. It turns out that the approximation of the region of controlled walking is only conservative with respect to the maximal controller bounds.

The desired walking speed is set to 0.5 m/s, which is about three times slower than the walking speed that humans prefer to walk at. Unfortunately, the desired walking speed cannot be set much higher, because it is already close to the maximum achievable walking speed for the given specification of the controller bounds. The other control parameters are specified such that the step size of the robot is equal to half of the maximum possible step size. In addition, the PD control gains are set relatively low in order to obtain a smooth walking motion, but also to ensure that the effects of the added gravitational acceleration terms are visible. These acceleration terms have been added to the control design in order to obtain inverted pendulum-like walking behavior. The settings of the PD control gains are such that the solutions of the closed-loop system are either critically damped or overdamped.

After the specification of the control parameters, multiple simulations are performed to show the behavior of the closed-loop system from different initial states. In all cases, the robot is able to walk to a desired location, where it gently slows down and asymptotically approaches a standstill at the desired location. Moreover, the robot is able to walk to the desired location from standstill as well as after an initial push, which can be regarded as a disturbance. The closed-loop system is able to attenuate these disturbances within one or two strides. In general, the robot is able to reach steady-state walking within the first two strides. This is considerably faster than going from steady-state walking to a standstill at the desired location. In addition, several important aspects of the walking gait are analyzed in detail for a typical simulation of the closed-loop system that includes the inverted pendulum-like walking behavior.

Lastly, the energy efficiency of the walking gait is analyzed. This is done in terms of the cost of transport, which is a dimensionless measure of the amount of energy that is used by the system for its displacement. It turns out that the inverted pendulum-like walking behavior does not improve the energy efficiency of the walking gait as much as one would have hoped. However, it is observed that backward walking can be up to twice as energy efficient as forward walking. Hereby, note that backward walking is equivalent to forward walking with inverted knee joints. Apparently, this configuration of the knee joints results in more energy efficient walking for the given robot, and for the particular control design that is presented in this thesis. The data also shows that the walking gait is more energy efficient at lower walking speeds in both walking directions.

# Chapter 6

# Conclusions and recommendations

Here, the conclusions of this thesis are presented, and recommendations are given for future work.

## 6.1 Conclusions

A bipedal robot with point feet is considered that is constrained to the sagittal plane. The torso of the robot is also constrained so that it cannot rotate, which makes the robot fully actuated if one foot is in contact with the ground. The kinematics of the robot is described using absolute coordinates, instead of relative coordinates. This allows for a relatively straightforward derivation of the dynamics of the bipedal robot with the Newton-Euler equations of motion. It also automatically provides the kinematic constraint forces that act between the joints of the robot, including the contact forces between the stance foot and the ground. The interaction with the ground is modeled as a revolute joint, which forms a bilateral constraint with the ground, while the contact with the ground is only a unilateral constraint. To make up for this modeling discrepancy, Coulomb's model of static friction is used to assess the validity of the bilateral constraint. In addition, Coulomb's model is used to establish whether or not the stance foot is slipping on the ground.

It turns out that the controlled walking of a bipedal robot to a desired location requires more than just full actuation. Yet, the balance criteria for bipedal robots that are found in the literature focus on conditions for which a bipedal robot is fully actuated, or at least not underactuated. Essentially, these balance criteria ensure that the stance foot of a bipedal robot remains flat on the ground. Of course, it is very useful to have such criteria, because full actuation enables complete control over the motion of the robot. However, a bipedal robot that is controlled with the assumption of full actuation might still fall over if it becomes underactuated somehow. For example, the stance foot can start to rotate around an edge after an unexpected disturbance. Underactuation is indeed one of the main ways in which a bipedal robot can fall over during walking, especially if the robot is not intended to be underactuated at all. However, there are also other ways in which a bipedal robot can fall over. In this thesis it is shown that a fully actuated bipedal robot can still fall over by slipping or stumbling during walking.

A bipedal robot stands or falls by the amount of torque that is applied to the joints of the robot. For example, too much torque at the ankle joint can cause underactuation if the stance foot does not remain flat on the ground. Too much torque can also make the robot push too hard against the ground during walking such that the contact forces exceed the threshold for static friction, causing the stance foot to slip. On the other hand, not enough torque can make the robot stumble and fall over if it cannot keep up with the desired walking speed. In short, the amount of torque that is applied by the controller must be considered carefully during walking. Therefore, it makes sense to use controller bounds. A bound on the control action limits the amount of torque that is applied by the controller. If this bound is set relatively small to prevent slipping, then the desired walking speed of the robot must be set small enough as well in order to prevent stumbling. Notice that the controller bounds are primarily used to try to maintain the delicate contact with the ground, as described by Coulomb's model. Inadvertently, the use of controller bounds introduces the possibility of stumbling over.

There is a set of parameter values for both the controller bounds and the desired walking speed that results in controlled walking. This set is named the region of controlled walking. An approximation of

this set is derived based on an analytical approximation of the maximum achievable walking speed, based on Coulomb's model of static friction, and based on the torque limits of the robot. Notice that none of these things make use of the fact that the robot is fully actuated. So, by approximation the same region of controlled walking should apply for the bounded control of underactuated bipedal robots as well. The analytical approximation of the maximum achievable walking speed is used as an upper bound for the specification of the desired walking speed. It turns out that the maximum achievable walking speed is proportional to the square root of one of the controller bounds, at least for the control design that is presented in this thesis. Note that the approximation of the maximum achievable walking speed can also be used to monitor the actual walking speed of the robot in real-time. In particular, if the actual walking speed exceeds the maximum achievable walking speed, e.g. after a push, then the robot knows it can start to fall over. Thus, the robot can use this information to warn its surroundings and at the same time protect itself by bracing for the fall.

Notice that the controller bounds prevent the use of high-gain feedback control, which is commonly used for increased performance on stationary robots, such as robot manipulators. Obviously, this approach does not work for mobile robots, where it is vital to maintain the delicate contact with the ground. In the case of bipedal robots, controlled walking is much more important than the performance. The control design reflects this by the use of low control gains and a bound on the control action. The low control gains not only make for a smooth walking motion, which looks natural, but it also keeps the energy consumption low. This is useful, because mobile robots can only hold a limited amount of energy. In addition, it is also proposed to try to emulate the inverted pendulum-like walking motion that is observed in humans in order to improve the energy efficiency of the walking gait. However, it turns out that this is not much more efficient than regular walking at a constant speed.

Besides the energy efficiency, it is arguably even more important that the walking gait is robust against disturbances. To achieve robustness, the desired walking gait is defined as a function of the state of the robot. This enables the robot to dynamically react to unexpected disturbances by changing the desired position of the swing foot based on the horizontal velocity of the torso, as well as the location of the torso relative to the stance foot. The former is also regarded as the walking speed of the robot. Hereby, it is essential that the swing foot and the stance foot are always on opposite sides of the torso. The exact foot placement location for the swing foot is less important, especially for a fully actuated bipedal robot. So, it is best to simply use a desired foot placement location that is proportional to the walking speed of the robot.

In this thesis the bipedal robot is controlled so that it walks to a desired location, at which the robot asymptotically comes to a standstill. Note that this requires the robot to be at least fully actuated. The control design is done in such a way that the location of the torso is the only degree of freedom of the closed-loop system, whereby the desired location of the torso is used as the setpoint. This creates a setup that is practical from an operational standpoint, because the robot only needs to know its desired location and it will just go there. Moreover, it will stay there until it receives a different desired location.

## 6.2 Recommendations

Based on the research that is done in this thesis, a number of recommendations can be given with regard to future research on the topic of controlled walking of bipedal robots. First of all, it is recommended to use only relative joint angles to describe the kinematics of the robot. This will reduce the dimension of the state space considerably, albeit at the cost of an increase in the complexity of the equations of motion, which might best be derived now according to the formulation of Lagrange. However, it is necessary to maintain the absolute coordinates of the position and orientation of the torso in order to be able to know the absolute position of the robot and to control the robot to a desired location.

Also, consider to reverse the role that the torque limits play in the control design process. In this thesis, the torque limits are specified based on the actuators that are mounted on the humanoid robot TUlip. These given torque limits are then used as a constraint in the optimization of the controller bounds to try to keep the applied torques within the torque limits during walking. However, if one is in a position to choose or change the actuators of the robot, then it is useful to determine the appropriate torque limits that do not form an active constraint in the optimization of the controller bounds. In this way, the torque limits are specified based on the controller bounds.

The optimization of the controller bounds itself can be improved as well. Currently, all possible cases

of complete saturation of the control action are considered in this optimization problem. The constraints of the optimization problem are evaluated at a commonly used initial state of the robot, which is the state that comes closest to a violation of the contact constraint. However, only one of the cases of complete saturation will actually occur at the initial state. Therefore, consider only the relevant case of complete saturation, because it might produce less conservative controller bounds.

In general, it is important to know the set of parameter values that results in controlled walking. This set is named the region of controlled walking and it has been determined in the parameter subspace of the controller bounds and the desired walking speed of the robot, which are the main parameters in this regard. It has also been shown how the region of controlled walking shrinks in this subspace if larger control gains are used. However, it remains to be shown what the effect is of the other parameters on the size of the region of controlled walking, including the step size, the swing foot clearance, and the desired walking height of the torso. Furthermore, the region of controlled walking should be determined for different initial states of the robot in addition to the initial state that is currently used. It is also important to determine the region of controlled walking experimentally in order to validate the analytical and numerical results that are given in this thesis.

In addition, it is recommended to study the energy efficiency of walking more in depth. In this thesis, the main focus has been on the controlled walking of bipedal robots. However, the energy efficiency of these mobile robots is also important. Therefore, consider to determine the energy efficiency of the walking gait for different values of both the controller bounds and the desired walking speed in the region of controlled walking. Here, the energy efficiency of the walking gait is expressed in terms of the cost of transport. An infinite cost of transport can be assigned to the points outside of the region of controlled walking. Hopefully, there is a minimum cost of transport, which can reveal the natural walking speed of the robot for the current gait design. In addition, the cost of transport can be computed for different step sizes and different walking heights so as to find the parameter values that minimize the cost of transport overall.

With regard to the walking gait, it is recommended to adjust the gait design to incorporate a double support phase during which both feet are in contact with the ground. Notice that this will introduce phases of over-actuation, which the controller needs to be able to handle. Double support is useful in cases where the horizontal velocity of the torso is close to zero. Furthermore, it is desirable to have both feet on the ground if the robot is at a standstill. Currently, the robot still holds the swing foot in the air once it reaches a standstill at the desired location. This issue of the current gait design has to be fixed in order to make it practical and ready for implementation on a real bipedal robot.

The planar bipedal robot that is considered in this thesis is relatively simple in the sense that it has point feet instead of actual feet, and the torso of the robot is constrained to stay upright at all times. By the way, the torso is only constrained in order to have a fully actuated robot. In future research, actual feet can be added to the robot with actuators at the ankle joints, so that the robot can still be fully actuated if the stance foot is flat on the ground. This eliminates the need to constrain the orientation of the torso. Notice that a bipedal robot with actual feet can become underactuated by tipping around an edge of the stance foot, which makes the control of the robot more complex. Fortunately, these cases of unwanted underactuation can be expressed as a violation of the contact constraint as described by Coulomb's model, where the violation will be a result of the loss of contact of one of the contact points of the stance foot. Thus, the control design that is presented in this thesis can be used as a basis for the control of the more complex robot.

# Bibliography


[1] A. Ames, K. Galloway, and J.W. Grizzle. Control Lyapunov functions and hybrid zero dynamics. In *IEEE Conference on Decision and Control*, 2012.

[2] T.M. Assman. Balance recovery of a planar biped using the foot placement estimator. Eindhoven University of Technology, Internship report, 2010.

[3] T.M. Assman. Humanoid push recovery stepping in experiments and simulations. Master's thesis, Eindhoven University of Technology, 2012.

[4] J.A.J. Baelemans. Parameter estimation of humanoid robots using the center of pressure. Master's thesis, Eindhoven University of Technology, 2013.

[5] H. Berghuis and H. Nijmeijer. Global regulation of robots using only position measurements. *Systems and Control Letters*, 21(4):289–295, 1993.

[6] 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, 2009.

[7] Steve Collins, Andy Ruina, Russ Tedrake, and Martijn Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085, 2005.

[8] T. de Boer. *Foot Placement in Robotic Bipedal Locomotion*. PhD thesis, Delft University of Technology, 2012.

[9] R.J. Semeijns de Vries van Doesburgh. Step location prediction for bipedal robots: Evaluating and improving capture point theory. Master's thesis, Delft University of Technology, 2012.

[10] M.H.P. Dekker. Zero-moment point method for stable biped walking. Master's thesis, Eindhoven University of Technology, 2009.

[11] H. Diedam, D. Dimitrov, P.B. Wieber, K. Mombaur, and M. Diehl. Online walking gait generation with adaptive foot positioning through linear model predictive control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1121–1126, 2008.

[12] M. Doi, Y. Hasegawa, and T. Fukuda. Realization of 3-dimensional dynamic walking based on the assumption of point-contact. In *IEEE International Conference on Robotics and Automation*, pages 4120–4125, 2005.

[13] T. Fukuda, M. Doi, Y. Hasegawa, and H. Kajima. *Fast Motions in Biomechanics and Robotics. Lecture Notes in Control and Information Sciences*. Springer, 2006.

[14] R. Goebel, R.G. Sanfelice, and A.R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, 2012.

[15] A. Goswami. Postural stability of biped robots and the foot-rotation indicator point. *International Journal of Robotics Research*, 18(6):523–533, 1999.

[16] J.W. Grizzle, G. Abba, and F. Plestan. Proving asymptotic stability of a walking cycle for a five DOF biped robot model. In *International Conference on Climbing and Walking Robots*, pages 69–81, 1999.

[17] J.W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *IEEE Transactions on Automatic Control*, 46(1):51–64, 2001.

[18] J.W. Grizzle, C. Chevallereau, A.D. Ames, and R.W. Sinnet. 3D bipedal robotic walking: Models, feedback control, and open problems. In *IFAC Symposium on Nonlinear Control Systems*, pages 505–532, 2010.

[19] J.W. Grizzle, F. Plestan, and G. Abba. Poincare's method for systems with impulse effects: Application to mechanical biped locomotion. In *IEEE Conference on Decision and Control*, pages 3869–3876, 1999.

[20] W.P.M.H. Heemels and B. de Schutter. Modeling and control of hybrid dynamical systems. Lecture notes, Eindhoven University of Technology, 2010.

[21] A.L. Hof, S.M. Vermerris, and W.A. Gjaltema. Balance responses to lateral perturbations in human treadmill walking. *Journal of Experimental Biology*, 213(15):2655–2664, 2010.

[22] A.G. Hofmann. *Robust Execution of Bipedal Walking Tasks from Biomechanical Principles*. PhD thesis, Massachusetts Institute of Technology, 2006.

[23] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, second edition, 2012.

[24] J. Hu, J. Pratt, C.M. Chew, H. Herr, and G. Pratt. Adaptive virtual model control of a bipedal walking robot. In *IEEE International Joint Symposia on Intelligence and Systems*, pages 245–251, 1998.

[25] Y. Hurmuzlu, F. Genot, and B. Brogliato. Modeling, stability and control of biped robots: A general framework. *Automatica*, 40(10):1647–1664, 2004.

[26] S.H. Hyon and G. Cheng. Gravity compensation and full-body balancing for humanoid robots. In *IEEE-RAS International Conference on Humanoid Robots*, pages 214–221, 2006.

[27] S.H. Hyon and G. Cheng. Disturbance rejection for biped humanoids. In *IEEE International Conference on Robotics and Automation*, pages 2668–2675, 2007.

[28] S.H. Hyon, R. Osu, and Y. Otaka. Integration of multi-level postural balancing on humanoid robots. In *IEEE International Conference on Robotics and Automation*, pages 1549–1556, 2009.

[29] 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 *IEEE International Conference on Robotics and Automation*, pages 1620–1626, 2003.

[30] K. Kondak and G. Hommel. Control and online computation of stable movement for biped robots. In *International Conference on Intelligent Robots and Systems*, 2003.

[31] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt. Capturability-based analysis and control of legged locomotion part 1: Theory and application to three simple gait models. *International Journal of Robotic Research*, 31(9):1094–1113, 2012.

[32] R.I. Leine and N. van de Wouw. *Stability and Convergence of Mechanical Systems with Unilateral Constraints*. Springer, 2008.

[33] B.E. Maki and W.E. McIlroy. The role of limb movements in maintaining upright stance: The "change-in-support" strategy. *Physical Therapy*, 77(5):488–507, 1997.

[34] M. Millard, D. Wight, J. McPhee, E. Kubica, and D. Wang. Human foot placement and balance in the sagittal plane. *Journal of Biomechanical Engineering*, 131(12), 2009.

[35] 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. In *IEEE Conference on Decision and Control*, pages 4199–4206, 2005.

[36] X. Mu and Q. Wu. Development of a complete dynamic model of a planar five-link biped and sliding mode control of its locomotion during the double support phase. *International Journal of Control*, 77(8):789–799, 2004.

[37] M. Nikkhah, H. Ashrafiuon, and F. Fahimi. Robust control of underactuated bipeds using sliding modes. *Robotica*, 25(3):367–374, 2007.

[38] M.B. Popovic, A. Goswami, and H. Herr. Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. *International Journal of Robotic Research*, 24(12):1013–1032, 2005.

[39] J. Pratt, J. Carff, S. Drakunov, and A. Goswami. Capture point: A step toward humanoid push recovery. In *IEEE-RAS International Conference on Humanoid Robots*, pages 200–207, 2006.

[40] J. Pratt, P. Dilworth, and G. Pratt. Virtual model control of a bipedal walking robot. In *IEEE International Conference on Robotics and Automation*, pages 193–198, 1997.

[41] J.E. Pratt. Virtual model control of a biped walking robot. Master's thesis, Massachusetts Institute of Technology, 1995.

[42] J.E. Pratt, C.M. Chew, A. Torres, P. Dilworth, and G.A. Pratt. Virtual model control: An intuitive approach for bipedal locomotion. *International Journal of Robotic Research*, pages 129–143, 2001.

[43] M.H. Raibert. *Legged Robots that Balance*. MIT Press, 1986.

[44] C.L. Shih, J.W. Grizzle, and C. Chevallereau. Asymptotically stable walking of a simple underactuated 3D bipedal robot. In *Annual Conference of the IEEE on Industrial Electronics Society*, pages 2766–2771, 2007.

[45] R.W. Sinnet and A.D. Ames. 3D bipedal walking with knees and feet: A hybrid geometric approach. In *Proceedings of the IEEE Conference on Decision and Control*, pages 3208–3213, 2009.

[46] R.W. Sinnet and A.D. Ames. Bio-inspired feedback control of three-dimensional humanlike bipedal robots. *Journal of Robotics and Mechatronics*, 24(4):595–601, 2012.

[47] R.W. Sinnet, M.J. Powell, R.P. Shah, and A.D. Ames. A human-inspired hybrid control approach to bipedal robotic walking. In *Proceedings of the IFAC World Congress*, pages 6904–6911, 2011.

[48] G. Song and M. Zefran. Stabilization of hybrid periodic orbits with application to bipedal walking. In *American Control Conference*, 2006.

[49] G. Song and M. Zefran. Underactuated dynamic three-dimensional bipedal walking. In *IEEE International Conference on Robotics and Automation*, pages 854–859, 2006.

[50] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, first edition, 2005.

[51] B. Stephens. Humanoid push recovery. In *IEEE-RAS International Conference on Humanoid Robots*, pages 589–595, 2007.

[52] B. Stephens. Integral control of humanoid balance. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4020–4027, 2007.

[53] B.J. Stephens. *Push Recovery Control for Force-Controlled Humanoid Robots*. PhD thesis, Robotics Institute, Carnegie Mellon University, 2011.

[54] B.J. Stephens and C.G. Atkeson. Dynamic balance force control for compliant humanoid robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1248–1255, 2010.

[55] S.G. Tzafestas, T.E. Krikochoritis, and C.S. Tzafestas. Robust sliding-mode control of nine-link biped robot walking. *Journal of Intelligent and Robotic Systems*, 20(2-4):375–402, 1997.

[56] J. Urata, K. Nshiwaki, Y. Nakanishi, K. Okada, S. Kagami, and M. Inaba. Online decision of foot placement using singular LQ preview regulation. In *IEEE-RAS International Conference on Humanoid Robots*, pages 13–18, 2011.

[57] S.J. van Dalen. A linear inverted pendulum walk implemented on tulip. Master's thesis, Eindhoven University of Technology, 2012.

[58] N. van de Wouw. Multibody dynamics. Lecture notes, Eindhoven University of Technology, 2010.

[59] P.W.M. van Zutven. Modeling, identification and stability of humanoid robots. Master's thesis, Eindhoven University of Technology, 2009.

[60] P.W.M. van Zutven. *Control and Identification of Bipedal Humanoid Robots: Stability Analysis and Experiments*. PhD thesis, Eindhoven University of Technology, 2014.

[61] P.W.M. van Zutven, T.M. Assman, J. Caarls, C. Cilli, T.P.M. Boshoven, E. Ilhan, J.A.J. Baelemans, D.J.F. Heck, M.P.A. Spoelstra, and H. Nijmeijer. Tech United Eindhoven RoboCup adult size humanoid team description. In *RoboCup 2013 Symposium Papers and Team Description Papers*, 2013.

[62] P.W.M. van Zutven, D. Kostic, and H. Nijmeijer. Foot placement for planar bipeds with point feet. In *IEEE International Conference on Robotics and Automation*, pages 983–988, 2012.

[63] M. Vukobratovic and B. Borovac. Zero-moment point - Thirty five years of its life. *International Journal of Humanoid Robotics*, 1(1):157–173, 2004.

[64] E.R. Westervelt, J.W. Grizzle, C. Chevallereau, J.H. Choi, and B. Morris. *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press, first edition, 2007.

[65] E.R. Westervelt, J.W. Grizzle, and D.E. Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE Transactions on Automatic Control*, 48(1):42–56, 2003.

[66] P.B. Wieber. On the stability of walking systems. In *International Workshop on Humanoid and Human Friendly Robotics*, 2002.

[67] P.B. Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *IEEE-RAS International Conference on Humanoid Robots*, pages 137–142, 2006.

[68] D.L. Wight, E.G. Kubica, and D.W.L. Wang. Introduction of the foot placement estimator: A dynamic measure of balance for bipedal robotics. *Journal of Computational and Nonlinear Dynamics*, 3(1), 2008.

[69] S.N. Yadukumar, M. Pasupuleti, and A.D. Ames. From formal methods to algorithmic implementation of human inspired control on bipedal robots. In *International Workshop on the Algorithmic Foundations of Robotics*, 2012.

[70] S.N. Yadukumar, M. Pasupuleti, and A.D. Ames. Human-inspired underactuated bipedal robotic walking with AMBER on flat-ground, up-slope and uneven terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2478–2483, 2012.

# Appendix A

# The multibody dynamics of the planar bipedal robot

The multibody dynamics is given by the differential algebraic equation

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{W}_\sigma(\boldsymbol{q}) \\ \boldsymbol{W}_\sigma^\top(\boldsymbol{q}) & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ \boldsymbol{\lambda}_\sigma \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}\boldsymbol{\tau} - \boldsymbol{H}(\dot{\boldsymbol{q}}) \\ -\boldsymbol{w}_\sigma(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix}, \text{ for } \sigma \in \{\text{lfp}, \text{rfp}\}, \tag{A.1}$$

where the vector of generalized coordinates

$$\boldsymbol{q} = \begin{bmatrix} x_{\text{tr}} & y_{\text{tr}} \,\vdots\, x_{\text{lth}} & y_{\text{lth}} & \theta_{\text{lth}} \,\vdots\, x_{\text{lsh}} & y_{\text{lsh}} & \theta_{\text{lsh}} \,\vdots\, x_{\text{rth}} & y_{\text{rth}} & \theta_{\text{rth}} \,\vdots\, x_{\text{rsh}} & y_{\text{rsh}} & \theta_{\text{rsh}} \end{bmatrix}^\top,$$

and the vector of applied torques

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{\text{lh}} & \tau_{\text{lk}} & \tau_{\text{rh}} & \tau_{\text{rk}} \end{bmatrix}^\top.$$

The mode-dependent vector of kinematic constraint forces $\boldsymbol{\lambda}_\sigma$ is given by

$$\boldsymbol{\lambda}_{\text{lfp}} = \begin{bmatrix} \boldsymbol{\lambda} \\ \lambda_{\text{lf1}} \\ \lambda_{\text{lf2}} \end{bmatrix}, \text{ and } \boldsymbol{\lambda}_{\text{rfp}} = \begin{bmatrix} \boldsymbol{\lambda} \\ \lambda_{\text{rf1}} \\ \lambda_{\text{rf2}} \end{bmatrix},$$

with

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_{\text{lh1}} & \lambda_{\text{lh2}} \,\vdots\, \lambda_{\text{lk1}} & \lambda_{\text{lk2}} \,\vdots\, \lambda_{\text{rh1}} & \lambda_{\text{rh2}} \,\vdots\, \lambda_{\text{rk1}} & \lambda_{\text{rk2}} \end{bmatrix}^\top.$$

The rest of the vectors and matrices that appear in the multibody dynamics (A.1) are composed as follows:

$$\boldsymbol{M} = \text{diag}(\begin{bmatrix} m_{\text{tr}} & m_{\text{tr}} \,\vdots\, m_{\text{th}} & m_{\text{th}} & J_{\text{th}} \,\vdots\, m_{\text{sh}} & m_{\text{sh}} & J_{\text{sh}} \,\vdots\, m_{\text{th}} & m_{\text{th}} & J_{\text{th}} \,\vdots\, m_{\text{sh}} & m_{\text{sh}} & J_{\text{sh}} \end{bmatrix}),$$

$$\boldsymbol{S} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{H}(\dot{\boldsymbol{q}}) = \begin{bmatrix} 0 \\ m_{\text{tr}} g_{\text{a}} \\ 0 \\ m_{\text{th}} g_{\text{a}} \\ d_{\text{h}} \dot{\theta}_{\text{lth}} - d_{\text{k}} \left( \dot{\theta}_{\text{lsh}} - \dot{\theta}_{\text{lth}} \right) \\ 0 \\ m_{\text{sh}} g_{\text{a}} \\ d_{\text{k}} \left( \dot{\theta}_{\text{lsh}} - \dot{\theta}_{\text{lth}} \right) \\ 0 \\ m_{\text{th}} g_{\text{a}} \\ d_{\text{h}} \dot{\theta}_{\text{rth}} - d_{\text{k}} \left( \dot{\theta}_{\text{rsh}} - \dot{\theta}_{\text{rth}} \right) \\ 0 \\ m_{\text{sh}} g_{\text{a}} \\ d_{\text{k}} \left( \dot{\theta}_{\text{rsh}} - \dot{\theta}_{\text{rth}} \right) \end{bmatrix},$$

$$\boldsymbol{W}_{\mathrm{lfp}}(\boldsymbol{q}) = \left[\begin{array}{c:cc} & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 1 & 0 \\ \boldsymbol{W}(\boldsymbol{q}) & 0 & 1 \\ & \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}} & \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}} \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \end{array}\right], \quad \boldsymbol{W}_{\mathrm{rfp}}(\boldsymbol{q}) = \left[\begin{array}{c:cc} & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ \boldsymbol{W}(\boldsymbol{q}) & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 0 & 0 \\ & 1 & 0 \\ & 0 & 1 \\ & \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}} & \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}} \end{array}\right],$$

with

$$\boldsymbol{W}(\boldsymbol{q}) = -\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{l_{\mathrm{th}}}{2}\cos\theta_{\mathrm{lth}} & \frac{l_{\mathrm{th}}}{2}\sin\theta_{\mathrm{lth}} & \frac{l_{\mathrm{th}}}{2}\cos\theta_{\mathrm{lth}} & \frac{l_{\mathrm{th}}}{2}\sin\theta_{\mathrm{lth}} & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{l_{\mathrm{sh}}}{2}\cos\theta_{\mathrm{lsh}} & \frac{l_{\mathrm{sh}}}{2}\sin\theta_{\mathrm{lsh}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{l_{\mathrm{th}}}{2}\cos\theta_{\mathrm{rth}} & \frac{l_{\mathrm{th}}}{2}\sin\theta_{\mathrm{rth}} & \frac{l_{\mathrm{th}}}{2}\cos\theta_{\mathrm{rth}} & \frac{l_{\mathrm{th}}}{2}\sin\theta_{\mathrm{rth}} \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{l_{\mathrm{sh}}}{2}\cos\theta_{\mathrm{rsh}} & \frac{l_{\mathrm{sh}}}{2}\sin\theta_{\mathrm{rsh}} \end{bmatrix},$$

and

$$\boldsymbol{w}_{\mathrm{lfp}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} \boldsymbol{w}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ -\dot{\theta}^2_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}} \\ \dot{\theta}^2_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}} \end{bmatrix}, \quad \boldsymbol{w}_{\mathrm{rfp}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} \boldsymbol{w}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ -\dot{\theta}^2_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}} \\ \dot{\theta}^2_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}} \end{bmatrix},$$

with

$$\boldsymbol{w}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} \dot{\theta}^2_{\mathrm{lth}} \frac{1}{2} l_{\mathrm{th}} \sin\theta_{\mathrm{lth}} \\ -\dot{\theta}^2_{\mathrm{lth}} \frac{1}{2} l_{\mathrm{th}} \cos\theta_{\mathrm{lth}} \\ \dot{\theta}^2_{\mathrm{lth}} \frac{1}{2} l_{\mathrm{th}} \sin\theta_{\mathrm{lth}} + \dot{\theta}^2_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}} \\ -\dot{\theta}^2_{\mathrm{lth}} \frac{1}{2} l_{\mathrm{th}} \cos\theta_{\mathrm{lth}} - \dot{\theta}^2_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}} \\ \dot{\theta}^2_{\mathrm{rth}} \frac{1}{2} l_{\mathrm{th}} \sin\theta_{\mathrm{rth}} \\ -\dot{\theta}^2_{\mathrm{rth}} \frac{1}{2} l_{\mathrm{th}} \cos\theta_{\mathrm{rth}} \\ \dot{\theta}^2_{\mathrm{rth}} \frac{1}{2} l_{\mathrm{th}} \sin\theta_{\mathrm{rth}} + \dot{\theta}^2_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}} \\ -\dot{\theta}^2_{\mathrm{rth}} \frac{1}{2} l_{\mathrm{th}} \cos\theta_{\mathrm{rth}} - \dot{\theta}^2_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}} \end{bmatrix}.$$

# Appendix B

# Differentiation of the system output with respect to time

The output vector of the system is defined as

$$\boldsymbol{y}_\sigma = \left[x_{\mathrm{tr}}, \quad y_{\mathrm{tr}} - \tfrac{1}{2} l_{\mathrm{tr}}, \quad x_{\mathrm{swf},\sigma}(\boldsymbol{q}) - x^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}}, \rho), \quad y_{\mathrm{swf},\sigma}(\boldsymbol{q}) - y^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}}, \rho)\right]^\top,$$

where the position of the swing foot is given by

$$x_{\mathrm{swf},\sigma}(\boldsymbol{q}) = \begin{cases} x_{\mathrm{rsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ x_{\mathrm{lsh}} + \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$
$$y_{\mathrm{swf},\sigma}(\boldsymbol{q}) = \begin{cases} y_{\mathrm{rsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ y_{\mathrm{lsh}} - \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$

and the desired position of the swing foot is given by

$$x^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}}, \rho) = x_{\mathrm{tr}} + \breve{\eta}(x_{\mathrm{tr}}, \rho),$$
$$y^{\mathrm{d}}_{\mathrm{swf}}(x_{\mathrm{tr}}, \rho) = \bar{y}_{\mathrm{swf}} \tanh(\phi \breve{\zeta}(x_{\mathrm{tr}}, \rho)),$$

with

$$\breve{\eta}(x_{\mathrm{tr}}, \rho) = \gamma \tanh(\gamma^{-1}(x_{\mathrm{tr}} - x_{\mathrm{stf}})) \tanh(a\omega_0(\rho - x_{\mathrm{tr}})) \tanh(\gamma^{-1} c^{-\frac{1}{2}} \omega_0(\rho - x_{\mathrm{tr}})),$$
$$\breve{\zeta}(x_{\mathrm{tr}}, \rho) = \left(\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \omega_0(\rho - x_{\mathrm{tr}}) + \sqrt{(\gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \omega_0(\rho - x_{\mathrm{tr}}))^2 + \delta}\right)$$
$$\cdot \left(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \omega_0(\rho - x_{\mathrm{tr}}) + \sqrt{(\gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \omega_0(\rho - x_{\mathrm{tr}}))^2 + \delta}\right).$$

The first and second derivative of the output vector $\boldsymbol{y}_\sigma$ with respect to time will be expressed as

$$\dot{\boldsymbol{y}}_\sigma = \left[\dot{x}_{\mathrm{tr}} \quad \dot{y}_{\mathrm{tr}} \quad \dot{x}_{\mathrm{swf},\sigma} - \dot{x}^{\mathrm{d}}_{\mathrm{swf}} \quad \dot{y}_{\mathrm{swf},\sigma} - \dot{y}^{\mathrm{d}}_{\mathrm{swf}}\right]^\top,$$
$$\ddot{\boldsymbol{y}}_\sigma = \left[\ddot{x}_{\mathrm{tr}} \quad \ddot{y}_{\mathrm{tr}} \quad \ddot{x}_{\mathrm{swf},\sigma} - \ddot{x}^{\mathrm{d}}_{\mathrm{swf}} \quad \ddot{y}_{\mathrm{swf},\sigma} - \ddot{y}^{\mathrm{d}}_{\mathrm{swf}}\right]^\top,$$

respectively, where

$$\dot{x}_{\mathrm{swf},\sigma} = \begin{cases} \dot{x}_{\mathrm{rsh}} + \dot{\theta}_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ \dot{x}_{\mathrm{lsh}} + \dot{\theta}_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$
$$\ddot{x}_{\mathrm{swf},\sigma} = \begin{cases} \ddot{x}_{\mathrm{rsh}} + \ddot{\theta}_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{rsh}} - \dot{\theta}^2_{\mathrm{rsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ \ddot{x}_{\mathrm{lsh}} + \ddot{\theta}_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \cos\theta_{\mathrm{lsh}} - \dot{\theta}^2_{\mathrm{lsh}} \frac{1}{2} l_{\mathrm{sh}} \sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$

$$\dot{x}^{\mathrm{d}}_{\mathrm{swf}} = \dot{x}_{\mathrm{tr}} + \frac{\partial \breve{\eta}}{\partial x_{\mathrm{tr}}}\dot{x}_{\mathrm{tr}} + \frac{\partial \breve{\eta}}{\partial \rho}\dot{\rho},$$
$$\ddot{x}^{\mathrm{d}}_{\mathrm{swf}} = \ddot{x}_{\mathrm{tr}} + \frac{\partial \breve{\eta}}{\partial x_{\mathrm{tr}}}\ddot{x}_{\mathrm{tr}} + \frac{\partial \breve{\eta}}{\partial \rho}\ddot{\rho} + \frac{\partial^2 \breve{\eta}}{\partial x_{\mathrm{tr}}^2}\dot{x}_{\mathrm{tr}}^2 + 2\frac{\partial^2 \breve{\eta}}{\partial x_{\mathrm{tr}}\partial \rho}\dot{x}_{\mathrm{tr}}\dot{\rho} + \frac{\partial^2 \breve{\eta}}{\partial \rho^2}\dot{\rho}^2,$$

$$\dot{y}_{\mathrm{swf},\sigma} = \begin{cases} \dot{y}_{\mathrm{rsh}} + \dot{\theta}_{\mathrm{rsh}}\frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ \dot{y}_{\mathrm{lsh}} + \dot{\theta}_{\mathrm{lsh}}\frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$
$$\ddot{y}_{\mathrm{swf},\sigma} = \begin{cases} \ddot{y}_{\mathrm{rsh}} + \ddot{\theta}_{\mathrm{rsh}}\frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{rsh}} + \dot{\theta}^2_{\mathrm{rsh}}\frac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{rsh}}, & \sigma = \mathrm{lfp} \\ \ddot{y}_{\mathrm{lsh}} + \ddot{\theta}_{\mathrm{lsh}}\frac{1}{2}l_{\mathrm{sh}}\sin\theta_{\mathrm{lsh}} + \dot{\theta}^2_{\mathrm{lsh}}\frac{1}{2}l_{\mathrm{sh}}\cos\theta_{\mathrm{lsh}}, & \sigma = \mathrm{rfp}, \end{cases}$$

$$\dot{y}^{\mathrm{d}}_{\mathrm{swf}} = \bar{y}_{\mathrm{swf}}\phi\left(1 - \tanh^2(\phi\breve{\zeta}(x_{\mathrm{tr}},\rho))\right)\left(\frac{\partial \breve{\zeta}}{\partial x_{\mathrm{tr}}}\dot{x}_{\mathrm{tr}} + \frac{\partial \breve{\zeta}}{\partial \rho}\dot{\rho}\right),$$
$$\ddot{y}^{\mathrm{d}}_{\mathrm{swf}} = -2\bar{y}_{\mathrm{swf}}\phi^2\tanh(\phi\breve{\zeta}(x_{\mathrm{tr}},\rho))\left(1 - \tanh^2(\phi\breve{\zeta}(x_{\mathrm{tr}},\rho))\right)\left(\frac{\partial \breve{\zeta}}{\partial x_{\mathrm{tr}}}\dot{x}_{\mathrm{tr}} + \frac{\partial \breve{\zeta}}{\partial \rho}\dot{\rho}\right)^2$$
$$+ \bar{y}_{\mathrm{swf}}\phi\left(1 - \tanh^2(\phi\breve{\zeta}(x_{\mathrm{tr}},\rho))\right)\left(\frac{\partial \breve{\zeta}}{\partial x_{\mathrm{tr}}}\ddot{x}_{\mathrm{tr}} + \frac{\partial \breve{\zeta}}{\partial \rho}\ddot{\rho} + \frac{\partial^2 \breve{\zeta}}{\partial x_{\mathrm{tr}}^2}\dot{x}_{\mathrm{tr}}^2 + 2\frac{\partial^2 \breve{\zeta}}{\partial x_{\mathrm{tr}}\partial \rho}\dot{x}_{\mathrm{tr}}\dot{\rho} + \frac{\partial^2 \breve{\zeta}}{\partial \rho^2}\dot{\rho}^2\right).$$

In order to be able to write the partial derivatives of the functions $\breve{\eta}(x_{\mathrm{tr}},\rho)$ and $\breve{\zeta}(x_{\mathrm{tr}},\rho)$ in a concise manner, consider that the functions $\breve{\eta}(x_{\mathrm{tr}},\rho)$ and $\breve{\zeta}(x_{\mathrm{tr}},\rho)$ are expressed in the following way:

$$\breve{\eta}(x_{\mathrm{tr}},\rho) = \gamma\Theta_1(x_{\mathrm{tr}})\Theta_2(x_{\mathrm{tr}},\rho)\Theta_3(x_{\mathrm{tr}},\rho),$$

where

$$\Theta_1(x_{\mathrm{tr}}) = \tanh(\gamma^{-1}(x_{\mathrm{tr}} - x_{\mathrm{stf}})),$$
$$\Theta_2(x_{\mathrm{tr}},\rho) = \tanh(a\omega_0(\rho - x_{\mathrm{tr}})),$$
$$\Theta_3(x_{\mathrm{tr}},\rho) = \tanh(\gamma^{-1}c^{-\frac{1}{2}}\omega_0(\rho - x_{\mathrm{tr}})),$$

and

$$\breve{\zeta}(x_{\mathrm{tr}},\rho) = \left(\Theta_4(x_{\mathrm{tr}},\rho) + \left(\Theta_5^2(x_{\mathrm{tr}},\rho) + \delta\right)^{\frac{1}{2}}\right)\left(\Theta_6(x_{\mathrm{tr}},\rho) + \left(\Theta_7^2(x_{\mathrm{tr}},\rho) + \delta\right)^{\frac{1}{2}}\right),$$

where

$$\Theta_4(x_{\mathrm{tr}},\rho) = \gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \omega_0(\rho - x_{\mathrm{tr}}),$$
$$\Theta_5(x_{\mathrm{tr}},\rho) = \gamma + (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \omega_0(\rho - x_{\mathrm{tr}}),$$
$$\Theta_6(x_{\mathrm{tr}},\rho) = \gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) + \omega_0(\rho - x_{\mathrm{tr}}),$$
$$\Theta_7(x_{\mathrm{tr}},\rho) = \gamma - (x_{\mathrm{tr}} - x_{\mathrm{stf}}) - \omega_0(\rho - x_{\mathrm{tr}}).$$

Hereafter, the arguments of the functions $\Theta_i(\cdot)$, for $i = 1,\ldots,7$, are omitted in the interest of abbreviating the expressions of the partial derivatives as much as possible. Now, observe that the partial derivatives

of $\breve{\eta}(x_{\mathrm{tr}}, \rho)$ are given by

$$\begin{aligned}
\frac{\partial \breve{\eta}}{\partial x_{\mathrm{tr}}} &= \left(1 - \Theta_1^2\right) \Theta_2 \Theta_3 - \gamma a \omega_0 \Theta_1 \left(1 - \Theta_2^2\right) \Theta_3 - c^{-\frac{1}{2}} \omega_0 \Theta_1 \Theta_2 \left(1 - \Theta_3^2\right), \\
\frac{\partial \breve{\eta}}{\partial \rho} &= \gamma a \omega_0 \Theta_1 \left(1 - \Theta_2^2\right) \Theta_3 + c^{-\frac{1}{2}} \omega_0 \Theta_1 \Theta_2 \left(1 - \Theta_3^2\right), \\
\frac{\partial^2 \breve{\eta}}{\partial x_{\mathrm{tr}}^2} &= -2\gamma^{-1} \Theta_1 \left(1 - \Theta_1^2\right) \Theta_2 \Theta_3 - 2a\omega_0 \left(1 - \Theta_1^2\right) \left(1 - \Theta_2^2\right) \Theta_3 - 2\gamma^{-1} c^{-\frac{1}{2}} \omega_0 \left(1 - \Theta_1^2\right) \Theta_2 \left(1 - \Theta_3^2\right) \\
&\quad - 2\gamma a^2 \omega_0^2 \Theta_1 \Theta_2 \left(1 - \Theta_2^2\right) \Theta_3 + 2ac^{-\frac{1}{2}} \omega_0^2 \Theta_1 \left(1 - \Theta_2^2\right) \left(1 - \Theta_3^2\right) - 2\gamma^{-1} c^{-1} \omega_0^2 \Theta_1 \Theta_2 \Theta_3 \left(1 - \Theta_3^2\right), \\
\frac{\partial^2 \breve{\eta}}{\partial x_{\mathrm{tr}} \partial \rho} &= a\omega_0 \left(1 - \Theta_1^2\right) \left(1 - \Theta_2^2\right) \Theta_3 + \gamma^{-1} c^{-\frac{1}{2}} \omega_0 \left(1 - \Theta_1^2\right) \Theta_2 \left(1 - \Theta_3^2\right) \\
&\quad + 2\gamma a^2 \omega_0^2 \Theta_1 \Theta_2 \left(1 - \Theta_2^2\right) \Theta_3 - 2ac^{-\frac{1}{2}} \omega_0^2 \Theta_1 \left(1 - \Theta_2^2\right) \left(1 - \Theta_3^2\right) + 2\gamma^{-1} c^{-1} \omega_0^2 \Theta_1 \Theta_2 \Theta_3 \left(1 - \Theta_3^2\right), \\
\frac{\partial^2 \breve{\eta}}{\partial \rho^2} &= -2\gamma a^2 \omega_0^2 \Theta_1 \Theta_2 \left(1 - \Theta_2^2\right) \Theta_3 + 2ac^{-\frac{1}{2}} \omega_0^2 \Theta_1 \left(1 - \Theta_2^2\right) \left(1 - \Theta_3^2\right) - 2\gamma^{-1} c^{-1} \omega_0^2 \Theta_1 \Theta_2 \Theta_3 \left(1 - \Theta_3^2\right).
\end{aligned}$$

In addition, observe that the partial derivatives of $\breve{\zeta}(x_{\mathrm{tr}}, \rho)$ are given by

$$\begin{aligned}
\frac{\partial \breve{\zeta}}{\partial x_{\mathrm{tr}}} &= \left(1 + \omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \left(1 - \omega_0\right)\right) \left(\Theta_6 + \left(\Theta_7^2 + \delta\right)^{\frac{1}{2}}\right) \\
&\quad + \left(\Theta_4 + \left(\Theta_5^2 + \delta\right)^{\frac{1}{2}}\right) \left(-1 - \omega_0 + \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \left(-1 + \omega_0\right)\right), \\
\frac{\partial \breve{\zeta}}{\partial \rho} &= \left(-\omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right) \left(\Theta_6 + \left(\Theta_7^2 + \delta\right)^{\frac{1}{2}}\right) \\
&\quad + \left(\Theta_4 + \left(\Theta_5^2 + \delta\right)^{\frac{1}{2}}\right) \left(\omega_0 - \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right), \\
\frac{\partial^2 \breve{\zeta}}{\partial x_{\mathrm{tr}}^2} &= \left(\left(1 - \omega_0\right)^2 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} - \Theta_5^2 \left(\Theta_5^2 + \delta\right)^{-\frac{3}{2}} \left(1 - \omega_0\right)^2\right) \left(\Theta_6 + \left(\Theta_7^2 + \delta\right)^{\frac{1}{2}}\right) \\
&\quad + 2\left(1 + \omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \left(1 - \omega_0\right)\right) \left(-1 - \omega_0 + \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \left(-1 + \omega_0\right)\right) \\
&\quad + \left(\Theta_4 + \left(\Theta_5^2 + \delta\right)^{\frac{1}{2}}\right) \left(\left(-1 + \omega_0\right)^2 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} - \Theta_7^2 \left(\Theta_7^2 + \delta\right)^{-\frac{3}{2}} \left(-1 + \omega_0\right)^2\right), \\
\frac{\partial^2 \breve{\zeta}}{\partial x_{\mathrm{tr}} \partial \rho} &= \left(\omega_0 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \left(1 - \omega_0\right) - \Theta_5^2 \left(\Theta_5^2 + \delta\right)^{-\frac{3}{2}} \omega_0 \left(1 - \omega_0\right)\right) \left(\Theta_6 + \left(\Theta_7^2 + \delta\right)^{\frac{1}{2}}\right) \\
&\quad + \left(1 + \omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \left(1 - \omega_0\right)\right) \left(\omega_0 - \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right) \\
&\quad + \left(-\omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right) \left(-1 - \omega_0 + \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \left(-1 + \omega_0\right)\right) \\
&\quad + \left(\Theta_4 + \left(\Theta_5^2 + \delta\right)^{\frac{1}{2}}\right) \left(-\omega_0 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \left(-1 + \omega_0\right) + \Theta_7^2 \left(\Theta_7^2 + \delta\right)^{-\frac{3}{2}} \omega_0 \left(-1 + \omega_0\right)\right), \\
\frac{\partial^2 \breve{\zeta}}{\partial \rho^2} &= \left(\omega_0^2 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} - \Theta_5^2 \left(\Theta_5^2 + \delta\right)^{-\frac{3}{2}} \omega_0^2\right) \left(\Theta_6 + \left(\Theta_7^2 + \delta\right)^{\frac{1}{2}}\right) \\
&\quad + 2\left(-\omega_0 + \Theta_5 \left(\Theta_5^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right) \left(\omega_0 - \Theta_7 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} \omega_0\right) \\
&\quad + \left(\Theta_4 + \left(\Theta_5^2 + \delta\right)^{\frac{1}{2}}\right) \left(\omega_0^2 \left(\Theta_7^2 + \delta\right)^{-\frac{1}{2}} - \Theta_7^2 \left(\Theta_7^2 + \delta\right)^{-\frac{3}{2}} \omega_0^2\right).
\end{aligned}$$

Technische Universiteit
Eindhoven
University of Technology

# Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

27-6-2018

Name

Mark Spoelstra

ID-number

0632298

Signature

*Submit the signed declaration to the student administration of your department.*

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU