ZERO-MOMENT POINT METHOD FOR STABLE BIPED WALKING

M.H.P. Dekker
Eindhoven, July 2009
DCT no.: 2009.072

ZERO-MOMENT POINT METHOD FOR STABLE BIPED WALKING

DCT no.: 2009.072

Internship report
Eindhoven, July 2009.

M.H.P. Dekker,
ID: 0610544

Eindhoven, University of Technology

dr. D. Kostic
prof. dr. H. Nijmeijer

Dep. of Mechanical Engineering
Postbus 513
5600 MB Eindhoven
The Netherlands
www.tue.nl

Preface

This report is the outcome of a master's internship carried out at the Dynamics and Control group (DCT) of the Eindhoven University of Technology, the Netherlands.

I would like to thank my coach, dr. Dragan Kostic, for his supervision and his encouraging guidance during the project. Also I would like to thank my student-colleagues: Pieter van Zutven, Atilla Filiz and Richard Kooijman for a pleasant working ambiance and of course their assistance whenever I needed it. Finally, I would also like to give thanks to prof. dr. Henk Nijmeijer, also for supervising the project and for keeping me sharp.

I enjoyed doing this project: it was fun letting a model of a mechanical system move in a human like way and it also supplied me with some insight about the complexity of humans and their movements. The way we walk is pretty refined!

Maarten Dekker, July 2009

Abstract

Biped robots, or legged robots with only two legs, are supposed to mimic human like walking. These robots usually consist of rigid bodies interconnected with joints that are actively or passively actuated.
Human walking is a periodic orbit of a stable phase that alternates with an unstable phase. This is a complex, though highly energy efficient way of biped locomotion, since 'falling' (unstable phase), due to gravity, ensures the forward movement.
Because of its complexity, stability analysis of human walking can be simplified using the Zero-Moment Point (ZMP) criterion. At the cost of energy efficiency, the bipeds' locomotion can be controlled to be always dynamically stable using this criterion.
Bipeds often demonstrate kinematic redundancy, which means that there is no unique mapping from the ZMP reference path to the joint motions. Several strategies to solve this redundancy are proposed. A combination of two strategies, named Walking Primitives and Inverse Differential Kinematics is applied on a dynamic model of TUlip, a humanoid robot available at the Dynamics and Control group (DCT) of the Eindhoven University of Technology, the Netherlands.

Keywords:
Biped Robot, Biped Locomotion, Dynamically Stable Gait, Zero-Moment Point (ZMP), Center of Pressure (CoP), Walking Primitive (WP), Inverse Differential Kinematics.

Contents

Nomenclature

Symbols

F, F	Force vector, Component of force vector	[N]
g	Gravitational acceleration vector	[m s^{-2}]
G	Objective/penalty function	-
H	Angular momentum vector	[N m s]
I	Inertia matrix	[kg m^2]
I	Inertia tensor	[kg m^2]
J	Jacobian matrix	-
K, k	Controller matrix, controller scalar	-
m	Mass	[kg]
M, M	Moment vector, Component of moment vector	[N m]
p, p	Position vector, Component of vector	[m]
P	Linear momentum vector	[N s]
P	Mechanical power	[N m s^{-1}]
q,q	Joint angle vector, joint angle	[rad]
R	Rotation matrix	-
t	Time	[sec]
λ	Lagrange Multiplier	-
$\boldsymbol{\omega}$	Angular velocity vector	[rad s^{-1}]
τ	Torque	[N m]
$\boldsymbol{\Psi}$	Cost function	-

Notation

$\square_{CoM}$	Center of Mass
$\square_{CoP}$	Center of Pressure
$\square_{eq}$	Equality
$\square_{F}$	Friction
$\square_{FCoM}$	Floor Projection of Center of Mass
$\square_{final}$	Final
$\square_{ineq}$	Inequality
$\square_{max}$	Maximum
$\square_{min}$	Minimum

$\square_N$	Normal
$\square_{ref}$	Reference
$\square_R$	Resultant
$\square_{tot}$	Total
$\dot{\square}, \ddot{\square}$	First time derivative, Second time derivative
$\square^T$	Transpose of vector or matrix
$\square^\#$	Pseudo Inverse

Abbreviations

CoM	Center of Mass
CoP	Center of Pressure
DoF	Degree of Freedom
DS, DSP	Double Support, Double Support Phase
FCoM	Floor Projection of Center of Mass
FZMP	Fictitious Zero-Moment Point
SP	Support Polygon
SS, SSP	Single Support, Single Support Phase
TP	Transition Primitive
WP	Walking Primitive
ZMP	Zero-Moment Point

Chapter 1

Introduction

1.1 Biped Locomotion

Legged robots are very diverse, often subdivided in groups indicated by the number of legs. If a legged robot has two legs it is often called a biped. Bipeds are supposed to mimic human like walking. They usually consist of rigid bodies interconnected with joints that are actively or passively actuated.

The ground-breaking works in the field of bipeds were accomplished around 1970 by two famous researchers, Kato and Vukobratovic.
In Japan, the first anthropomorphic robot, WABOT 1, was demonstrated in 1973 by Kato at Waseda University. Using a very simple control scheme, it was able to realize a few slow steps, being statically stable. This achievement was the starting point of a prolific generation of bipeds in Japan.
Parallel to this research, Vukobratovic and his team were very involved in the problems generated by functional rehabilitation. In Belgrade Yugoslavia, his team designed the first active *exoskeletons*[a], but the most well-known outcome remains their analysis of locomotion stability, which exhibited around 1972 the concept of the *Zero-Moment Point*, widely used since then.

The Zero-Moment Point criterion takes the dynamical effects during walking into consideration; therefore it is an extension to the static stability criterion that was used by Kato. The exact interpretation and consequences of this criterion are explained in detail in chapter 3.

1.2 Dutch Robotics and TUlip

Research groups at three universities of technology in the Netherlands, being University of Twente, Delft University of Technology and Eindhoven University of Technology, have agreed to join efforts in creating humanoid robots, cooperating in the *Dutch Robotics* initiative. The Dutch Robotics project is part of a long term vision, shared by three Dutch universities and the Dutch industry for the development of a new generation of robots.

Together they designed *TUlip* (see figure 1.1), a biped robot, especially designed for mimicking human like walking. TUlip is about 1.2 [m] tall and consists of fourteen

[a] An exoskeleton is an external skeleton that supports and protects a human's body.

degrees of freedom (DoFs). Two DoFs in its arms and twelve DoFs in its legs out of which two are unactuated (x-rotation of the ankles, for direction see figure 1.1b). The joints all are rotational and are actuated with DC motors. Planetary gears are used for the transfer of the driving power and some joints are actuated by series elastic actuation: "*An electric motor drives a joint through an elastic/compliant element. By measuring the elongation of this element, the torque that the total actuator system delivers to the joint can be controlled*." [19]

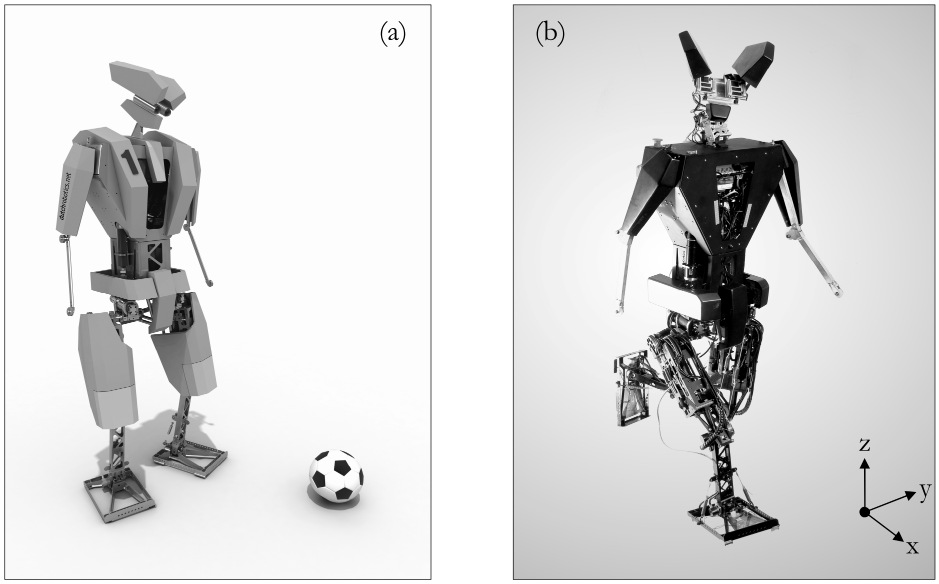

Figure 1.1: TUlip robot:
(a) Rendering of TUlip with protection covers. (b) Photograph of TUlip: no protection covers on the legs.

1.3 Scope of the Project

This report is the outcome of a master's internship carried out at the Dynamics and Control group of the Eindhoven University of Technology. This group is involved in the Dutch initiative to create biped humanoid robots. This initiative faces several research challenges, such as control of a large number of robot DoFs to ensure stable biped locomotion. To come up with systematic control design that would enable safe robot operation of high autonomy, it is important to have solid understanding of the principles of biped locomotion. Given the desired path of the robot in the world coordinate frame, there is a challenge of distributing motions among the robot joints to ensure stable robot walking along the desired path. A stable gait is any set of joint trajectories which enables stable biped locomotion. A stable gait which corresponds to the given world-frame robot path can systematically be determined using principles of the Zero-Moment Point.

In this assignment, focus is on understanding the principles of ZMP and their implementation in a realistic case-study. Therefore, the objective of this assignment is to deliver mathematical formalisms for:

1. *Modeling and monitoring the motion of ZMP of a humanoid robot during biped walking.*
2. *Design of a stable gait for the humanoid robot, given the desired robot path in the world frame.*

Usability of these formalisms is demonstrated in simulation using a realistic model of a humanoid robot.

Chapter 2

Biped Locomotion Fundamentals

2.1 Terminology

2.1.1 Three Dimensional Motion

Positioning a biped in three dimensional space can be done with a *base-frame-origin* and three planes that are all perpendicular to each other. To visualize this, the *anatomical position* [6] is depicted in figure 2.1. The anatomical position is the standing position with the face turned straight forward and the arms hanging along the sides of the body with the palms turned straight forward and the legs stretched with the feet close together. Occasionally, in robot anatomies this anatomical position is with the palms pointing inwards; to the body.
Motions can be described relative to three perpendicular planes through the body. These planes are also depicted in figure 2.1, as well as the base-frame-origin, that is placed with the x-axis pointing forward, the y-axis pointing to the left-hand side, and the z-axis pointing upwards. The origin of this base-frame-origin is at the floor. The definitions of these planes are:

Frontal plane
The plane parallel to the yz-plane is called the Frontal plane.

Sagittal plane
The plane parallel to the xz-plane is called the Sagittal plane. The plane parallel to the Sagittal plane and containing the Center of Mass (CoM) is called the *Median plane*.

Transverse plane
The plane parallel to the xy-plane is called the Transverse plane.

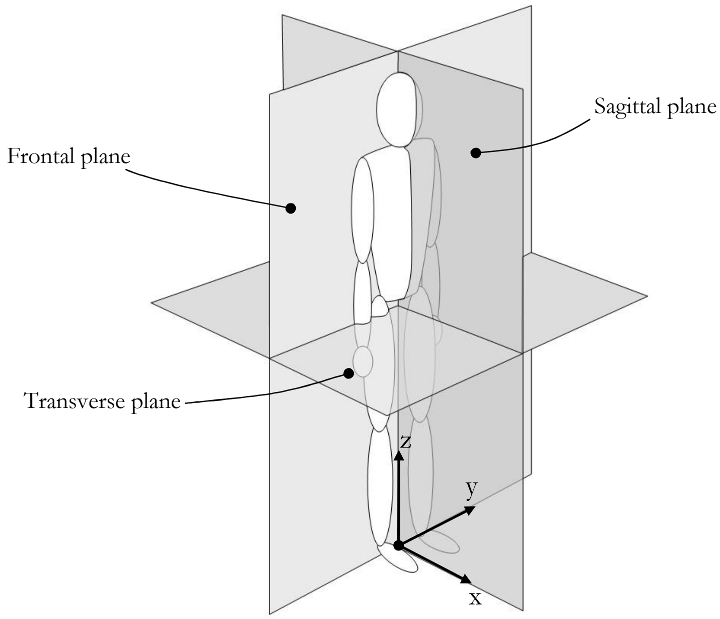

Figure 2.1: Human(oid) in anatomical position, base-frame-origin and world planes. In this case the Sagittal plane coincides with the Median plane.

2.1.2 Gait Analysis

Describing human gait requires some specific terms, which are defined in this section. Some key terms with respect to biped locomotion from [6, 16] are:

Walk
Walk is defined as: *"Movement by putting forward each foot in turn, not having both feet off the ground at once."* Walking backwards and running are not taken into consideration in this report.

Gait
Every human has a specific unique walk, hence gait means: *"Manner of walking or running"*. Moreover, every walk is realized with a certain gait.

Periodic gait
If the gait is realized by repeating each step in an identical way, it is a periodic gait.

Double Support (DS)
This term is used for situations where the biped has two isolated contact surfaces with the floor. This situation occurs when the biped is supported by both feet, but it is not necessarily that both feet are fully supported with the floor (see figure 2.2b).

Single Support: (SS)
This term is used for situations where the biped has only one contact surface with the floor. This situation occurs when the biped is supported with only one foot.

Support Polygon (SP)
The Support Polygon is formed by the *convex hull*[b] about the floor support points. This term is widely accepted for any support area and is shown in figure 2.2.

[b] The convex hull is the boundary of the minimal convex set containing a given non-empty finite set of points in the plane.

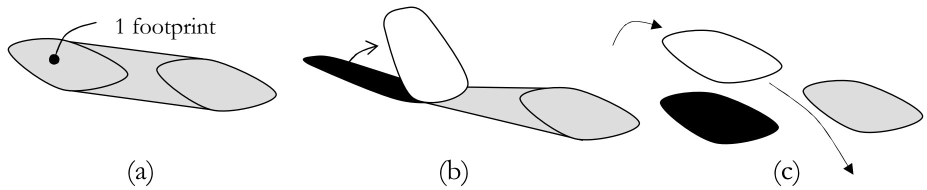

Figure 2.2: Typical shapes of the Support Polygon (SP), in grey:
(a) Double Support (b) Double Support (Pre-Swing Phase). (c) Single Support

Swing leg, swing foot
The leg that is performing a step (moving forward through the air) is denoted with the term swing leg. The foot that is attached to this leg is called the swing foot.

Stance leg, stance foot
While the swing leg is moving through the air, the stance leg is fully supported with the floor by the stance foot and supports all the weight of the biped.

Gait Phases
When the biped is in periodic gait, the gait can be divided into four phases:

1. **Double Support Phase (DSP)**
   This is the phase where both feet are fully supported with the floor, depicted in figure 2.2 a.
2. **Pre-Swing Phase**
   In this phase the heel of the rear foot is lifting from the floor but the biped is still in double support due to the fact that the toes of this foot are still on the floor as depicted in figure 2.2 b.
3. **Single Support Phase (SSP)**
   The phase where only one foot is fully supported with the floor and the other foot swings forward, depicted in figure 2.2 c.
4. **Post-Swing Phase**
   In this phase the toe of the front foot is declining towards the floor. The biped is in double support because the heel of this foot is contacting the floor.

These four phases for each leg form a walking gait; this is depicted in figure 2.3.

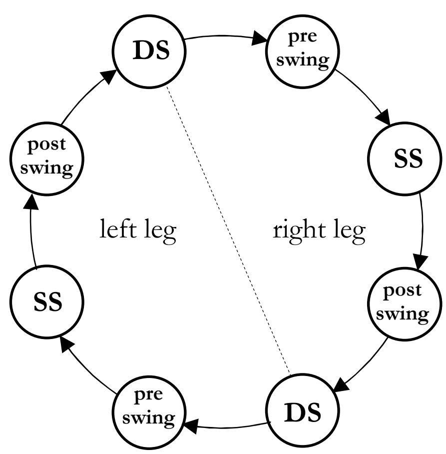

Figure 2.3: Phasing of a periodic gait.

2.2 Stable Gaits

2.2.1 Stability Criteria

To design joint motions offline or online and to design controllers for a biped, there should be some criteria to check if the designed motions and controllers are sufficient to ensure the bipeds stability, i.e. make sure that the biped does not fall or tip over. Therefore, biped stability could be defined as:

Stable Gait
A gait is stable if the only contact between the biped and the floor is realized with the soles of the foot or feet, i.e. no other extremity of the biped is in contact with the floor.

There are several criteria to ensure this stability; some mathematical criteria for this property are discussed in the following.

Floor Projection of the Center of Mass (FCoM)
A motionless biped only experiences gravitational forces, which are exerted on all parts of the robot. These forces can be replaced by a virtual force $\mathbf{R}_N$ acting at the equivalent *Center of Mass* of the biped, indicated with the acronym CoM. The vector $\mathbf{p}_{CoM}$ from the base-frame-origin to the CoM can be described with:

$$\mathbf{p}_{CoM} = \frac{\sum_{i=1}^{n} m_i \mathbf{p}_i}{\sum_{i=1}^{n} m_i} \tag{2.1}$$

Here, the biped has n links and the vectors $\mathbf{p}_i$ are the distances of the individual CoMs of all the links of the biped. The Floor Projection of the CoM, the vector $\mathbf{p}_{FCoM}$, can be taken from vector $\mathbf{p}_{CoM}$, by simply taking the x and y component and set the z component to zero, therefore it also fulfils the relation:

$$\sum_{i=1}^{n} \left(\left(\mathbf{p}_{FCoM} - \mathbf{p}_i \right) \times m_i \mathbf{g} \right) = \mathbf{0} \tag{2.2}$$

If $\mathbf{p}_{FCoM}$ remains in the SP, the motionless biped will not tip over or fall. However, when the motions become faster the dynamic forces will dominate the static forces and this criterion is not sufficient anymore. Hence, other criteria have to be applied.

Zero-Moment Point (ZMP)
The Zero-Moment Point, or ZMP, can be regarded as the dynamical equivalent of the FCoM. The ZMP criterion does take dynamic forces, as well as static forces, into consideration. The derivation of it and its consequence to the stability will not be discussed in this section; but will be explained in detail in chapter 3.

Center of Pressure (CoP)
The Center of Pressure, in short CoP, is the point on the SP of the biped where the total sum of the contact forces $\mathbf{F}_R$ acts, causing a force but no moment. When standing, the part(s) of the body exerted by contact forces is (are) the foot (feet).
There are two types of forces that can be exerted on a foot: tangential and normal forces. These forces are called friction forces, $\mathbf{F}_F$, and (normal) pressure forces, $\mathbf{F}_N$, respectively. If the assumption is made that the foot cannot slide over the surface of the floor, the friction forces cancel each other out, what remains is a pressure field

depicted in figure 2.4b. The resultant (in normal direction) of this pressure field with n contact points, namely

$$F_{RN} = \sum_{i=1}^{n} F_{Ni} \tag{2.3}$$

acts on the CoP. So the position of the CoP with respect to the base-frame-origin, denoted with $\mathbf{p}_{CoP}$, can be calculated with the equation:

$$\mathbf{p}_{CoP} = \frac{\sum_{i=1}^{n} \mathbf{p}_{FNi} F_{Ni}}{F_{RN}} \tag{2.4}$$

The graphical representation of (2.4) is depicted in figure 2.4b. If the CoP is outside the SP, the biped tends to tip over, i.e. the biped will fall.

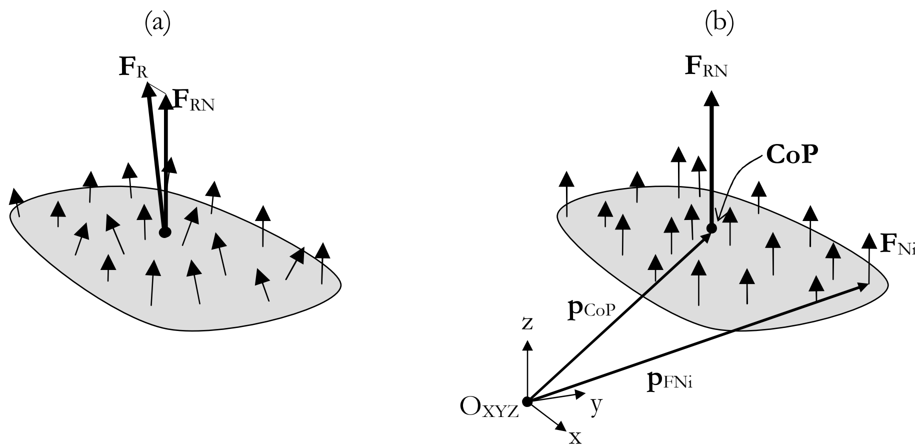

Figure 2.4: (a) the resultant force $\mathbf{F}_R$. (b) Position of the CoP.

2.2.2 Gait Classification

As mentioned in section 2.2.1, the dynamic forces will dominate the static forces if the bipeds' joint velocities increase. The faster the movements the more dominant these dynamic forces will be. There are several classifications for a gait:

Statically Stable Gait

The movement or gait of a biped is called statically stable, if the FCoM and the ZMP always remain within the SP during the entire motion or gait. This implies that if the movement is stopped, the biped will remain in a stable position. These kind of stable gaits are only for really low walking velocities, which impose also low angular velocities in the joints.

Dynamically Stable Gait

If the ZMP resides within the SP during the motion or a gait of a human(oid) while the FCoM leaves the SP, then the motion or gait is called dynamically stable. This kind of gait can be stable for faster movements, but the gait has to meet the requirements of the definition of a *walk*. Running implies that at some time instance, both feet are off the floor, so there is no SP. Hence, running can not be classified with dynamically stable (so obviously not statically stable either).

Limit Cycle Walking

From [18]:

"Limit Cycle Walking" is a relatively new paradigm for the design and control of two-legged walking robots. It states that achieving stable periodic gait is possible without locally stabilizing the walking trajectory at every instant in time, as is traditionally done in most walking robots. Well-known examples of Limit Cycle Walkers are the Passive Dynamic Walkers, but recently there are also many actuated Limit Cycle Walkers. Limit Cycle Walkers generally use less energy than other existing bipeds, but thus far they have not been as versatile.

This report will not consider this type of biped locomotion from now on.

Chapter 3

Zero-Moment Point

Human walking is a periodic orbit of a stable phase that alternates with an unstable phase. This is a complex, though highly energy efficient way of biped locomotion, since 'falling' (unstable phase), due to gravity, ensures the forward movement. Because of its complexity, stability analysis of human walking can be simplified using the Zero-Moment Point (ZMP) criterion. At the cost of energy efficiency, the bipeds' locomotion can be controlled to be always dynamically stable using this criterion.

3.1 Definition

For biped locomotion, the "Zero-Moment Point" is one of the most used and famous terms, it is widely known by the acronym ZMP. Originally, it was defined by Vukobratovic *et al.* in 1972 [13]:

Zero-Moment Point (ZMP)
As the load has the same sign all over the surface, it can be reduced to the resultant force $\boldsymbol{F}_P$*, the point of attack of which will be in the boundaries of the foot. Let the point on the surface of the foot, where the resultant* $\boldsymbol{F}_P$ *passed, be denoted as the Zero-Moment Point.*

To clarify this statement, consider a rigid foot with a flat sole which is fully contacting and supported by the floor, as depicted in figure 3.1. For simplicity, the influence of the biped is replaced with the force $\mathbf{F}_A$ and the moment $\mathbf{M}_A$ acting on a point A on the floor. The gravitational acceleration is $\mathbf{g}$, acting in the negative z direction. To keep the whole biped in balance: in point P the reaction force $\mathbf{F}_P = (F_{PX}, F_{PY}, F_{PZ})$ and the moment $\mathbf{M}_P = (M_{PX}, M_{PY}, M_{PZ})$ are acting.
The horizontal reaction force (F_{PX}, F_{PY}) is the friction force that is compensating for the horizontal components of force $\mathbf{F}_A$. The vertical component of the reaction moment, being M_{PZ}, is balancing the vertical component of moment $\mathbf{M}_A$ and the moment induced by the force $\mathbf{F}_A$. Assuming there is no slip, the static friction is represented with (F_{PX}, F_{PY}) and M_{PZ}.
Before deriving the equilibrium equations, i.e. the static balance equations, a few remarks about the point P. First of all: to compensate for the horizontal components of $\mathbf{M}_A$, being (M_{AX}, M_{AY}), the point P is shifted in such a way that F_{PZ} is fully compensating for them (obviously, with the 'arm' $\mathbf{d} = \mathbf{p}_{OP} - \mathbf{p}_{OA}$, lying on the floor plane). This implies that the horizontal components of $\mathbf{M}_P$ are reduced to zero. Hence:

$$M_{PX} = M_{PY} = 0 \tag{3.1}$$

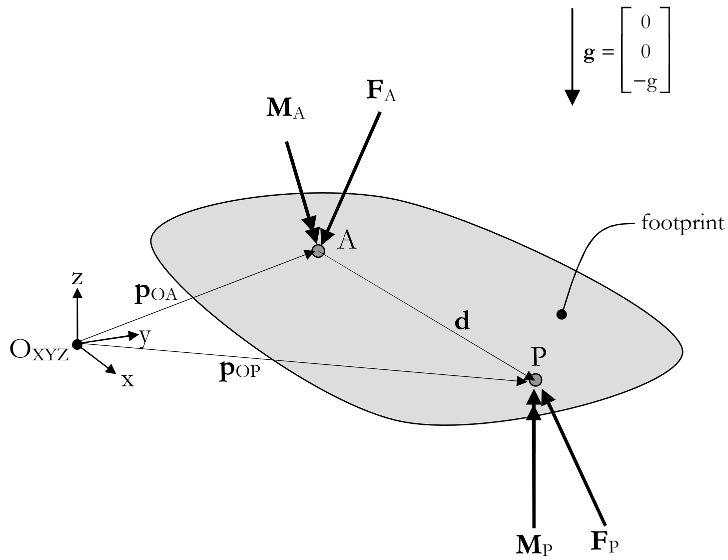

Figure 3.1: Forces and Moments acting on a rigid foot with a flat sole; fully supported with the floor.

This equality led to the natural choice of the term Zero-Moment Point. *"Or in other words, all the time the reaction of the ground due to the foot resting on it can be reduced to the force* $\boldsymbol{F}_P$ *and vertical component of the moment* M_{pz}*; the point P at which the reaction force is acting represents ZMP"* [17].

Secondly, in case the SP is not large enough to include the point P, the force $\mathbf{F}_P$ will act on the foot's edge[c] and the uncompensated part of $\mathbf{M}_A$ and $\mathbf{F}_A$ will result in a rotation about that edge which can result in overturning, i.e. falling of the system (biped). With this information, the static equilibrium equations can be stated as the conservation of forces and of moments around the origin:

$$\mathbf{F}_P + \mathbf{F}_A = \mathbf{0} \tag{3.2}$$

$$\mathbf{p}_{OP} \times \mathbf{F}_P + \mathbf{M}_A + M_{PZ} + \mathbf{p}_{OA} \times \mathbf{F}_A = \mathbf{0} \tag{3.3}$$

In (3.3), the terms $\mathbf{p}_{OP}$ and $\mathbf{p}_{OA}$ are the vectors from the base-frame-origin O_{XYZ} to respectively points P and A as shown in figure 3.1. If the base-frame-origin is placed on the XY-plane, (3.3) provides:

$$(\mathbf{p}_{OP} \times \mathbf{F}_P)_{XY} + (M_A)_{XY} + (\mathbf{p}_{OA} \times \mathbf{F}_A)_{XY} = 0 \tag{3.4}$$

With (3.4) the position of the ZMP can be computed. However, the resulting position of the ZMP does not give an answer to the question: *"Whether for the given motion the mechanism is in dynamic equilibrium?"* [17]. The answer to this question is simple: When the computed point P, being ZMP, is within the SP, the mechanism is in dynamic equilibrium. This leads to the following criterion:

ZMP Criterion
In order to achieve a dynamically stable gait the ZMP should be within the support polygon, at every time instance.

---

[c] To apply a force on a system it is obviously necessary to contact it: outside the SP there is no contact with the system. In addition, in DSP the point P is outside both feet but in the SP; remind that the force $\mathbf{F}_P$ is the resultant of forces for both feet.

3.2 ZMP, FZMP and CoP

As stated above, the point P must lie within the SP of the biped. If the dynamics of the biped change in such a way that the point P in the SP reaches the SP's edge and then moves outside this polygon, what would then be the meaning of this point?

Fictitious ZMP (FZMP)

In (3.4), the point P was determined from the condition $M_{PX} = M_{PY} = 0$. So point P, outside the SP can be stated as a *fictitious ZMP*[d], or in short FZMP. This is because, in reality, the ZMP can only exist within the SP. If the ZMP reaches it edge, the biped is marginally stable, because a given perturbation that leads to a movement of the point P outside the SP (and becoming FZMP) will result in turning over. So the stable region is the SP, without its edges.

Relation with CoP

If, and only if the force $\mathbf{F}_R$, see figure 3.2a, is equilibrated with all *active forces*[e] on the system (biped), the CoP coincides with the ZMP. So in other words, ZMP only coincides with CoP if the biped is dynamically stable, see figure 3.2.

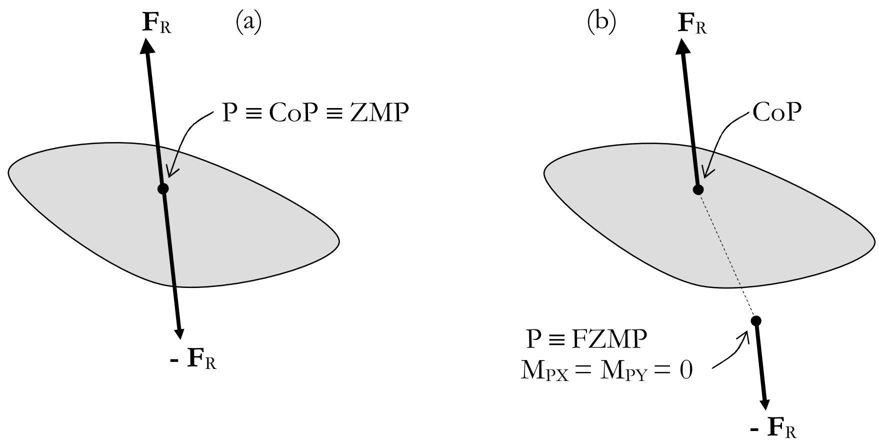

Figure 3.2: Relationship between ZMP, FZMP and CoP:
(a) Dynamically stable. (b) Dynamically unstable.

Summarizing:
If point P is within the SP then the relation P ≡ CoP ≡ ZMP holds. If point P is outside the SP, then points CoP and FZMP do not coincide and P ≡ FZMP. Moreover, if the biped robot is dynamically stable, the position of the ZMP can be calculated with the CoP, e.g. using force sensors on the sole of the feet.

---

[d] Another term that is often used instead of FZMP is FRI, which stands for *Foot Rotation Indicator* [15].

[e] These active forces are forces due to gravity, inertia and Coriolis effects.

3.3 Derivation

3.3.1 By Forward Kinematics

To calculate the point P, there are several assumptions that have to be made:

a) The biped robot consists of n rigid links.
b) All kinematic information, such as position of CoM, link orientation, velocities, etc. are known and calculated by *forward kinematics*.
c) The floor is rigid and motionless.
d) The feet can not slide over the floor surface.
e) All joints are actively actuated.

Under these constraints, the first thing to calculate is the total mass m_{tot}:

$$m_{tot} = \sum_{i=1}^{n} m_i \tag{3.5}$$

of the biped and the distance from the base-frame-origin to its CoM; $\mathbf{p}_{CoM}$ according to (2.1) and a graphical interpretation is illustrated in figure 3.3.

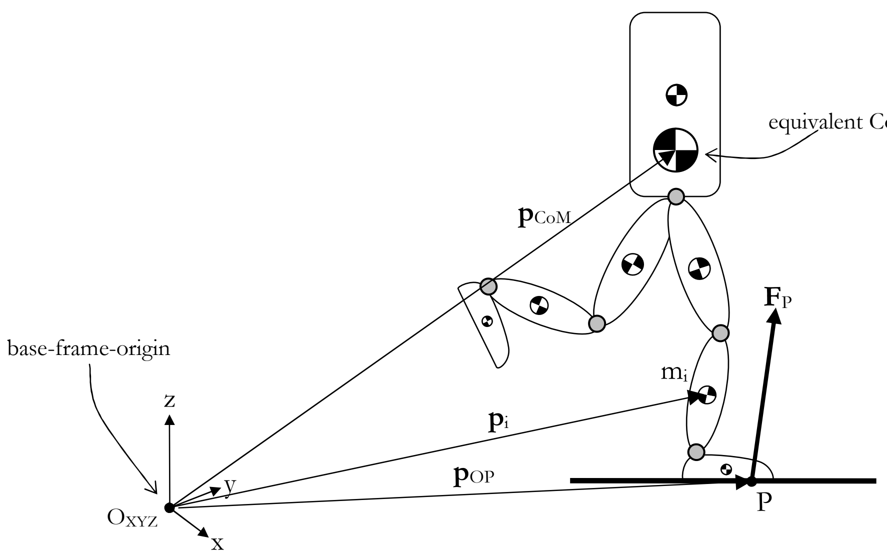

Figure 3.3: Schematic 3D Biped model and point P.

With this the total *linear momentum* **P** and, respectively, the total *angular momentum* **H** with respect of the base-frame-origin can be stated as:

$$\mathbf{P} = \sum_{i=1}^{n} m_i \dot{\mathbf{p}}_i \tag{3.6}$$

$$\mathbf{H} = \sum_{i=1}^{n} \left\{ \mathbf{p}_i \times m_i \dot{\mathbf{p}}_i + \mathbf{I}_i \boldsymbol{\omega}_i \right\} \tag{3.7}$$

In (3.7), $\mathbf{I}_i$ and $\boldsymbol{\omega}_i$ are respectively the inertia tensor and the angular velocities of the i-th link with respect to the base-frame-origin.

For $\mathbf{I}_i$ the following equation holds:

$$\mathbf{I}_i = R_i I_i R_i^T \tag{3.8}$$

with R_i the rotation matrix of the i-th link w.r.t. the base-frame-origin and I_i the inertia matrix of the i-th link w.r.t. the link-frame-origin attached to their links.
The time derivative of **P** and **H** are the rate of change of linear and angular momentums (being a force and a moment), respectively. They can be stated as:

$$\dot{\mathbf{P}} = \sum_{i=1}^{n} m_i \ddot{\mathbf{p}}_i \tag{3.9}$$

$$\dot{\mathbf{H}} = \sum_{i=1}^{n} \left(\dot{\mathbf{p}}_i \times (m_i \dot{\mathbf{p}}_i) + \mathbf{p}_i \times (m_i \ddot{\mathbf{p}}_i) + \mathbf{I}_i \dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times \left(\mathbf{I}_i \boldsymbol{\omega}_i \right) \right) \tag{3.10}$$

where $\dot{\mathbf{p}}_i \times (m_i \dot{\mathbf{p}}_i) = \mathbf{0}$ because $\dot{\mathbf{p}}_i$ and $m_i \dot{\mathbf{p}}_i$ are parallel (note that $m_i \dot{\mathbf{p}}_i$ is a scalar multiplication of $\dot{\mathbf{p}}_i$). With this information the following holds:

$$\mathbf{F}_P = -\mathbf{F}_A = \dot{\mathbf{P}} - m_{tot} \mathbf{g} \tag{3.11}$$

$$\mathbf{M}_O = \dot{\mathbf{H}} - \mathbf{p} \times m_{tot} \mathbf{g} \tag{3.12}$$

Where, as said earlier $\mathbf{F}_P$ and $\mathbf{M}_O$ are the *external*[f] force and moment that describe how the floor is reacting to the biped with respect to the base-frame-origin. $\mathbf{F}_A$ is the force that the biped is acting upon the floor. The *external force*[g] $\mathbf{F}_P$ is acting on point P, so the moment $\mathbf{M}_O$ is:

$$\mathbf{M}_O = \mathbf{p}_{OP} \times \mathbf{F}_P + \mathbf{M}_P \tag{3.13}$$

where $\mathbf{M}_P$ is the moment at P and $\mathbf{p}_{OP}$ is the vector from the base-frame-origin to point P. Because $\mathbf{M}_P$ is on the point P, being either ZMP or FZMP, it is $\mathbf{M}_P = [\ 0\ 0\ M_z\ ]^T$. Now we can substitute (3.12) into (3.13) resulting in:

$$\mathbf{M}_P = \dot{\mathbf{H}} - \mathbf{p}_{CoM} \times m_{tot} \mathbf{g} + (\dot{\mathbf{P}} - m_{tot} \mathbf{g}) \times \mathbf{p}_{OP} \tag{3.14}$$

From this, the distance from the base-frame-origin to the location of the ZMP (or FZMP) $\mathbf{p}_{ZMP} = \mathbf{p}_{OP} = [\ x_{ZMP}, y_{ZMP}, z_{ZMP}\ ]^T$ can be computed:

$$x_{ZMP} = \frac{m_{tot} g_z p_{CoMx} + z_{ZMP} \dot{P}_x - \dot{H}_y}{m_{tot} g_z + \dot{P}_z} \tag{3.15}$$

$$y_{ZMP} = \frac{m_{tot} g_z p_{CoMy} + z_{ZMP} \dot{P}_y + \dot{H}_x}{m_{tot} g_z + \dot{P}_z} \tag{3.16}$$

Remind that z_{ZMP} is the height of the floor. When the XY-plane is placed on the floor, z_{ZMP} obviously becomes zero.

---

[f] To hold the equilibrium of (3.11), the forces should be $\mathbf{F}_{ROBOT} + \mathbf{F}_{EXTERNAL} = \mathbf{0}$. This is stated in (3.2).
[g] $\mathbf{F}_P$, being the external force, is the exact same force as $\mathbf{F}_R$, depicted in figure 3.2a if and only if the robot is dynamically stable.

Equations of Huang *et al.*

Huang *et al.* (2001) [7] assumed that $z_{ZMP} = 0$, and stated the following equations to derive the position of the ZMP:

$$x_{ZMP} = \frac{\sum_{i=1}^{n}\left(m_i(p_{ix}(\ddot{p}_{iz}+g_z) - p_{iz}(\ddot{p}_{ix}+g_x)) - I_{iy}\omega_{iy}\right)}{\sum_{i=1}^{n} m_i(\ddot{p}_{iz}+g_z)} \tag{3.17}$$

$$y_{ZMP} = \frac{\sum_{i=1}^{n}\left(m_i(p_{iy}(\ddot{p}_{iz}+g_z) - p_{iz}(\ddot{p}_{iy}+g_y)) - I_{ix}\omega_{ix}\right)}{\sum_{i=1}^{n} m_i(\ddot{p}_{iz}+g_z)} \tag{3.18}$$

So, in fact, the only distinction from (3.15),(3.16) and (3.17),(3.18) is that the latter equations the rate of linear and angular momentums are expanded in components.

Cart-Table Model

An extremely simplified model to compute the ZMP is the so-called Cart-Table model [13]. Figure 3.4 illustrates a simplified model of a biped, which consists of a running cart on a mass-less table. The cart has mass m and its position (x, z) corresponds to the equivalent CoM of the biped. Also, the table is assumed to have the same SP as the biped.

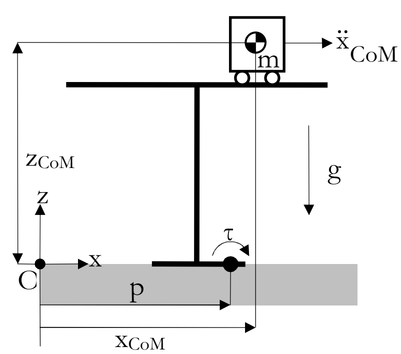

Figure 3.4: Cart-Table model.

In this case, the torque τ around point P can be written as:

$$\tau = -mg(x_{CoM} - p) + m\ddot{x}_{CoM} z_{CoM} \tag{3.19}$$

In (3.19), g is the gravitational acceleration downwards. Now, using the ZMP definition: $\tau = 0$ and thus $x_{ZMP} = p$, this results in

$$x_{ZMP} = p = x_{CoM} - \frac{\ddot{x}_{CoM}}{g} z_{CoM} \tag{3.20}$$

For the y direction the derivation is similar, so:

$$y_{ZMP} = y_{CoM} - \frac{\ddot{y}_{CoM}}{g} z_{CoM} \tag{3.21}$$

3.3.2 By Measuring CoP

As declared earlier, the CoP coincides with the ZMP if, and only if the robot is dynamically stable. If so, the position of it can be calculated through straight forward determination of the resultant of the pressure forces that the floor is acting on a finite number of pressure points. The exact determination of the position of the CoP is described in section 2.2.1.

Note that the position of this CoP can only be determined when at least one pressure point has contact with the floor. To be stable, the ZMP should remain in the SP which is an area, so at least 3 points are necessary to determine this region. Furthermore, if the biped is not dynamically stable while one or more pressure points in contact with the floor the CoP can be determined but this location does not represent the FZMP (see figure 3.2b).

Chapter 4

Trajectory Generation

Bipeds often demonstrate kinematic redundancy, which means that there is no unique mapping from the ZMP reference path to the joint motions. A strategy to solve this redundancy is proposed where stability of the biped walk is maintained.

4.1 Examples of Trajectory Generation Strategies

There are numerous ways to generate joint trajectories in order to achieve a dynamically stable gait. Motion planning algorithms that account for system dynamics typically approach the problem in one of two ways [10]:

- *State-space formulation*: Searching the system state-space directly by reasoning about the possible controls that can be applied.
- *Decoupled approach*: Solving the problem by first computing a kinematic path, and subsequently transforming the path into a dynamic trajectory.

Below the basics of the *Inverted Pendulum Mode* methods are described. These methods are examples of the state-space formulation approach. After this, The *Full-Body Posture Goals* method, an example of the decoupled approach, is described.

Inverted Pendulum Mode Methods

A method to obtain the joint motions is by means of the Inverted Pendulum Mode, with acronym IPM [9]. This method is often denoted with 3D-LIPM [9], because of the three dimensional nature of it.

In this approach, the dynamics of the biped are approximated by the dynamics of an inverted pendulum, linearized about the upper, unstable equilibrium point. The mass is concentrated at the equivalent CoM of the biped, and the base of the pendulum coincides with the center of the stance foot of the biped, as illustrated in figure 4.1. Based on this dynamic model, an appropriate location for the foot placement of the swing foot can be computed in order to counterbalance the tilting motion of the biped.

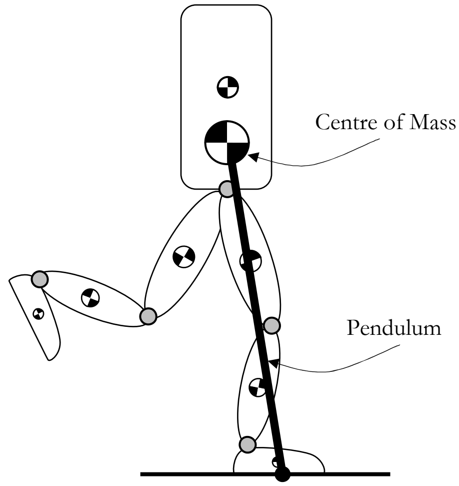

Figure 4.1: Inverted Pendulum Mode.

For the sake of simplicity, the base-frame-origin will be located on the center of the stance foot. If the ZMP is on the center of the stance foot, the biped is dynamically stable. Than the following holds:

$$x_{ZMP} = y_{ZMP} = 0 \tag{4.1}$$

Consequently, the Cart-Table Model equations, namely (3.20) and (3.21) can be written as two ordinary differential equations:

$$\ddot{x}_{CoM} - \frac{g}{z_{CoM}} x_{CoM} = 0 \tag{4.2}$$

$$\ddot{y}_{CoM} - \frac{g}{z_{CoM}} y_{CoM} = 0 \tag{4.3}$$

These equations can be used to define trajectories in Cartesian coordinates for the CoM of the biped, by means of numerical or even analytical integration. Together with a sufficient number of constraints the desired motion in all joints can be derived.

There are numerous methods based on this model, a few examples [5]:
- Virtual Height Inverted Pendulum Mode, VHIPM,
- Two Masses Inverted Pendulum Mode, TMIPM,
- Multiple Masses Inverted Pendulum Mode, MMIPM,
- Gravity Compensated Inverted Pendulum Mode, GCIPM.

Full-body Posture Goals

Given a three dimensional geometric model of the environment and a statically stable goal (end) posture; the configuration space of the biped is searched in order to obtain a collision free path that satisfies that the ZMP remains in the SP. This method is denoted with Full-body Posture Goals [10, 11] and works as follows:

At first the configuration space of the biped is modeled. All biped configurations are stored in a database. All configurations that are characterized by collision between the biped links or between the links and the environment (hence the three dimensional geometric model of the environment is needed), are removed from this space. The resulting space C_{free} consists of all collision-free configurations of the biped. Subsequently, C_{free} is checked for statically stable postures (configurations). The remaining collision free and statically stable configuration space is called C_{valid}.

With a randomized path planning technique and imposing balance constraints the space C_{valid} is searched for a final path. A dynamics filtering function, that constrains the ZMP on the SP, is used to transform this statically stable (and collision free) path into a dynamically stable path.

4.2 Walking Primitives

The Walking Primitive method is a *decoupled approach* to obtain joint trajectories.

4.2.1 Strategy

A systematic approach to generate reference trajectories for all joints is with the use of so called Walking Primitives [3, 4], denoted with acronym WP. WPs are fractions of the walking gait of the biped. Below some examples of WPs are specified:

- performing a left step with step length L_1,
- performing a (right or left) step with (another) step length: $L_2 \neq L_1$,
- performing a (right or left) step on a slope or stair,
- moving the ZMP from the center of one foot to the center of the other in DSP,
- etcetera.

All these WPs are computed offline and are then stored in a database. Using three dimensional information about (moving) obstacles in the walking trail that has to be performed, a step sequence planner can select and concatenate these WPs (during runtime) in order to obtain a situation dependent walking pattern.
A WP is typically characterized by a statically stable begin and end posture in order to connect WPs without dynamical effects. To clarify this: a statically stable posture is a posture with all joint velocities and accelerations exactly zero and the ZMP lies within the SP. No velocities and accelerations imply that the ZMP coincides with FCoM, thus, the FCoM should lie within the SP.
Between those two postures the trajectories of the joints are determined by means of an optimal control problem:

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \begin{cases} \mathbf{c}_{eq}(\mathbf{x}) = \mathbf{0} \\ \mathbf{c}_{ineq}(\mathbf{x}) \leq \mathbf{0} \\ \mathbf{A}_{eq}\mathbf{x} = \mathbf{b}_{eq} \\ \mathbf{A}_{ineq}\mathbf{x} \leq \mathbf{b}_{ineq} \\ \mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub} \end{cases} \tag{4.4}$$

The objective function f($\mathbf{x}$) of this optimal control problem could be a performance index that is based on the mechanical power that the biped uses to go from one posture to another. Also the ZMP error, i.e. the distance that the ZMP is diverted from its ideal position (center of the foot in swing phase), could be used. These objective functions need to be minimized subject to different equality and inequality constraints denoted with the subscripts $\square_{eq}$ and $\square_{ineq}$, respectively. Linear and non-linear constraints, respectively in the form $\mathbf{Ax} = \mathbf{b}$ and $\mathbf{c}(\mathbf{x}) = \mathbf{0}$ are separated just to speed up the optimization algorithm. Examples of the constraints are ZMP-stability, positional and collision-avoidance conditions. Physical admissibility of the WPs also demands restrictions given by the performance limits of the motors and by the mechanics. The $\mathbf{x}$ variables that are optimized can be bounded with a lower (**lb**) and an upper (**ub**) bound.

4.2.2 Search Space Specification for a General WP: a Step

A stated earlier, in general a step could be divided into three phases:

1) pre-swing phase, time: t_0 - t_1,
2) swing phase, time: t_1 - t_2,
3) post-swing phase, time: t_2 - t_3,

These phases are supported by figure 4.2.

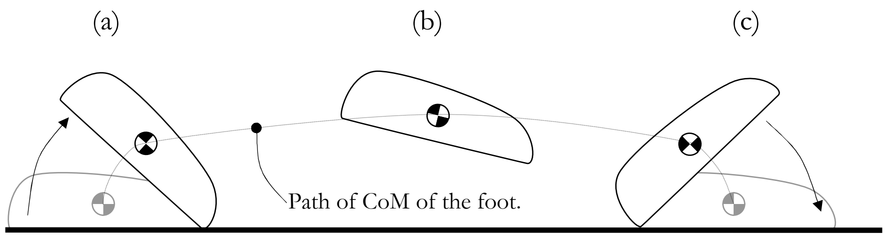

Figure 4.2: Swing foot during the three phases of the step motion: (a) pre-swing. (b) swing. (c) post-swing.

In figure 4.2, the grey lined feet at the beginning and at the end of the step are part of the characterizing stable postures of the step. For example, for a left step, these stable postures are (schematically) depicted in figure 4.3. The step starts with a stable posture Q_0 where both feet are fully supported by the floor and the right foot is half a step length ahead of the left foot. The step ends with the left foot half a step length in advance of the right foot. This is posture will be denoted with Q_3.

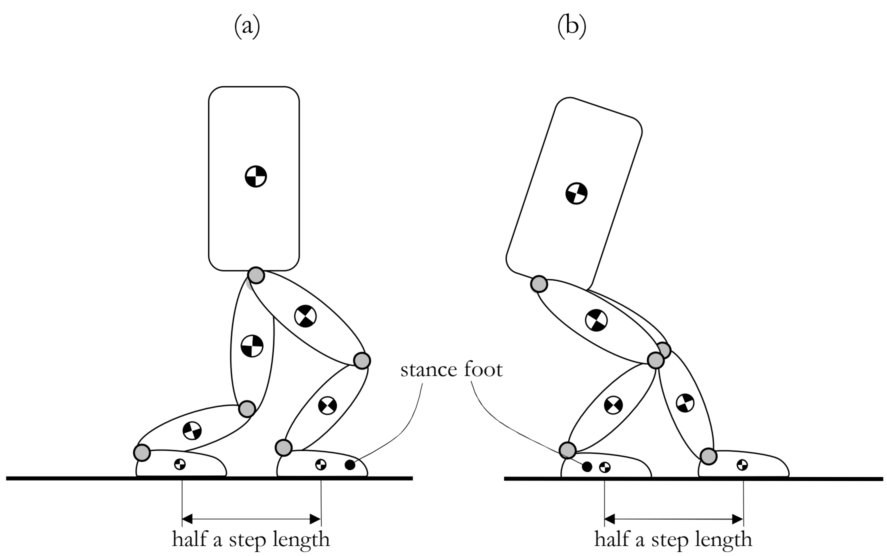

Figure 4.3: Statically stable postures that characterize the start (a) and end (b) configuration of the biped when performing a (left) step, posture Q_0 and Q_3, respectively. Note that he stance foot does not move during the whole step.

Optimal Control Problem Formulation

The search space for a WP together with the dynamical system specifies a whole family of feasible joint motions. The question that arises now is: *"What is the optimal set of joint trajectories?"* The answer to that question is dependent on who is answering this, so the choice of the objective function can be argued.

One option of this choice is an objective function that calculates the mechanical power of by the biped:

$$P(t) = \sum_{i=1}^{n} \tau_i(t)\dot{q}_i(t) \tag{4.5}$$

With n the number of actively actuated joints, the joint torque is denoted with τ_i and the joint velocity with $\dot{q}_i$. This function needs to be minimized by taking into account the dynamics of the system. The input to the dynamical system, namely the desired joint motions $q_{di}(t)$, are polynomial functions whose coefficients represent:

$$q_{di}(t) = \sum_{j=0}^{m} x_{ij} t^j \tag{4.6}$$

Where m is the order of the polynomials and t is the time. The number of optimization variables is n×(m+1). These optimization variables are unbounded, thus, this optimization problem is also unbounded.

Constraints

The following constraints must be satisfied during all three phases of the step:

- The stance foot does not move during the step. This imposes equality constraints to the position and the orientation of this foot.
- The joints can not move beyond their physical boundaries, so inequality constraints need to be constructed.
- At the end of phase i (i = 1,2,3), at t_i the state of the robot $[\mathbf{q}_{end}, \dot{\mathbf{q}}_{end}]$ should be the same as the initial state $[\mathbf{q}_{ini}, \dot{\mathbf{q}}_{ini}]$ of the next phase i+1, in order to get a smooth concatenation between phases without dynamical effects due to discontinuities. This implies equality constraints at t_i and t_{i+1} to the whole state of the biped.

The following phases are subject to constraints that are solely for that particular phase:

Pre-swing phase

- The toe of the moving foot remains on the floor, the foot only changes in orientation. So, an equality constraint on the position of the toe is required.
- At the end of this phase, at t_1, the ZMP should be in the convex hull of the stance foot (in the swing phase this becomes the SP). Because the biped is not in motion at t_1, the ZMP coincides with the projection of the equivalent CoM on the floor. The whole weight of the biped is now supported by the stance foot. The swing foot is now ready to lift from the floor.

Swing phase

- The foot that swings must be above the floor until t_2. Thus, an inequality constraint is needed.
- To remain stable, the ZMP should remain in the SP which is formed only by the sole of the stance foot. This results in inequality constraints.

Post-swing phase

- To remain stable, the ZMP should remain in the SP which is formed now by the sole of the stance foot and the heel of the moving foot. This results in inequality constraints.
- The heel of the moving foot remains on the floor, the foot only changes in orientation. So, an equality constraint on the position of the heel is required.

Extra constraints can be added, at will, to each phase. If a lot of constraints are added, a solution can be found if the order m of the polynomials is increased. Some constraints can even lead to no solution.

4.2.3 Concatenation of WPs

It is assumed that a gait always consists of the alternation of left and right steps. So a left step is always followed by a right step and vice versa. All WPs should be designed in such way that they can smoothly be concatenated into a resulting smooth reference trajectory for each joint. This imposes: to connect WP_a to WP_b, the end state of WP_a $[\mathbf{q}_{end}, \dot{\mathbf{q}}_{end}]$ should be exactly the same as the initial state $[\mathbf{q}_{ini}, \dot{\mathbf{q}}_{ini}]$ of WP_b.
Not all WPs can be concatenated. The step sequence planner that is used should be provided with information which WPs can be cancatenated to each other and which can not. This can be done easily with marking the WPs with 'flags' for the initial and final state.

WP mapping
If the biped is symmetrical, it is simple to derive a WP for a right step, if a left step is already generated. This is easily achieved by interchanging the joint angles (and velocities) of the left and right leg, if the direction of rotations of the joints is chosen appropriately. If several WPs are computed a choice has to be made. If a lot of memory is provided, one can choose to derive all left and right steps in advance and place them all in a WP-database. The concatenation then is very fast, there are (almost) no computations needed. If there is not much memory available one can choose to store only the left steps in the database and with a mapping the right steps can be computed, this increases the computation time but decreases the memory requirement. Thus, this choice is very much dependent on the configuration of the biped and the other functions that it has to perform.

Transition Primitives (TP)
If many WPs are stored, the biped can perform complex gaits, such as (multiple combinations of):

- stepping over objects,
- walking in different curves
- walking on stairs or slopes (both up and down),
- walking backwards,
- etcetera.

In order to be able to do this, the biped needs so called *Transition Primitives* [4], denoted with TP. These primitives ensure concatenation of WPs of one cyclic gait to another. For instance: concatenate a gait with step length L_1 (cyclic gait 1) to a gait with step length L_2 (cyclic gait 2). In figure 4.4 this TP is shown.
Another example of a TP could be the shifting of the ZMP from one foot center to the other in order to perform a following step. The choice that has to be made is: "Is the shifting part connected to each walking primitive, so the biped is immediately ready for the next step, or is the shifting part a TP, so it can be used in more situations?"

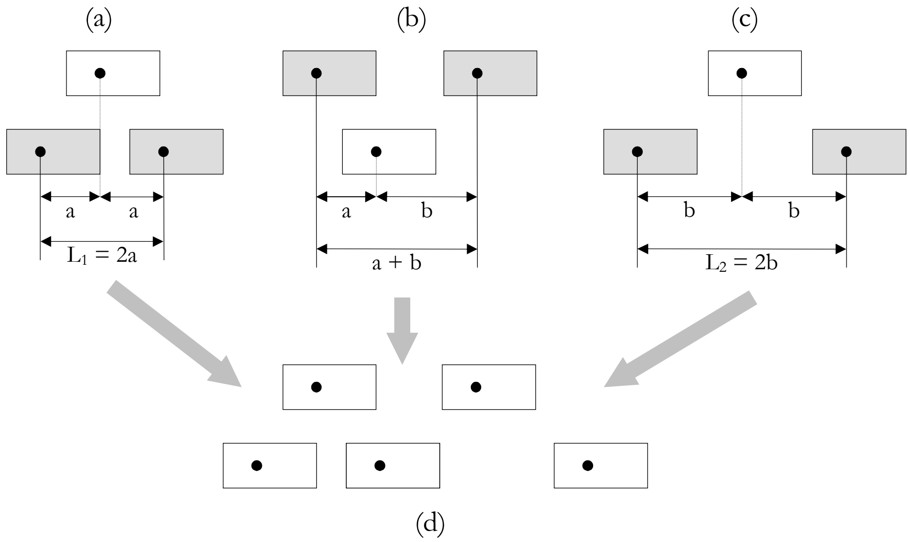

Figure 4.4: Changing from one step length to the other by a transition primitive: (a) Cyclic walking primitive with step length L_1. (b) Transition primitive. (c) Cyclic walking primitive with step length L_2. (d) Walking pattern after concatenation. Stance foot is in white, Swing foot is in grey, the ankle is indicated with the black dot.

4.3 Inverse Differential Kinematics

The following method is an example of the state-space formulation approach. This section is obtained from [12, 14] and is another way to solve the optimal control problem.

4.3.1 Introduction

Generally in manipulator robotics, the mapping of the joint space **q** to the position and orientation of the end-effector **x**, defined with:

$$\mathbf{x} = [x_1 \quad x_2 \quad \cdots \quad x_m]^T \tag{4.7}$$

is described by non-linear functions, especially if there are a lot of rotational joints. Often, in biped locomotion, this end-effector is the swing foot, whereas the base is the stance foot.
If the kinematic relation between the joint space and the operation space is derived with respect to time, the *Differential Kinematics Equation* can be defined as:

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{4.8}$$

The dimension of **x** is (m × 1), with $0 \le m \le 6$. In (4.8) **J** is the Jacobian matrix, which can be defined as:

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \frac{\partial x_1}{\partial q_1} & \cdots & \frac{\partial x_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_m}{\partial q_n} & \cdots & \frac{\partial x_m}{\partial q_n} \end{bmatrix} \tag{4.9}$$

where **x** is the trajectory for the tip (task space). Suppose that a motion trajectory is assigned to the end-effector, for instance the trajectory for the swing foot, in terms of velocities and initial conditions on position and orientation. The aim is to determine feasible joint trajectories [$\mathbf{q}$(t), $\dot{\mathbf{q}}$(t)] that reproduces the given trajectory. By considering n = m, the joint velocities can be computed via the inversion of the Jacobian matrix, since this matrix is square:

$$\dot{\mathbf{q}} = \left(\mathbf{J}(\mathbf{q})\right)^{-1}\dot{\mathbf{x}} \tag{4.10}$$

From this and the initial posture $\mathbf{q}(0)$ the joint positions can be obtained by integrating the velocities over time:

$$\mathbf{q}(t) = \int_0^t \dot{\mathbf{q}}(t)dt + \mathbf{q}(0) \tag{4.11}$$

In discrete time this integration can be done, e.g. with the *Euler integration method*: given a integration interval Δt and known positions and velocities at t_k, the joint positions at t_{k+1} can be computed as:

$$\mathbf{q}(t_{k+1}) = \mathbf{q}(t_k) + \dot{\mathbf{q}}(t_k)\Delta t \tag{4.12}$$

Redundancy
If a system is kinematically redundant, so m < n, the inverse of the Jacobian can not be computed since the Jacobian has more columns than rows. This redundancy, often seen in humanoids or bipeds, ensures that the above strategy will not give a feasible solution. Such a feasible solution method is to recast the gait design as a constrained linear optimization problem, which will be explained in detail in the next section.

4.3.2 Solving the Inverse Kinematics

Once the end-effector velocity $\dot{\mathbf{x}}$ and Jacobian $\mathbf{J}$ are given (for a given configuration $\mathbf{q}$), it is desired to find the solutions $\dot{\mathbf{q}}$ that satisfy the linear equation in (4.8) and minimize the quadratic cost function of joint velocities:

$$\psi(\dot{\mathbf{q}}) = \tfrac{1}{2}\dot{\mathbf{q}}^T\mathbf{W}\dot{\mathbf{q}} \tag{4.13}$$

Where $\mathbf{W}$ is a, yet to be determined, suitable (n × n) weighting matrix. This problem could be solved using the *method of Lagrange multipliers*. Consider the modified cost function:

$$\psi'(\mathbf{q},\boldsymbol{\lambda}) = \tfrac{1}{2}\dot{\mathbf{q}}^T\mathbf{W}\dot{\mathbf{q}} + \boldsymbol{\lambda}^T\left(\dot{\mathbf{x}} - \mathbf{J}\dot{\mathbf{q}}\right) \tag{4.14}$$

where the Lagrange multipliers are $\boldsymbol{\lambda}$ (m × 1). In this way, (4.8) is embedded as a constraint, because if the biped exactly follows the reference trajectory, (4.14) reduces to (4.13) again. The requested solution of the minimization has to satisfy the necessary conditions:

$$\left(\frac{\partial\psi}{\partial\dot{\mathbf{q}}}\right) = \mathbf{0} \quad \text{and} \quad \left(\frac{\partial\psi}{\partial\boldsymbol{\lambda}}\right) = \mathbf{0} \tag{4.15}$$

From the first condition this yields: $\mathbf{W}\dot{\mathbf{q}} - \mathbf{J}^T\boldsymbol{\lambda} = \mathbf{0}$ and rewriting this leads to:

$$\dot{\mathbf{q}} = \mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}\boldsymbol{\lambda} \tag{4.16}$$

The other condition, leads to the constraint stated in equation (4.8). If the two conditions are combined, the result is:

$$\dot{\mathbf{x}} = \mathbf{J}\mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}\boldsymbol{\lambda} \tag{4.17}$$

If **J** has full rank, $\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}$ is a square matrix, size (m × m) which can be inverted. So, (4.17) can be solved for **λ**:

$$\boldsymbol{\lambda} = \left(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}\right)^{-1}\dot{\mathbf{x}} \tag{4.18}$$

If (4.18) is substituted into (4.16) the optimal solution can be obtained:

$$\dot{\mathbf{q}} = \mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}\left(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^{\mathrm{T}}\right)^{-1}\dot{\mathbf{x}} \tag{4.19}$$

If (4.19) is pre-multiplied with **J** on both sides of the equal sign, the equation reduces to the differential kinematics equation (4.8). When **W** is identical to the identity matrix a particular case occurs: (4.19) simplifies to

$$\dot{\mathbf{q}} = \mathbf{J}^{\#}\dot{\mathbf{x}} \tag{4.20}$$

In which $\mathbf{J}^{\#}$ is called the *(right) pseudo inverse* of **J**, indicated with:

$$\mathbf{J}^{\#} = \mathbf{J}^{\mathrm{T}}\left(\mathbf{J}\mathbf{J}^{\mathrm{T}}\right)^{-1} \tag{4.21}$$

To add more constraints to the redundant system, for instance $\dot{\mathbf{q}}_{\alpha}$; another modified cost function, as an alternative for (4.14), can be stated as:

$$\psi'(\mathbf{q},\boldsymbol{\lambda}) = \tfrac{1}{2}\left(\dot{\mathbf{q}} - \dot{\mathbf{q}}_{\alpha}\right)^{\mathrm{T}}\left(\dot{\mathbf{q}} - \dot{\mathbf{q}}_{\alpha}\right) + \boldsymbol{\lambda}^{\mathrm{T}}\left(\dot{\mathbf{x}} - \mathbf{J}\dot{\mathbf{q}}\right) \tag{4.22}$$

If the same algorithm is performed to solve the Lagrange multipliers, the result is:

$$\dot{\mathbf{q}} = \mathbf{J}^{\#}\dot{\mathbf{x}} + \left(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J}\right)\dot{\mathbf{q}}_{\alpha} \tag{4.23}$$

This solution can be recognized as the combination of two terms. The first term is relative to the minimum joint velocities, the second term, which is named *homogeneous solution*, attempts to satisfy the additional constraint specified with $\dot{\mathbf{q}}_{\alpha}$. The specification of this constraint can be done with:

$$\dot{\mathbf{q}}_{\alpha} = -k_{\alpha}\left(\frac{\partial G(\mathbf{q},\dot{\mathbf{q}})}{\partial \mathbf{q}} + \frac{\partial G(\mathbf{q},\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right)^{\mathrm{T}} \tag{4.24}$$

The solution moves along the negative gradient of the objective function G, so this attempts to find a local minimum.

For G any function can be chosen. A typical function is to 'constrain' the ZMP, for example:

$$G = x_{ZMP}^2 + y_{ZMP}^2 \tag{4.25}$$

or

$$G = \frac{\left(x_{SP,max} - x_{SP,min}\right)^2}{\left(x_{SP,max} - x_{ZMP}\right)\left(x_{ZMP} - x_{SP,min}\right)} + \frac{\left(y_{SP,max} - y_{SP,min}\right)^2}{\left(y_{SP,max} - y_{ZMP}\right)\left(y_{ZMP} - y_{SP,min}\right)} \tag{4.26}$$

In which $x_{SP,max}$ is the maximum x position for the SP and $x_{SP,min}$ is the minimum. Similar to this are the y boundaries stated, $y_{SP,max}$ and $y_{SP,min}$, respectively. In this case, G is a penalty function that increases when the ZMP moves towards the edges, which is supported with figure 4.5.

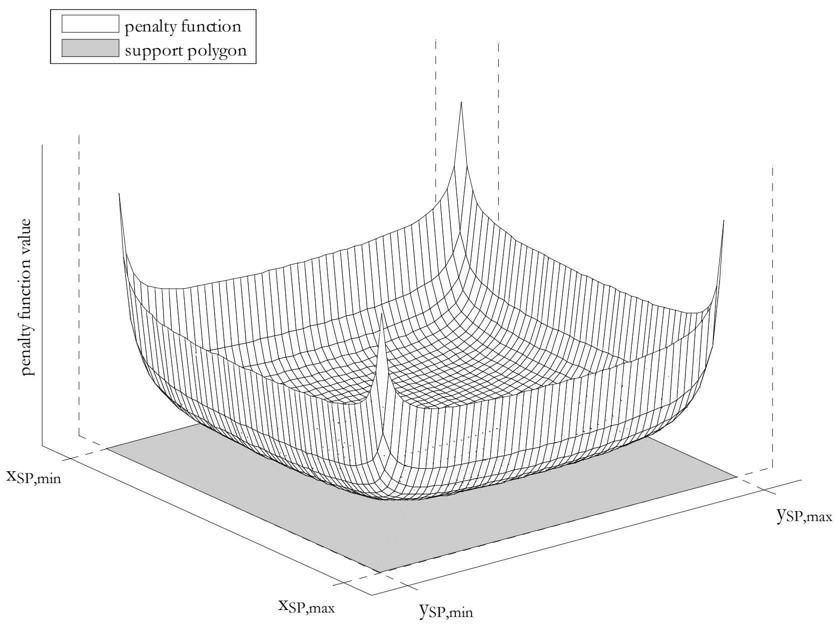

Figure 4.5: Physical interpretation of penalty function described with equation (4.26).

Another interesting choice of the objective function G is to reach a final posture for the biped:

$$G = (\mathbf{Q} - \mathbf{q})^T (\mathbf{Q} - \mathbf{q}) \tag{4.27}$$

Where **Q** is the desired posture and **q** the actual posture of the biped. So for each joint the distance to the desired posture has to be minimized, which eventually leads to reaching **Q**. Note that for G also combinations of functions can be used, so:

$$G = \sum_{i=1}^{r} k_i G_i \tag{4.28}$$

With r the number of objective (or penalty) functions. The weight of a function is set with a gain k_i.

4.3.3 Minimizing the Objective Function

By putting together equations (4.23) and (4.24) the result is:

$$\dot{\mathbf{q}} = \mathbf{J}^{\#}\dot{\mathbf{x}}_{\mathrm{ref}} + (\mathbf{I} - \mathbf{J}^{\#}\mathbf{J})\left(-k\frac{\partial G}{\partial \mathbf{q}}\right)^{T} \tag{4.29}$$

If $\dot{\mathbf{x}} = \mathbf{0}$, (4.29) guarantees monotonous decrease of the objective function G because:

$$\begin{aligned} \dot{G} &= \frac{\partial G}{\partial \mathbf{q}}\dot{\mathbf{q}} \\ &= -k\frac{\partial G}{\partial \mathbf{q}}(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J})\left(\frac{\partial G}{\partial \mathbf{q}}\right)^{T} \\ &= -k\frac{\partial G}{\partial \mathbf{q}}(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J})(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J})^{T}\left(\frac{\partial G}{\partial \mathbf{q}}\right)^{T} \\ &\le 0 \end{aligned} \tag{4.30}$$

In this proof, the *Idempotency*[h] and *symmetry*[i] [12] of (**I** - **J**#**J**) is used. Note that if $\dot{\mathbf{x}} \neq \mathbf{0}$, equation (4.29) does not necessarily mean the nonpositiveness of $\dot{G}$ or the monotonous decrease of G, although the second term of equation (4.29) still tends to decrease G.

4.3.4 Second-order Algorithms

For control purposes, it can be convenient to specify trajectories of the joint motions at acceleration level. Moreover, a robot can be regarded as a second order dynamical system. Time differentiation of the differential kinematics equation (4.8) leads to:

$$\ddot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} \tag{4.31}$$

If the system is kinematically redundant, the solution according to the method of Lagrange Multipliers boils down to:

$$\ddot{\mathbf{q}} = \mathbf{J}^{\#}\left(\ddot{\mathbf{x}} - \dot{\mathbf{J}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\right) + \left(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J}\right)\ddot{\mathbf{q}}_{\alpha} \tag{4.32}$$

In (4.32), the additional constraint $\ddot{\mathbf{q}}_{\alpha}$ can be specified with:

$$\ddot{\mathbf{q}}_{\alpha} = -k_{\alpha}\left(\frac{\partial G(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})}{\partial \mathbf{q}} + \frac{\partial G(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} + \frac{\partial G(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})}{\partial \ddot{\mathbf{q}}}\right)^{T} \tag{4.33}$$

---

[h] A square matrix $\mathbf{A} \in R^{n \times n}$ is called idempotent if $\mathbf{A}^2 = \mathbf{A}$

[i] A square matrix $\mathbf{A} \in R^{n \times n}$ is called symmetric if $\mathbf{A}^T = \mathbf{A}$

Numerical Integration

Numerical integration of the above system, to reconstruct the joint velocities and positions, unavoidably lead to *drift* [j] [22] in the solution. Therefore, it is useful to consider the so called *operational space error* between the desired and actual end-effector position and orientation. Let:

$$\mathbf{e} = \mathbf{x}_{\text{ref}} - \mathbf{x} \tag{4.34}$$

where $\mathbf{x}_{\text{ref}}$ is a desired trajectory for the tip which is described in position and orientation. A feedback control scheme is designed so that the following equation prescribes the closed loop dynamics of the operational space error:

$$\ddot{\mathbf{e}} + \mathbf{K}_{\text{d}}\dot{\mathbf{e}} + \mathbf{K}_{\text{p}}\mathbf{e} = \mathbf{0} \tag{4.35}$$

Where $\mathbf{K}_{\text{d}}$ and $\mathbf{K}_{\text{p}}$ are positive definite matrices, typically diagonal. These feedback coefficient matrices guarantee that $\mathbf{e} \rightarrow \mathbf{0}$ as $t \rightarrow 0$.
The second time derivative of this error is simply

$$\ddot{\mathbf{e}} = \ddot{\mathbf{x}}_{\text{ref}} - \ddot{\mathbf{x}} \tag{4.36}$$

Thus, together with (4.31), this can be rewritten as:

$$\ddot{\mathbf{e}} = \ddot{\mathbf{x}}_{\text{ref}} - \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} \tag{4.37}$$

If (4.35) and (4.37) are put together, the result is:

$$\ddot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{-1}\left(\ddot{\mathbf{x}}_{\text{ref}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{K}_{\text{d}}\dot{\mathbf{e}} + \mathbf{K}_{\text{p}}\mathbf{e}\right) \tag{4.38}$$

Subsequently, the whole redundant system with additional constraints is:

$$\ddot{\mathbf{q}} = \mathbf{J}^{\#}\left(\ddot{\mathbf{x}}_{\text{ref}} - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{K}_{\text{d}}\dot{\mathbf{e}} + \mathbf{K}_{\text{p}}\mathbf{e}\right) + \left(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J}\right)\ddot{\mathbf{q}}_{\alpha} \tag{4.39}$$

Summarizing:
With (4.39), the inverse kinematics for a redundant system such as a biped robot can be solved, under constraints for the tip: $\ddot{\mathbf{x}}_{\text{ref}}$ and additional constraints $(\mathbf{I} - \mathbf{J}^{\#}\mathbf{J})\ddot{\mathbf{q}}_{\alpha}$.
In (4.39) a PD-controller can be recognized in: $\mathbf{K}_{\text{d}}\dot{\mathbf{e}} + \mathbf{K}_{\text{p}}\mathbf{e}$. This part will take care of the drift that would occur when numerical integration is performed.

[j] Due to numerical errors, the satisfaction of constraint equations can deteriorate when the simulation process proceeds. This phenomenon is called drift [22].

Chapter 5

Implementation to Model of TUlip

5.1 Dynamic Model

As mentioned before, in order to run optimizations, a *dynamical* model is needed. A *kinematic* configuration of TUlip is illustrated in figure 5.1. Here, TUlip is modeled as a kinematic chain with eleven links, out of which ten can be actuated actively. The kinematic modeling can be done, for instance using the *Denavit-Hartenberg convention*[k] [15]. In order to orientate and position TUlip in the operational space, a six-link arm attached to its upper body is added to the model. This arm exhibits three translational DoFs and three rotational DoFs. All six links are mass-less and with (initial) length zero.
Based on this DH representation, the equations of motions can be derived[l]. The derived equations of motion can be numerically integrated with *Matlab*®. This model also takes care of the impact and lift off events of the toes and heels of TUlip with the floor.

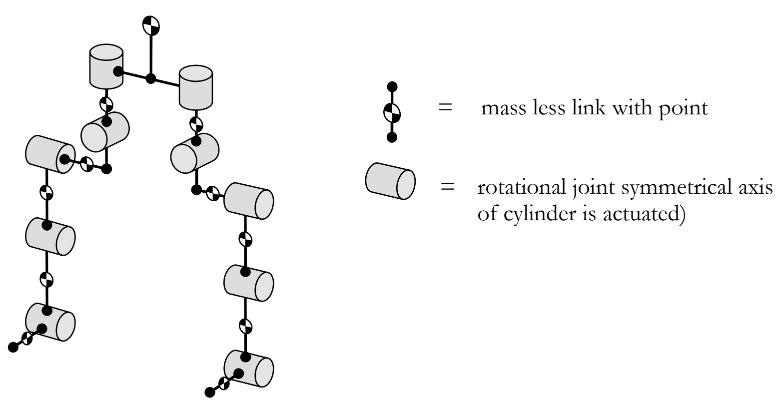

Figure 5.1: Schematic overview of anatomical kinematic structure of TUlip.

[k] The Denavit-Hartenberg convention allows modeling with four parameters per link instead of the usual six variables to represent the operational space.
[l] The complete dynamical model is created as part of the graduation project of Pieter van Zutven. This project is carried out at the department of Mechanical Engineering in Eindhoven, University of Technology.

5.2 Generating Trajectories

5.2.1 Strategy

A strategy to obtain all joint motions for a stable gait is a combination of the *Method of Walking Primitives* and *Inverse Differential Kinematics*. Thus, a decoupled approach is used: trajectories generated with a kinematic model are subsequently implemented in the dynamic model. The assignment of the Denavit-Hartenberg parameters to Tulips' kinematic model is explained in Appendix A. One WP, in this case a left step, will be characterized by initial and final (both statically stable) postures, as well as 2 intermediate postures to ensure that the step mimics a human like step. The trajectories of one WP will be derived with Inverse Differential Kinematics. To do this, the trajectory of swing foot is described together with additional constraints; the ZMP should remain within the SP, the postures should be reached and the velocities of each joint should be as low as possible (to minimize the used energy).

5.2.2 Performing a Left Step

Below, the postulated conditions for generating the joint motions to let TUlip perform a left step are explained.

Characterizing Statically Stable Postures

With *Robotics Toolbox for Matlab*®, the initial (Q_0), intermediate (Q_1 and Q_2) and final (Q_3) postures are determined to be statically stable. In this way it is possible to design trajectories from one posture to another, which can be concatenated to each other without discontinuities in joint acceleration, joint velocity and joint position.
The following can be stated about the postures:

- The t*rack width*[m] of the biped is the same for the whole step.
- The feet have the same *orientation* in the two postures at t_0 and t_3: pointing straight ahead and flat on the floor.
- At t_1, the *toes* of the swing foot and the whole stance foot are exactly on the floor, at t_2 the whole stance foot and the *heel* of the swing foot are on the floor.

These postures are depicted in figure 5.2.

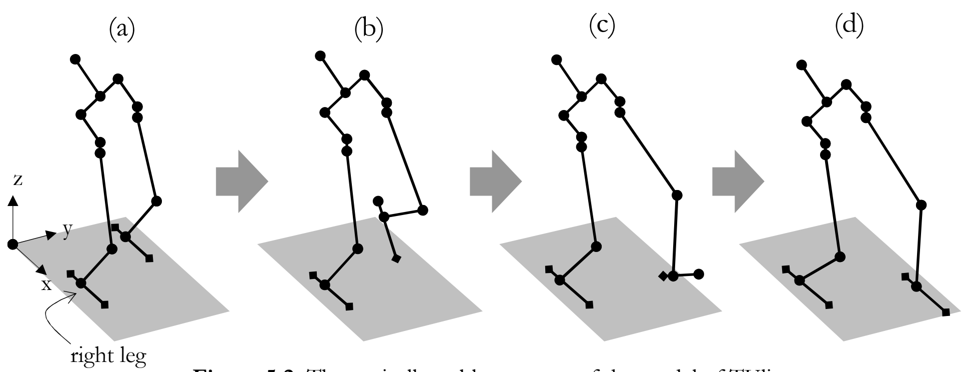

Figure 5.2: The statically stable postures of the model of TUlip:
(a) Q_0: start of step; both feet fully supported with the floor. (b) Q_1: end of pre-swing phase, start of SSP; stance foot and toes of swing foot on the floor (c) Q_2: end of SSP, start of post-swing phase; stance foot and heel of swing foot on the floor (d) Q_3: end of step; both feet fully supported with the floor.
Square dot: this point is in contact with the floor

---

[m] The track width of the robot is the horizontal distance between the feet, if they are parallel, measured in the frontal plane.

Reference Trajectory

According to section 4.3.4, the trajectory of the tip of the end-effector, in this case being the *toe of the swing foot*, needs to be pre-described:

$$\mathbf{x}_{ref}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \theta_{roll}(t) \\ \theta_{pitch}(t) \\ \theta_{yaw}(t) \end{bmatrix} \tag{5.1}$$

At time instances t_0, t_1, t_2 and t_3 $\mathbf{x}_{ref}$ is known from the postures as determined above. The trajectories in between are computed using *quintic interpolation*[n] [1]. The results are in figure 5.3 and 5.4.

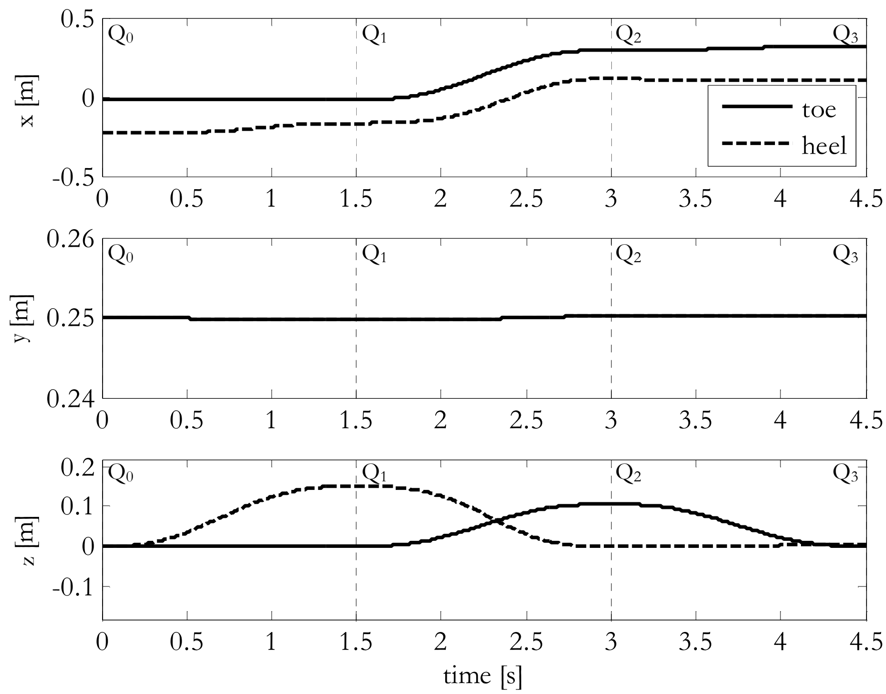

Figure 5.3: The first three elements of $\mathbf{x}_{ref}$; the toe positions, determined with quintic interpolation between postures Q_0, Q_1, Q_2 and Q_3. The heel positions are derived from x, y, z in combination with θ_{roll}, θ_{pitch} and θ_{yaw}.

Objective function

Also according section 4.3.4, additional constraints should be added. During a left step, the SP changes per phase, depicted in figure 5.5. As can be seen in this figure, the stance foot is always part of the SP. The ZMP is constrained to be always within this stance foot. Because this foot is square, the SP can be identified with:

$$\mathbf{SP} = \begin{bmatrix} SP_{x,min} & SP_{x,max} & SP_{y,min} & SP_{y,max} \end{bmatrix}^T \tag{5.2}$$

So for each phase, the penalty function G_1, is exactly as stated in (4.26).

[n] To avoid discontinuities on acceleration level there are constraints on initial and final position, velocity and acceleration (six in total). A fifth order polynomial is required to ensure these constraints, hence: quintic interpolation [1].

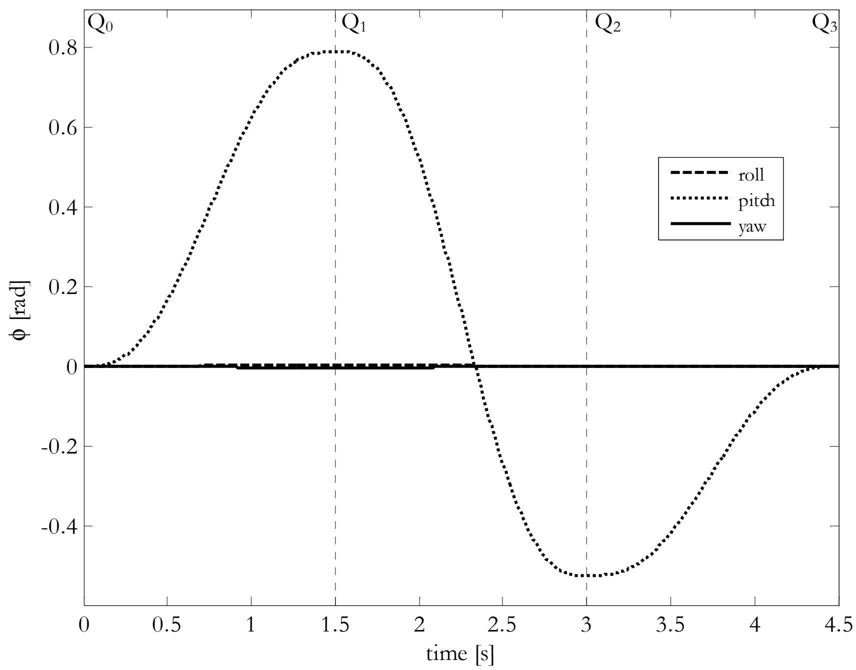

Figure 5.4: The last three elements of $\mathbf{x}_{ref}$; the roll, pitch and yaw angles of the swing foot, determined with quintic interpolation in between postures Q_0, Q_1, Q_2 and Q_3.

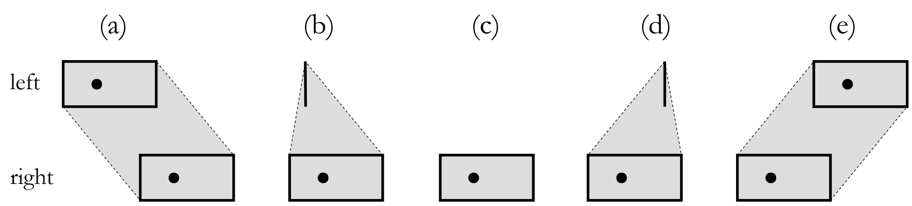

Figure 5.5: The SP during a left step:
(a) Q_0; only at instance t_0. (b) Q_1; pre-swing phase; from t_0 - t_1. (c) swing phase; from t_1 - t_2. (d) Q_2; post-swing phase; from t_2 - t_3. (e) Q_3; only at instance t_3.

To reach the final postures of each phase, necessary to concatenate each phase without discontinuities in the acceleration or velocity, function G_2 is stated as (4.27). The end postures are depicted in figure 5.2. Finally, to locally minimize the actuation effort the function G_3 is chosen as the sum of squares of all joint velocities:

$$G_3 = \dot{\mathbf{q}}^T \dot{\mathbf{q}} \tag{5.3}$$

The total objective function is now stated as:

$$G = k_1 G_1 + k_2 G_2 + k_3 G_3 \tag{5.4}$$

5.2.3 Algorithm

As mentioned before, the algorithm computes the joint acclerations by means of Inverse Differential Kinematics; see equation (4.39). The corresponding joint velocities and positions are computed with the Euler integration method; see equation (4.12).

The algorithm that computes these joint motions is shown in Appendix B. In this, the computation of $\ddot{q}_\alpha$ is slightly different from (4.33); not one tuning parameter k_α but three parameters are used. This is done to get more freedom in tuning the gate. As a matter of fact, more parameters can be implemented at will, to get an improved result. Improved means: a gait that is considered more aesthetic or better according to a criterion which is adopted in addition to the ZMP criterion.

5.3 Simulation Results

The results of the algorithm can be evaluated accurately enough with some plots. The first three figures; 5.6, 5.7 and 5.8 depict the movement of both legs and feet. The three phases, being pre-swing, swing and post-swing can be recognized in these plots.

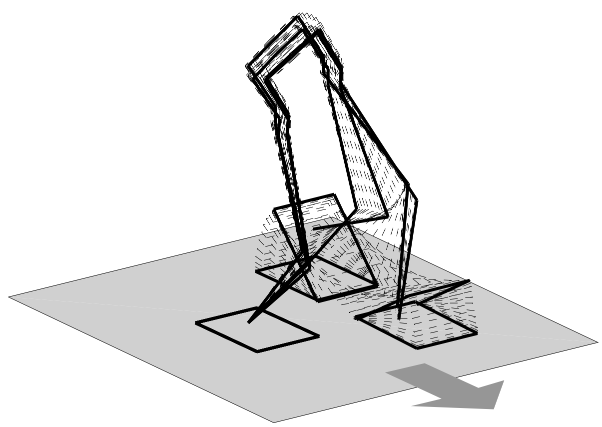

Figure 5.6: A three dimensional view of a left step.
Gray area represents the floor, bold lines: postures Q_0, Q_1, Q_2 and Q_3, arrow indicates direction of the gait.

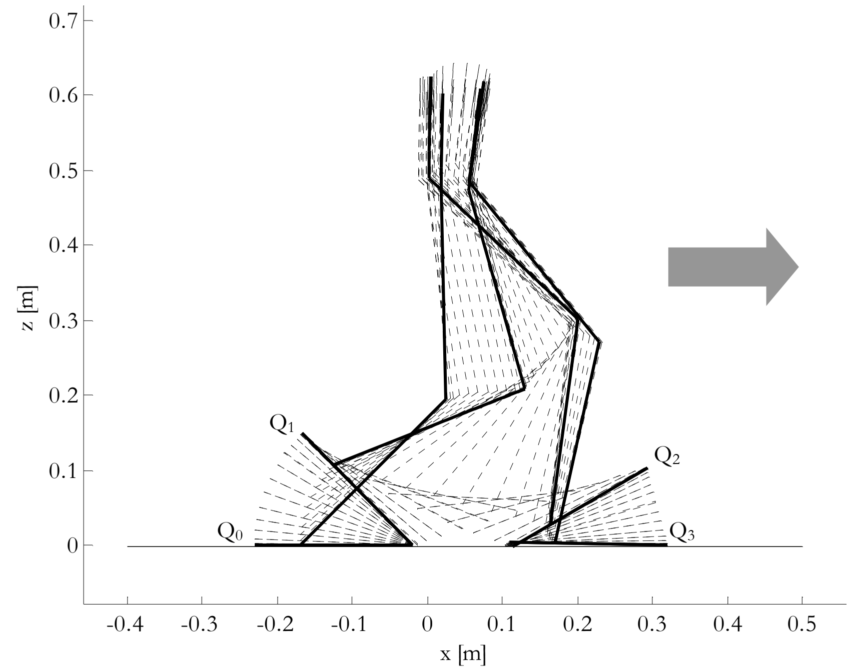

Figure 5.7: Movement of the swing leg and foot (left) in the Sagittal plane. Bold lines: postures Q_0, Q_1, Q_2 and Q_3, arrow indicates direction of the gait.

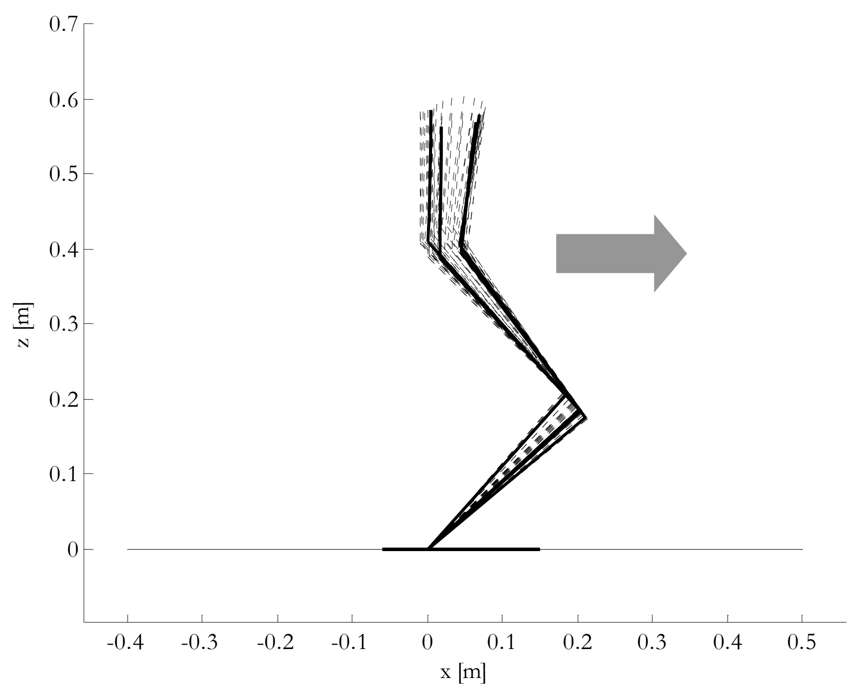

Figure 5.8: Movement of the stance leg and foot (right) in the Sagittal plane. Bold lines: postures Q_0, Q_1, Q_2 and Q_3, arrow indicates direction of the gait.

The ZMP and the FCoM both remain within the SP, so the step can be classified dynamically stable and statically stable. This is supported by figure 5.9. In figure 5.10 the paths of the ZMP and FCoM are again plotted, more focused. This figure shows that the postures are statically stable because at that time instance the ZMP coincides with the FCoM.

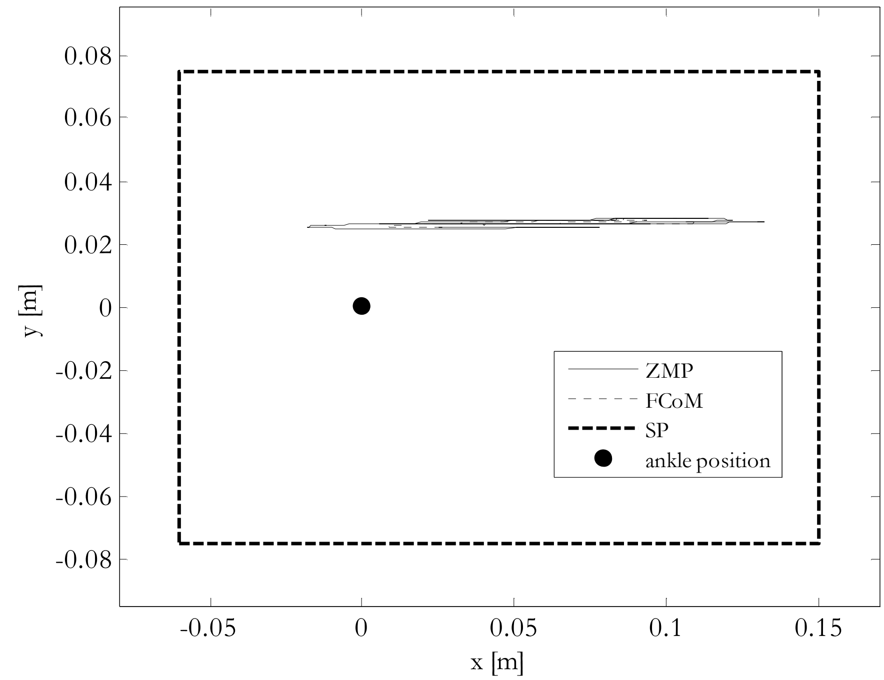

Figure 5.9: The paths of ZMP and FCoM in relation with the SP.

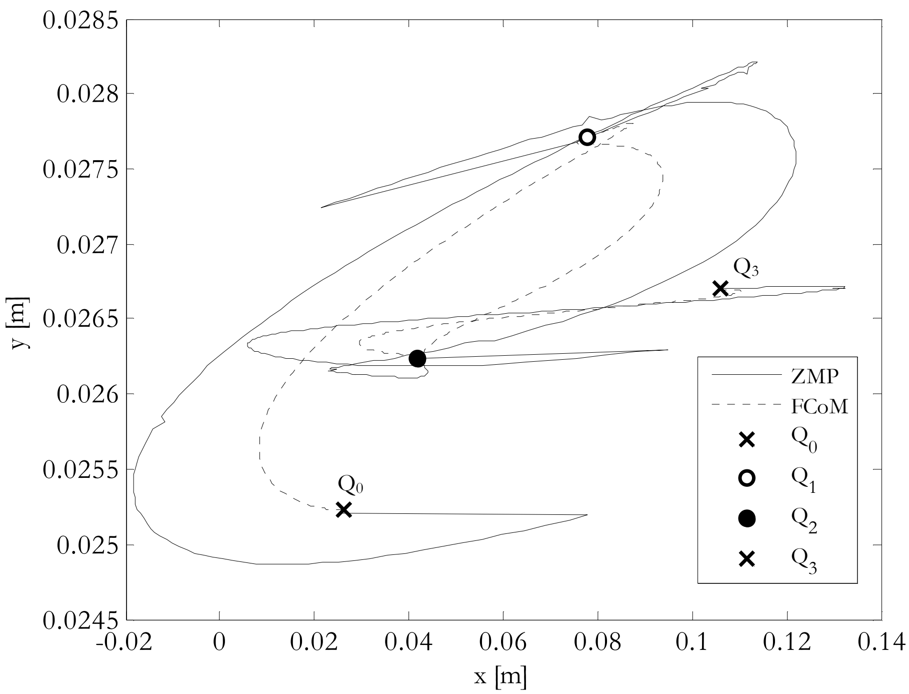

Figure 5.10: The paths of ZMP and FCoM and the position of both at postures Q_0, Q_1, Q_2 and Q_3.

The joint trajectories are depicted in figure 5.11, in which can be seen that there are no discontinuities and each motion is in a admissible range.

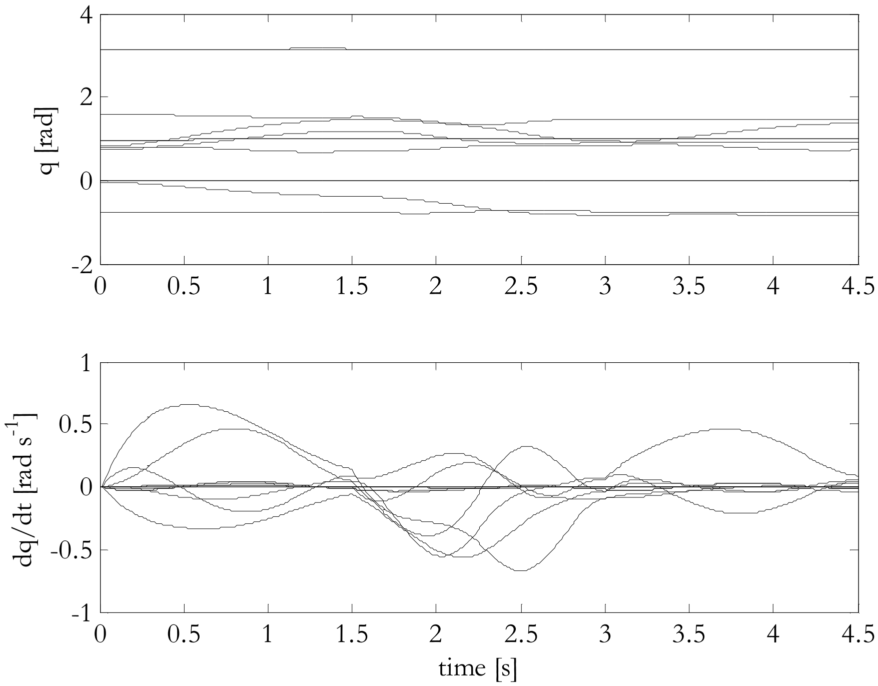

Figure 5.11: Joint motions and velocities.

Mapping

The fact that TUlip is symmetric in the Sagittal plane allows the direct derivation of a right step by mapping the generated left step. The (derivated) right step is depicted in figure 5.12.

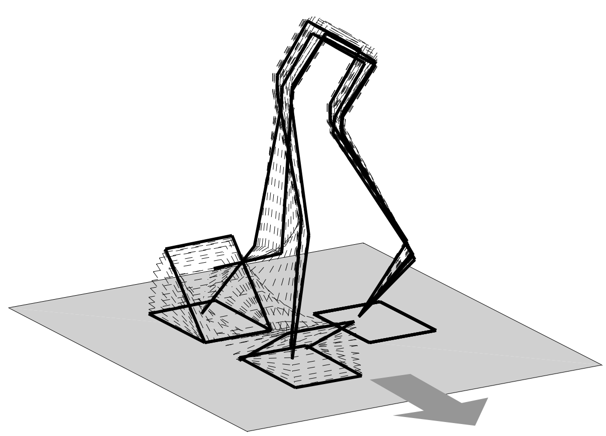

Figure 5.12: A three dimensional view of a right step.
Gray area represents the floor, bold lines: postures Q_0, Q_1, Q_2 and Q_3,
arrow indicates direction of the gait.

5.4 Transition Primitives and Concatenation

The TP that moves the ZMP from the right foot to the left foot (thus after a left step) and its mapping are both necessary to achieve a realistic gait in which left steps alternate with right steps.
A difficulty occurs when the joint motions for this TP are derived with Inverse Differential Kinematics. The $\mathbf{x}_{ref}$ of the end-effector is not a trajectory anymore, because both feet do not move in DS, so:

$$\mathbf{x}_{ref} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \theta_{roll_0} \\ \theta_{pitch_0} \\ \theta_{yaw_0} \end{bmatrix} \neq \mathbf{x}_{ref}(t) \tag{5.5}$$

Another difficulty is that the ZMP should move from one location to the other and remain in the SP while moving to that other location. So the penalty function G_1, as stated in (4.26), should not only be dependent on $\mathbf{q}$ and its time derivatives, but also on time. So the function shown in figure 4.5, that resembles a kind of a 'cup', should move with the ideal position of the ZMP. This is illustrated in figure 5.13. From this figure it can be seen that the ideal path of the ZMP moves from one foot to the other.

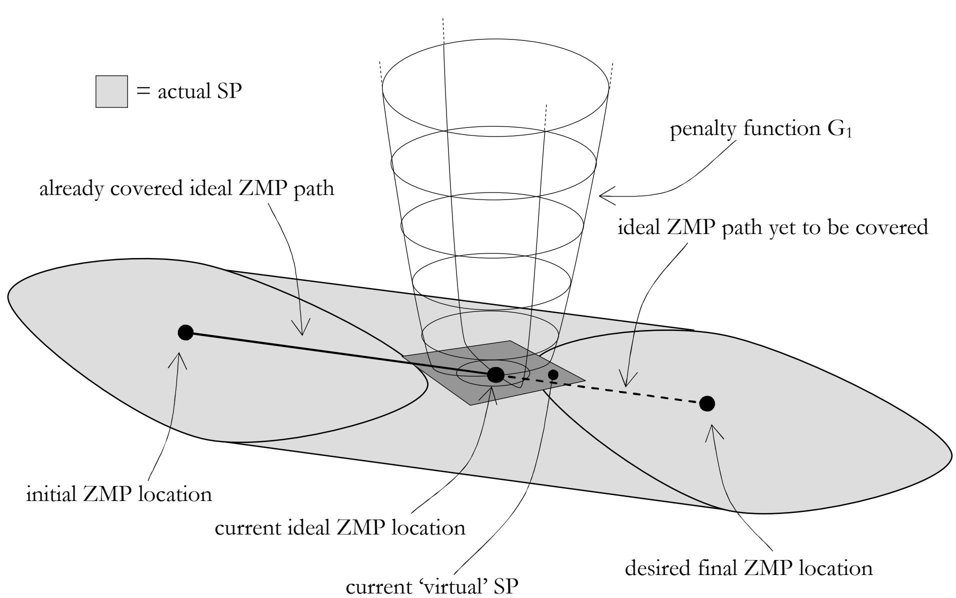

Figure 5.13: Moving constraint for ZMP.

The initial and final points of the ZMP are determined by the FCoM of the initial and final posture. The path in between can be computed e.g. with quintic interpolation. The edges of the now 'virtual' SP, needed for the penalty function G_1, are now dependent on time:

$$\mathbf{SP}_{virtual}(t) = \begin{bmatrix} SP_{x,min}(t) & SP_{x,max}(t) & SP_{y,min}(t) & SP_{y,max}(t) \end{bmatrix}^T \tag{5.6}$$

To follow this ideal path accurately, these edges should be chosen close to this path. The penalty function G_1 can be implemented as part of the code of the algorithm of Appendix B. For convenience it is stated below:

```
G1 = (SPxmax(k-1)-SPxmin(k-1))^2/4/(SPxmax(k-1)-ZMPx)/(ZMPx-SPxmin(k-1)) + …
     (SPymax(k-1)-SPymin(k-1))^2/4/(SPymax(k-1)-ZMPz)/(ZMPz-SPymin(k-1))
```

An outcome of the algorithm, the result is depicted in figure 5.14. As expected, TUlip moves its upper body from above its left foot to above its right foot, since its upper body contains $\frac{2}{3}$ of TUlip's total mass.

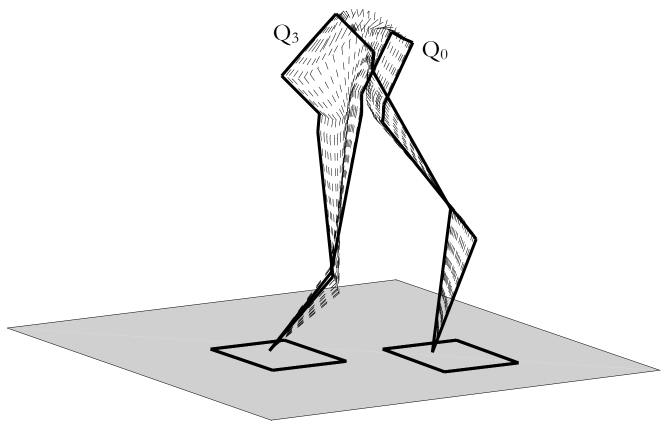

Figure 5.14: Shifting ZMP from left foot to right foot.
Bold lines are Q_3 of a left step en Q_0 of a right step.

The movement of the 'virtual' SP and the computed ZMP path are plotted, in figure 5.15. It can be recognized that the required end-posture (Q_0 of a right step) is not reached, due to the fact that the computed ZMP does not finish at the desired point.

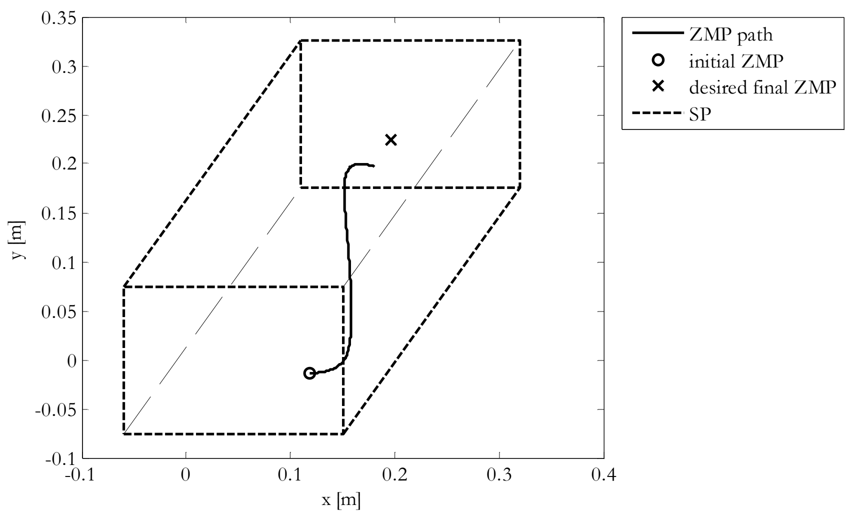

Figure 5.15: Shifting ZMP from left foot to right foot using Inverse Differential Kinematics.

The occurrence of this problem is caused by the number of (fully) actuated DoFs in TUlip. TUlip has 10 DoFs in its legs and six are constraint by the placement of the tip (foot) on the floor. So, there are four DoFs remaining what would imply that the system is still redundant. Because of dependency between joints, which is very likely, the solution is very conservative and thus, very hard to find. To gain some more freedom extra gains are added in the algorithm (see Appendix B), but no solution is found.

5.5 Dynamic Simulation

If the obtained joint motions for a left step are put in a dynamical simulation the step is considered to be stable, see figure 5.16. Due to some insufficience in managing the contact dynamics in this model, the stance foot (right) is lifted a fraction from the floor at multiple time instances, which allows the model to rotate. Due to this the end position of the swing foot is not the desired one but it is slightly diverted which is illustrated in figure 5.17. If these errors would not occur, the result would agree with the result obtained by kinematic simulation.

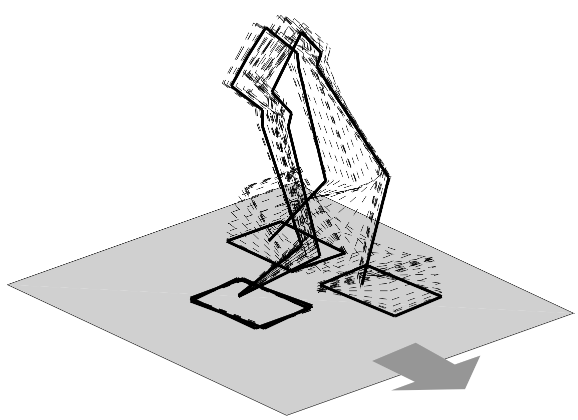

Figure 5.16: TUlip in dynamical simulation.
Grey arrow: direction of the gait.

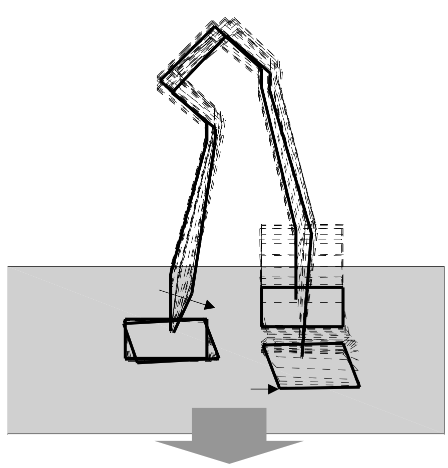

Figure 5.17: TUlip in dynamical simulation, front view: the end position of the swing foot is (slightly) diverted.
Grey arrow: direction of the gait.

Chapter 6

Discussion

6.1 Conclusions

6.1.1 ZMP Criterion

Using the ZMP criterion a biped can achieve a dynamically stable gait. This is the reason why this criterion is widely applied on biped walking. In simulation study using the dynamical model of the TUlip robot, as demonstrated in section 5.1, TUlip achieved a dynamically stable step.
The ZMP criterion can also be used if the robot has to perform other tasks but walking. For example: picking up a glass of water from a table, while bending its knees, can be performed without falling with this stability criterion. Hence, the ZMP is a very practical criterion, though it has its limitations. In order to be stable with this criterion the ZMP has to remain within the SP for *all* time instances. Running and jumping are never stable according to this criterion due to the fact that there is no continuous SP. Another limitation of walking dynamically stable according to the ZMP criterion is that the achievable gait resembles little of a *real* human gait. Humanlike walking is not stable according to the ZMP criterion. Nevertheless, humans don't fall while walking. If the ZMP of a human walk should be monitored, it would certainly leave the SP. Limit Cycle Walking (see section 2.2.2) is a more energy efficient way and it resembles more of humanlike walking. However, Limit Cycle Walking is not developed up till now in such way that it allows the biped (or humanoid) to:

- starts and stops walking,
- walk on stairs or slopes,
- make curves,
- etcetera.

All above can be realized with the ZMP criterion. Hence, in expectation of better stability criteria or other ways to prevent the biped from falling, the ZMP is a very practical criterion to use in biped locomotion and in *posture control*[o] of humanoids.

6.1.2 Walking Primitives Method

Walking Primitives are relatively easy to compute, compared to other methods that compute the joint motions of a whole gait in advance. In addition to this, a step-sequence planning can select and concatenate these WPs during runtime in order to

[o] Posture control represents the correction of the feet and body to maintain stable while achieving (multiple) tasks. For humanoids such a task can be picking up an object from e.g. a table or the floor.

obtain a situation dependent walking pattern. This adaptive behavior is a major advantage of the Walking Primitives Method. When performing a WP the sequence planner 'knows' what the end posture (the final state) of this WP is and can select a proper WP that can be concatenated to the actual WP of the biped. In order to achieve a walking trail in a operational space subject to moving objects, the sequence planner can select the next WP to concatenate in order to maintain stable while avoiding these (moving) objects.
If all final states of the WP are statically stable, the robot can stand still after every WP. This allows the robot to perform an emergency stop without falling (at the end of the actual WP) if there is something that is suddenly in its walking trail.

6.1.3 TUlip

TUlip is especially designed for Limit Cycle Walking. The equivalent CoM is located in the lower region of its upper body, around its "bellybutton. The drivelines of each joint are positioned as high as possible to ensure this location of the CoM. This is very suitable for Limit Cycle Walking because this method uses the mass of the body to 'fall' which ensures the forward movement.
For the ZMP criterion however, it is much more plausible the equivalent CoM is placed closer to the floor. If the CoM is close to the floor, movements of the biped result in less effect on the FCoM and the ZMP, so it costs less effort to keep the ZMP in the SP. The equivalent CoM dependens on CoMs of all individual links and, subsequently, the ZMP strongly depends on the equivalent CoM. TUlip has no actuated ankle joint in the x-direction (see figure 1.1b). In order to control the biped with the ZMP criterion, all joints should actively be actuated to control the biped according the ZMP criterion. In order to get accurate information about the angle and angular velocity of the links, to control the ZMP, the joints should be position controlled. This is also not the case with TUlip, since it is more suitable to torque control due to *series elastic actuation* [19]. Because of springs in the driveline, the bandwidth of the individual joints (plus controllers) is limited (because of the relatively low eigenfrequency of the springs) and, hence; accurate position control of each joint is difficult.

6.2 Recommendations

6.2.1 Time Dependent Inverse Differential Kinematics

In section 5.4 the used penalty function G_1 to constrain the ZMP is time dependent; $G_1 = G_1(t)$. This is up till now not be found in literature, so more research with respect to this kind of constraining is recommended. The Cartesian positions of both feet and a the trajectory of the ZMP can be pre-described, as depicted in figure 6.1.
From the ZMP trajectory a time dependent penalty function can be derived. From these (three) trajectories the Inverse Differential Kinematics can be solved. Maybe with some additional tuning parameters, the area where this penalty function has influence becomes very small, which would allow the ZMP to follow accurately the pre-described path. So a whole gait (thus not only one WP) can be computed in advance.

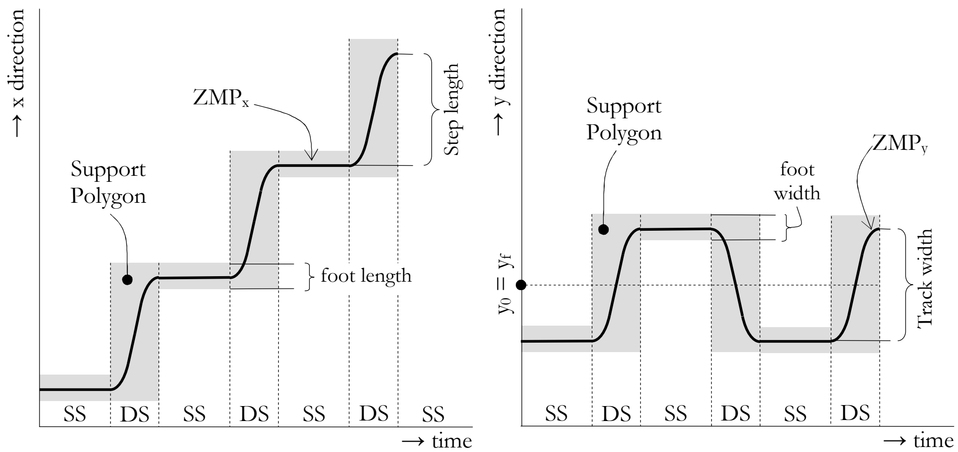

Figure 6.1: Prescribed ZMP path and SP in x (a) and y (b) direction.

6.2.2 Suitable Biped for ZMP Control

From section 6.1.3 it can be easily observed that the biped, that has to be stabilized with the ZMP criterion, has to meet the following demands:

1. The biped should contain at least 6 DoFs per leg:
   - three in the hip: x, y and z rotation,
   - one in the knee: y rotation,
   - two in the ankle: y and x rotation.
2. All joints should be actively actuated and position controlled.
3. The equivalent CoM should be placed as close to the floor as possible.

For TUlip this means: addition of ankle x-rotation motors is a necessity and bi-directional actuation of ankle y-motors is suggested.

Dynamical Model of TUlip

The actual model of Tulips dynamics has been derived analytically. This implies that non-linear equations of motions are (numerically) integrated in time. Also the contact dynamics, such as floor contact (impact as well as release), are analytically derived. These kind of dynamics remain the main difficulties in the design of robust control for bipeds. They may cause the issues described in section 5.5. To overcome such issues, investigation on modeling and analysis of contact dynamics between the bipeds' feet and the floor is suggested:

- In [8], it is proposed to carry out dynamical modeling in the framework of mechanical systems subject to complementary conditions. Constraints [2] that apply while a foot releases the floor [2] and Coulomb friction can be represented in this way.
- The impact conditions in the current model are infinitely stiff. It would be interesting to investigate the influence of elastic contact conditions (including damping) on stability of the gait or on stability of the bipeds' posture control.

Because the model is derived using the *Lagrange-Euler* formalism [22] in closed-form, its execution is very time consuming. A numerical model can usually be executed faster and for qualitative analysis of a certain gait a numerical representation of these dynamics can be sufficient.

One example is dynamical modeling using the *Open Dynamics Package*. This package derives the dynamics using the *Newton-Euler* formalism [22]. Because of this, the modeling is simple and the simulation is faster but drift in the solution can occur.

Appendix A

Denavit-Hartenberg parameters for TUlip

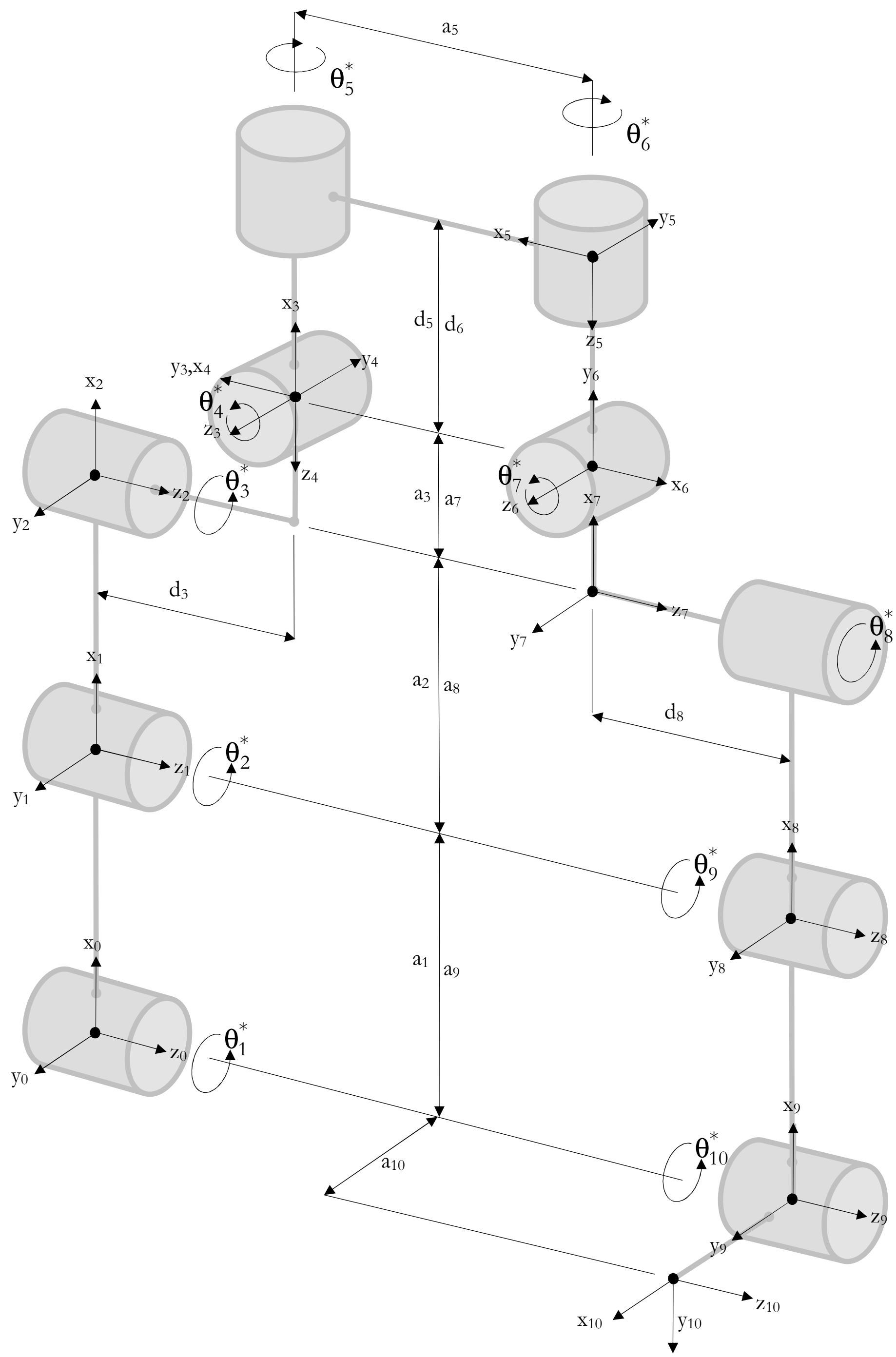

Figure A.1: Schematic view of kinematic model of TUlip with all active actuations.

Table A.1: Denavit-Hartenberg parameters belonging to figure A.1.

l_i	a_i	d_i	α_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*
3	a_3	d_3	$-\pi/2$	θ_3^*
4	0	0	$-\pi/2$	θ_4^*
5	$-a_5$	$-d_5$	0	θ_5^*
6	0	d_6	$-\pi/2$	θ_6^*
7	$-a_7$	0	$\pi/2$	θ_7^*
8	$-a_8$	d_8	0	θ_8^*
9	$-a_9$	0	0	θ_9^*
10	a_{10}	0	0	θ_{10}^*

* this is a driven joint

Appendix B

Inverse Differential Kinematics Algorithm

Algorithm B.1: Inverse Differential Kinematics

```
n      = length of q
kend   = last integration step (regularly the length of vector t)
Q      = end posture

for k = 1:kend

    %% Reference trajectory of the tip:
    Xref        = [x(k)    y(k)    z(k)    θr(k)    θp(k)    θy(k)]^T
    Xrefd       = [xd(k)   yd(k)   zd(k)   θrd(k)   θpd(k)   θyd(k)]^T
    Xrefdd      = [xdd(k)  ydd(k)  zdd(k)  θrdd(k)  θpdd(k)  θydd(k)]^T

    %% Jacobians:
    J(k)        = compute jacobian(q(k-1));
    Jd          = (J(k) -J(k-1))/Δt;
    J#          = compute pseudo inverse(J)

    I           = compute identity matrix(n)
    T           = compute homogeneous transformation matrix(q(k-1));
    RPY         = compute roll pitch yaw from homogeneous transformation …
                  matrix(T)
    Exyz        = [x(k)  y(k)  z(k)]^T  - [T14 T24 T34]^T
    Erpy        = [θr(k) θp(k) θy(k)]^T - [RPY1 RPY2 RPY3]^T

    %% Operational space error:
    E           = [Exyz,Erpy]
    Ed          = Xrefd(k) - J(k)*qd(k-1)

    [ZMPx,ZMPy]= compute ZMP(q(k-1); qd(k-1); qdd(k-1))

    %% Penalty function:
    G1          = (SPxmax - SPxmin)^2/4/(SPxmax - ZMPx)/(ZMPx - SPxmin) + …
                  (SPymax - Spymin)^2/4/(SPymax - ZMPy)/(ZMPy - SPymin)
    G2          = compute sum of absolute values of every instance (Q-q(k))
    G3          = compute sum of absolute values of every instance (qd(k-1))
    G(k)        = k1*G1 + k2*G2 + k3*G3
    ΔG          = G(k)      - G(k-1)
    Δq          = q(k-1)    - q(k-2)
    Δqd         = qd(k-1)   - qd(k-2)
    Δqdd        = qdd(k-1) - qdd(k-2)
    qαdd        = -(kα1*ΔG/Δq + kα2*ΔG/Δqd + kα3*ΔG/Δqdd)

    %% Inverse Differential kinematics:
    qdd(k)      = J#*(Xrefdd - Jd*qd(k-1) + Kd*Ed + Kp*E) + (I - J#*J)* qαdd

    %% Euler integration:
    qd(k)       = qd(k-1) + Δt*qdd(k)
    q(k)        = q(k-1)  + Δt*qd(k)
end
```

```
Legend:
%% text            : comment
k                  : current integration step
a(k)               : a at integration step k
ad                 : first time derivative of a
add                : second time derivative of a
a                  : known a priori
a = function(b;c): this function computes the output a as a function of b
                     and c
```

Bibliography

[1] Biagiotti, Melchiorri, *'Trajectory Planning for Automatic Machines and Robots'*, ISBN: 978-3-540-85628-3, p.43.

[2] Ten Dam, '*Unilaterally Constrained Dynamical Systems'*, 1997, http://irs.ub.rug.nl/ppn/159407869

[3] Denk, Schmidt, *'Synthesis of Walking Primitive Databases for Biped Robots in 3D-Environments'*, Int.l Conf. on Robotics and Automation (ICRA2003), Taipei, Taiwan, 2003, p. 1343–1349.

[4] Denk, Schmidt, '*Synthesis of a Walking Primitive Database for a Humanoid Robot using Optimal Control Techniques',* In. Conf. on Humanoid Robots, Tokyo, Japan, 2001, p. 319-326

[5] Ha, Choi, '*An Effective Trajectory Generation Method for Bipedal Walking',* Science Direct, Robotics and Autonomous Systems 55, 2007, p. 95-810.

[6] Helbo, '*Modelling and Control of a Biped Robot*', Dep. of Control Engineering, 8th Semester, Aalborg University, Denmark, 2007, p. 5-10.

[7] Huang, Yokoi, Kajita, Kaneko, Arai, Koyachi, et al., *'Planning Walking Patterns for a Biped Robot'*, IEEE Trans. on Robotics and Automation, 2001, p. 280-289.

[8] Hurmuzlu, Génot, Brogliato, '*Modeling, stability and control of biped robots - a general framework'*, Science Direct, 1999, p. 1647-1664

[9] Kajita, Kanehiro, Kaneko, Yokoi, Hirukawa, *'The 3D Linear Inverted Pendulum Mode: A Simple Modeling for a Biped Walking Pattern Generation',* Int. Conf. on Intelligent Robots and Systems, Maui, USA, 2001, p. 239-246.

[10] Kuffner, Kagami, Nishiwaki, Inaba, Inoue, *'Dynamically-stable Motion Planning for Humanoid Robots'*, Robots vol. 12 No. 1, 2002, p. 105-118.

[11] Kuffner, Kagami, Nishiwaki, Inaba, Inoue, *'Motion Planning for Humanoid Robots'*, 11th Int. Symp. of Robotics Research, 2003, p. 1-10.

[12] Nakamura, *'Advanced Robotics, Redundancy and Optimization'*, ISBN: 0-201-15198-7, Addison-Wesley Publishing Comp., 1991, p.125-170.

[13] Siciliano, Khatib (Eds.), *'Springer Handbook of Robotics'*, ISBN: 978-3-540-23957-4, Springer-Verlag Berlin Heidelberg, 2008, p. 371-378.

[14] Siciliano, Sciavicco, Villani, Oriolo, *'Robotics, Modelling, Planning and Control'*, ISBN 978-1-84628-641-4, Springer-Verlag London Limited, 2009.

[15] Spong, Hutchinson, Vidyasagar, '*Robot Modeling and Control'*, ISBN: 10-0-47-64990-2, John Wiley & Sons Inc., 2006.

[16] Vukobratović, Borovac, Potkonjak, '*Towards a Unified Understanding of Basic Notions and Terms in Humanoid Robotics'*, Robotica (2007), Vol. 25, p. 88-89.

[17] Vukobratović, Borovac, '*Zero-Moment Point - Thirty Five Years of Its Life'*, Int. J. Humanoid Robotics, Vol. 1 No. 1, 2004, p. 161,162.

[18] Wisse, Hobbelen, '*Controlling the Walking Speed in Limit Cycle Walking'*, the Int. Journal of Robotics Research, 2008, p. 989-1005.

[19] Wisse, Hobbelen, de Boer, *'System Overview of Bipedal Robots Flame and TUlip: Tailor-made for Limit Cycle Walking'*, Int. Conf. on Intelligent Robots and Systems, 2008, p. 2488.

[20] Wollher, Buss, *'Posture Modification for Biped Humanoid Robots Based on Jacobian Method'*, Int. Conf. on Intelligent Robots and Systems, Japan, 2004, p.124-129.

[21] Wollher, Sobotka, Buss, *'A Jacobian Method for Online Modification of Precalculated Gait Trajectories'*, 6th Int. Conf. on Climbing and Walking Robots, Italy, 2003, p.435-442.

[22] Wouw, *'Multibody Dynamics, lecture notes'*, University of Technology, Eindhoven, 2005.