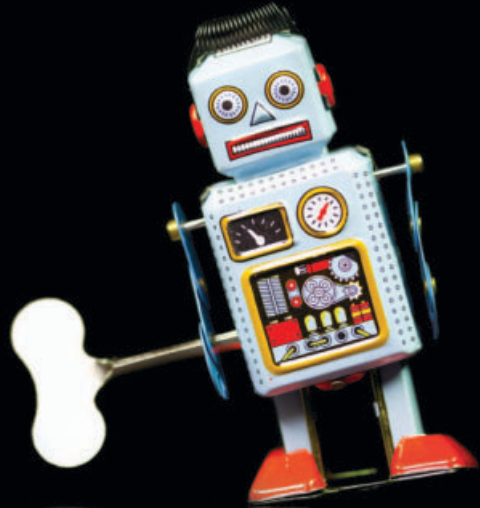


FOOT PLACEMENT

in robotic bipedal locomotion

Tomas de Boer



Foot placement

in robotic bipedal locomotion

Foot placement

in robotic bipedal locomotion

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus, prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties
in het openbaar te verdedigen op dinsdag 13 maart 2012 om 15:00 uur
door

Tomas de Boer

werktuigkundig ingenieur
geboren te Kortenhoef

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. F.C.T. van der Helm

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr. F.C.T. van der Helm	Technische Universiteit Delft, promotor
Dr. ir. M. Wisse	Technische Universiteit Delft, copromotor
Prof. dr. H. Nijmeijer	Technische Universiteit Eindhoven
Prof. dr. J.H. van Dieën	Vrije Universiteit Amsterdam
Prof. dr. ir. H. van der Kooij	Technische Universiteit Delft
Dr. ir. A.L. Hof	Rijksuniversiteit Groningen
Dr. J.E. Pratt	Institute for Human and Machine Cognition, USA
Prof. dr. ir. P.P. Jonker	Technische Universiteit Delft, reservelid

Dit onderzoek is financieel mogelijk gemaakt door de Europese Commissie onder Framework Programme 6 (contract FP6-2005-IST-61-045301-STP).



A digital copy of this thesis can be downloaded from <http://repository.tudelft.nl>.

ISBN 978-94-6169-211-5

DOI 10.4233/uuid:795fa8f5-84a0-4673-810c-a8265e29791c

Contents in brief

Summary	xiii
1. Introduction	1
2. Background	11
3. Capturability-based analysis and control of legged locomotion: theory and application to three simple gait models	21
4. Foot placement control: step location and step time as a function of the desired walking gait	63
5. Mechanical analysis of the preferred strategy selection in human stumble recovery.	83
6. Robot prototype: design	101
7. Robot prototype: capturability based control	123
8. Robot prototype: performance	163
9. Discussion, conclusions and future directions	177
References	195
Acknowledgements	211
Samenvatting	213
Curriculum vitae	217

Contents

Summary	xiii
1. Introduction	1
1.1 Motivation	2
1.2 Problem statement	3
1.3 Research focus	5
1.4 Goal	7
1.5 Approach	8
1.6 Thesis outline	9
2. Background	11
2.1 Introduction	12
2.2 Zero Moment Point	12
2.3 Limit Cycle Walking	15
2.4 Hybrid Zero Dynamics	16
2.5 Decoupled control	17
2.6 Conclusion	20
3. Capturability-based analysis and control of legged locomotion: theory and application to three simple gait models	21
3.1 Introduction	22
3.2 Background	23
3.3 Capturability framework	26
3.4 Three simple gait models	28
3.5 3D-LIPM with point foot	29
3.5.1 Equations of motion	30
3.5.2 Allowable control inputs	31
3.5.3 Dimensional analysis	32
3.5.4 Instantaneous capture point	32
3.5.5 Instantaneous capture point dynamics	34
3.5.6 Capturability	35
3.5.7 Capture regions	37
3.6 3D-LIPM with finite-sized foot	39
3.6.1 Equations of motion	39
3.6.2 Allowable control inputs	41
3.6.3 Dimensional analysis	41
3.6.4 Equivalent constant center of pressure	42

3.6.5	Capturability	43
3.6.6	Capture regions	45
3.7	3D-LIPM with finite-sized foot and reaction mass	46
3.7.1	Equations of motion	47
3.7.2	Allowable control inputs	49
3.7.3	Dimensional analysis	49
3.7.4	Effect of the hip torque profile	50
3.7.5	Capturability	51
3.7.6	Capture regions	54
3.8	Capturability comparison	55
3.9	Discussion	57
3.9.1	Simple models	57
3.9.2	Robustness metrics	59
3.9.3	More complex models	59
3.9.4	Capturability for a specific control system	59
3.9.5	Capturability and Viability	60
3.9.6	Future work	60
3.10	Conclusion	61
4.	Foot placement control: step location and step time as a function of the desired walking gait	63
4.1	Introduction	64
4.2	Model description	65
4.2.1	Equations of motion	66
4.2.2	Stepping constraints	68
4.3	Walking gait	69
4.4	Control problem	69
4.5	Dynamic Foot Placement controller	73
4.5.1	0-step strategy	73
4.5.2	1-step strategy	73
4.5.3	2-step strategy	75
4.5.4	($N > 2$)-step strategy	76
4.6	Control performance	76
4.6.1	Number of steps	76
4.6.2	Time response	78
4.7	Discussion	79
4.8	Conclusion	81

5. Mechanical analysis of the preferred strategy selection in human stumble recovery.	83
5.1 Introduction	84
5.2 Models and simulation methods	85
5.2.1 The walking models	87
5.2.2 The induced stumble	89
5.2.3 The recovery	90
5.2.4 The hypothesized cost of recovery	92
5.3 Results	93
5.3.1 Recovery of the simple model	93
5.3.2 Recovery of the complex models	94
5.3.3 Increased obstruction duration	94
5.3.4 Multiple-step recovery	94
5.3.5 Hypothesized cost measures	96
5.4 Discussion	97
5.5 Conclusion	100
6. Robot prototype: design	101
6.1 Introduction	102
6.2 Robot evolution	102
6.3 Mechanics	105
6.3.1 Actuation	106
6.3.2 Actuator implementation in robot design	111
6.4 Electronics	116
6.4.1 Computing	116
6.4.2 Sensors	116
6.4.3 Power supply	117
6.5 Software	117
6.6 Conclusion	120
7. Robot prototype: capturability based control	123
7.1 Introduction	124
7.2 Background	125
7.3 Description of M2V2 Robot	127
7.4 Simulation environment	128
7.5 Control tasks	130
7.5.1 Balancing	130
7.5.2 Walking	130
7.6 Control concepts	130

7.6.1	Capturability-based control using an approximate model	130
7.6.2	Force control	131
7.6.3	Virtual model control	132
7.7	Controller implementation	132
7.7.1	State machine	134
7.7.2	Capture region calculator	135
7.7.3	Desired footstep calculator	138
7.7.4	Swing sub-controller	139
7.7.5	Stance sub-controller	142
7.8	Results	149
7.8.1	Balancing task	149
7.8.2	Walking task	149
7.9	Discussion and future work	153
7.9.1	Using simple models for complex robots	153
7.9.2	1-step versus N -step capture regions	155
7.9.3	Capturability margin	155
7.9.4	Estimation of center of mass position and velocity	156
7.9.5	Uneven ground	156
7.9.6	Controlling velocity versus coming to a stop	157
7.9.7	Cross-over steps	157
7.9.8	Virtual toe points and center of pressure	157
7.9.9	Foot placement speed and accuracy	158
7.9.10	Application to other robots	158
7.10	Conclusion	158
8.	Robot prototype: performance	163
8.1	Introduction	164
8.2	Two-legged balancing	164
8.3	Single-legged balancing	165
8.4	Foot placement	167
8.5	Push recovery by stepping	169
8.6	Discussion	169
8.6.1	Force controllability	171
8.6.2	Foot placement	171
8.6.3	Mechanics	172
8.6.4	Electronics	174
8.6.5	Software	175
8.7	Acknowledgements	175

9. Discussion, conclusions and future directions	177
9.1 Introduction	178
9.2 Recapitulation	178
9.2.1 Foot placement and robustness	178
9.2.2 Foot placement and versatility	179
9.2.3 Foot placement and energy-efficiency	180
9.3 General discussion	180
9.3.1 On the effectiveness of foot placement	180
9.3.2 On the availability of foot placement	181
9.3.3 On the energetic cost of foot placement	182
9.3.4 On the accuracy of foot placement	183
9.4 Conclusions	189
9.4.1 Research questions	189
9.4.2 General conclusions	190
9.5 Future directions	191
9.5.1 On the mechanics of walking	191
9.5.2 On the control of walking	192
References	195
Acknowledgements	211
Samenvatting	213
Curriculum vitae	217

Summary

Human walking is remarkably *robust*, *versatile* and *energy-efficient*: humans have the ability to handle large unexpected disturbances, perform a wide variety of gaits and consume little energy. A bipedal walking robot that performs well on all of these aspects has not yet been developed. Some robots are versatile, others are energy-efficient, and none are robust since all robots often lose balance. This lack of performance impedes their applicability in daily life. Also, it indicates that the fundamental principles of walking are not adequately understood. The goal of this thesis is to increase the understanding of the mechanics and control of bipedal locomotion and thereby increase the performance of robotic bipedal locomotion. This increased understanding will also be useful for the development of robotic devices that can help people with a decreased ambulatory ability or that can augment the performance of able-bodied persons.

Bipedal locomotion is in essence about the ability to maintain control over the position and velocity of the body's center of mass (CoM). This requires controlling the forces that act on the CoM through the foot. The contact forces between the foot and the ground can be manipulated to some extent through ankle torques or upper body motions, but are mostly determined by the location of the foot relative to the CoM. The limited influence that ankle torques and upper body motions have on the contact forces and consequently on the CoM is best illustrated when one tries to remain balanced on one foot without taking a step. When slightly perturbed, balance is quickly lost and a step must be taken to prevent a fall. This demonstrates that balance control in walking relies on adequate control of foot placement (i.e., the location and timing of a step), which therefore is our main focus in the control of robotic gait.

The focus on foot placement control is different from other popular control approaches in robotics. In ZMP-based control, one typically adjusts the robot's state to achieve a predefined foot placement. In Limit Cycle Walking, passive system dynamics mostly determine foot placement. This thesis presents foot placement strategies that can be adapted both in step time and step location, are an explicit function between the initial robot state and the desired future robot state, and are computationally relatively inexpensive to allow for real-time application on the robot. The contributions of this thesis to bipedal walking research are: a theoretical framework, simulation studies, and prototype experiments. These contributions provide insight in how foot placement control can improve the robustness, versatility and energy-efficiency of bipedal gait.

Regarding robustness, this thesis introduces the theoretical framework of capturability to analyze or synthesize actions that can prevent a fall. Fall avoidance is analyzed by considering N -step capturability: the system's ability to eventually come to a stop without falling by taking N or fewer steps, given its dynamics and actuation limits. Low-dimensional gait models are used to approximate capturability of complex systems. It is shown how foot placement, ankle torques and upper body motions affect the CoM motion and contribute to N -step capturability. N -step capture regions can be projected on the floor: these define where the system can step to remain capturable. The size of these regions can be used as a robustness metric.

Regarding versatility, this thesis derives foot placement strategies that enable the system to evolve from the initial state to a desired future state in a minimal number of steps. Simulations on simple gait models demonstrate how these foot placement strategies can be used to change walking speed or walking direction.

Regarding energy-efficiency, we learn that simple gait models demonstrate human-like foot placement strategies in response to a stumble when optimizing for either one of the following cost measures for foot placement: peak torque, power, impulse, and torque divided by time. For robotic control, these results indicate that actuator limitations should be taken into account in the execution and planning of foot placement strategies.

Regarding robot experiments, we integrate the concepts from the capturability framework into the control of a robot. The low-dimensional gait models are shown to be useful for the robust control of a complex robot. The model takes only the CoM dynamics with respect to the center of pressure (CoP) into account. The application of this model together with force-based control strategies lead to robust robot behavior: upright postural balance is maintained when the robot is pushed and one of the feet is placed on a moving platform. Successful application is also shown for single legged balancing with compensatory stepping to regain balance after a push and (simulated) walking.

The main conclusion is that analyzing walking control as a combination of decoupled and low dimensional control tasks allows us to derive simple and useful control heuristics for the control of a complex bipedal robot. We find that the key control task is foot placement, which mostly determines the system's CoM motion by defining possible CoP locations. We can approximate the set of possible foot placement strategies that will not lead to a fall. This set specifies the bounds to which foot placement strategies can be adjusted to achieve more versatile or energy-efficient behavior.

1

Introduction

1.1 Motivation

This thesis investigates the control and mechanics of walking, aiming to advance the state-of-the-art in bipedal walking robots. This research intends to contribute to the creation of human-like robots that can one day support humans in their daily life. A common motivation for biped robot research is that the human-like appearance of these robots is best suited to interact with people and the environments that are specifically designed for the human morphology.

The motivation for this research goes beyond the advancement in the field of bipedal robots, because its results are also useful for research in the field of rehabilitation. Development of a human-like walking robot requires fundamental insight in the mechanics and control of human locomotion. This insight is crucial for the development of better rehabilitation devices and diagnoses to solve problems that people may have with walking. Robot-like devices that can replace part of a missing leg have already been developed (Figure 1.1a). However, more insight on human walking is required because the functionality of such prosthetic devices is still lacking compared to that of their biological counterpart: a disabled person using such a device requires significantly more effort to perform the same ambulatory tasks as an able-bodied person.

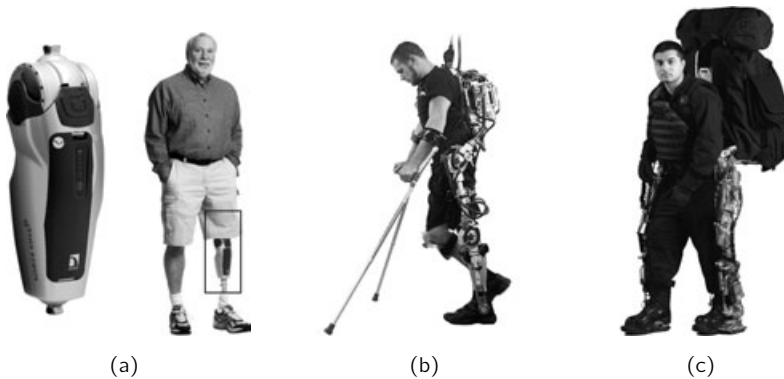


Figure 1.1: Three examples of assistive robotic devices for human walking. (a) 'PowerKnee', by Össur (Össur, 2011). A commercially available powered knee prosthesis that can aid in flexing and stretching the knee during walking. (b) 'Mina', by the Institute for Human and Machine Cognition (IHMC) (Neuhaus et al., 2011). A prototype of a powered exoskeleton to provide up-right mobility to paraplegic users. An onboard computer controls the motors to move in a walking fashion while the user provides balance with a pair of forearm crutches. (c) 'BLEEX', by University of California, Berkeley (Kazerooni et al., 2005). A prototype of a powered exoskeleton developed with the aim to augment the strength and endurance of an able-bodied wearer.

Recently, a growing number of advanced orthoses (Dollar and Herr, 2008), or exoskeletons, are being developed. These “wearable robots” fit closely to the body and work in concert with the operator’s movement. These devices use hardware and control schemes derived from bipedal robot research to assist a person with a decreased ambulatory ability (Figure 1.1b) or augment the performance of an able-bodied wearer (Figure 1.1c). Just as for bipedal robots, greater understanding of human locomotion, morphology and control is required before these devices can be made robust, efficient, safe and thereby truly functional.

1.2 Problem statement

The walking performance of robots can be evaluated on three important aspects:

- *robustness*, i.e., the ability to handle large unexpected disturbances;
- *versatility*, i.e., the ability to perform a range of different gaits;
- *energy-efficiency*, i.e., the ability to consume little energy.

After about 40 years of research on walking robots, there is no bipedal robot that performs well on all of these aspects. The human outperforms its robotic counterpart by far. Figure 1.2 illustrates the performance of current bipedal robots relative to the human. One can see that a robot performs typically well on only one of three aspects. There even seems to exist a clear trade-off. For example, passive limit cycle walkers are energy-efficient but are neither versatile nor robust. Powered limit cycle walkers that incorporate passive-dynamic concepts into their design, such as Denise (Wisse et al., 2007) or Flame (Hobbelen and De Boer, 2008), quickly become less energy-efficient for a relative small gain in robustness or versatility. Versatile robots that can negotiate stairs, make turns and change walking speed, such as Asimo (Honda, 2011), are typically neither energy-efficient nor capable of handling unexpected disturbances. And a robot such as Petman (Boston Dynamics, 2011) is robust, but not energy-efficient since it relies on an a several horse-power engine to drive an off-board hydraulic pump.

Especially the low level of robustness of bipedal robots (i.e., they fall often) impedes their applicability in everyday environments. The poor robotic walking performance indicates that the fundamental principles of human walking are not adequately understood. It is simply not true that creating a bipedal walking robot which performs well in terms of versatility, energy-efficiency and robustness is only a matter of integration of existing technology or knowledge.

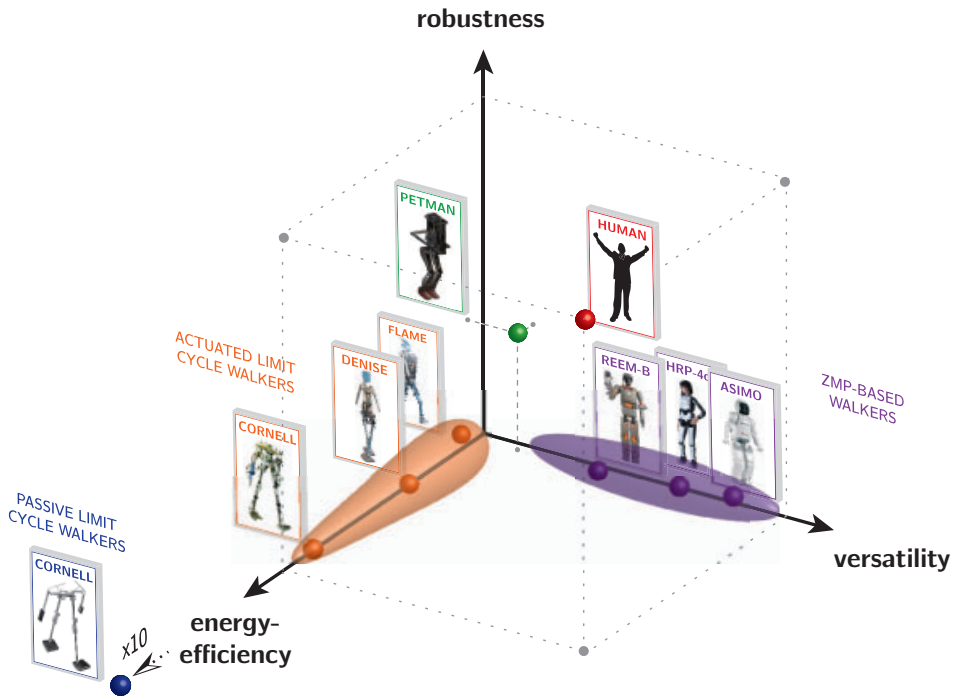


Figure 1.2: A schematic representation of the performance of various bipedal robots (considered to be representative for a larger group of robots). The performance of the robot is evaluated in terms of energy-efficiency, versatility and robustness, relative to the human. More information on the mentioned robots and their control methodology can be found in Chapter 2. As a measure for energy-efficiency we use the specific mechanical cost of transport (as reported in Collins et al. (2005)). Robustness is measured by the robot's demonstrated ability to withstand unexpected perturbations in different magnitude and directions. Versatility gives an indication of the robot's ability to change walking velocity, change walking direction, walk sideways or walk on slopes or stairs. Due to the lack of robot data and the not well-defined performance measures, this illustration is subjective and has no other intention but to give an impression of the current robot performance.

1.3 Research focus

There is currently no bipedal robot that is simultaneously robust, versatile and energy-efficient because of the following two reasons.

Firstly, controlling walking can be considered difficult, since the dynamics of walking are non-linear and high-dimensional. Also, the dynamics are a hybrid between continuous dynamics during the single support phase and discrete collision dynamics at foot impact or locking of the knee. Moreover, the walking system is essentially unstable since it is most of the time supported by only one leg. Stabilizing this unstable system requires interaction with the ground through the feet, but the feet can only push on the ground, not pull. Thus, the control of walking is difficult not only because the dynamics are complex but also because the control authority over the system is limited.

Secondly, the walking performance is not only determined by the control of the system, but also by the mechanics. All the physical properties of the bipedal system (e.g., geometry, stiffness, mass, inertia) and its limitations (e.g., finite torque output or sensing resolution) influence the walking performance. Thus, besides control issues, there are also hardware issues to consider.

To overcome the above-mentioned problems, we look at the fundamental principles of walking. This reduces dimensionality of the walking problem and leads to valuable insight. In essence, walking is about progressing to a desired location while preventing a fall. A fall obviously is potentially very harmful for the system. Preventing a fall is related to the ability to maintain control over the global position and velocity of the system, represented by the position and velocity of the system's center of mass (CoM). The CoM dynamics are a direct result of the gravitational force and the contact forces with the environment that act on the CoM. The relation between the CoM dynamics and the contact forces with the ground during walking is schematically shown in Figure 1.3.

We focus on three basic control strategies that can be used to manipulate the interaction forces with the ground. These control strategies involve the use of ankle torques, angular momentum around the CoM or a step. The way these strategies influence the interaction forces with the ground to control the CoM motion is illustrated in Figure 1.4 for the example task of single-legged upright balancing. The amount of control authority that these strategies have over the CoM are dependent on many system parameters such as foot size, peak joint torque, joint range of motion, mass distribution, etc. A thorough analysis on this matter follows in Chapter 3. However, when performing the balancing tasks as illustrated in Figure 1.4, one easily

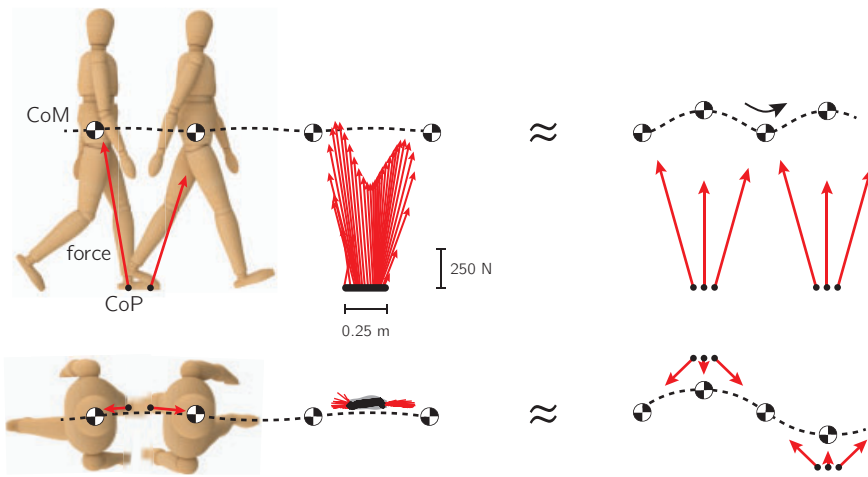


Figure 1.3: Walking is in essence the result of the forces that act on the center of mass (CoM). This figure illustrates the evolution of the center of mass and the ground reaction force vector (GRF) during walking (side and overhead view). The GRF is the net force vector of the forces that the ground applies on the foot. Its point of application is the center of pressure (CoP). The GRF data is obtained during a single walking trial at approximately 1 ms^{-1} .

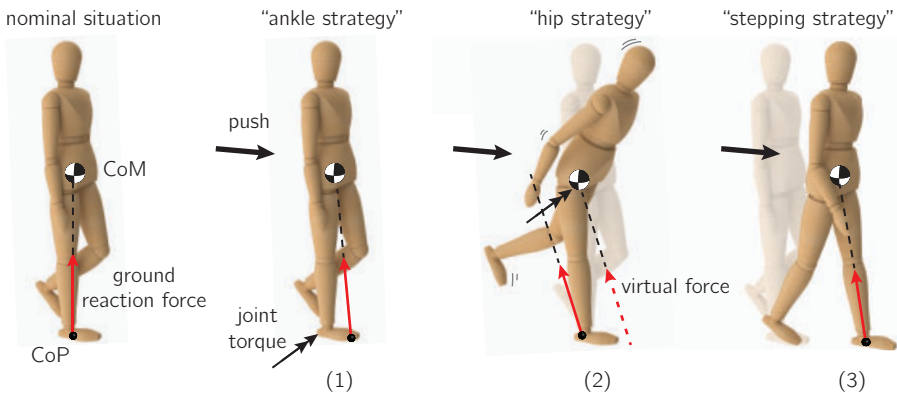


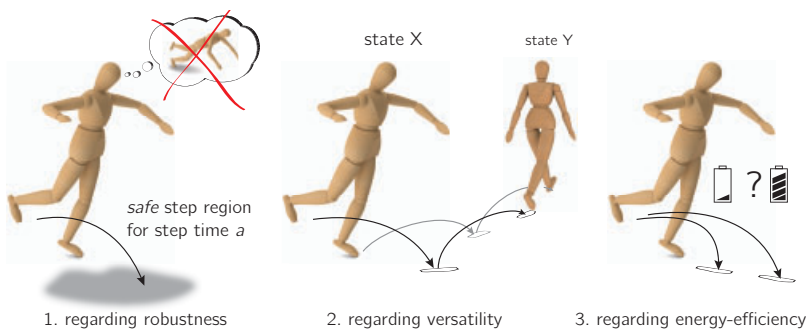
Figure 1.4: Three basic balancing strategies (Horak and Nashner, 1986; Maki and Mclroy, 1997) used during bipedal locomotion, illustrated during the task of single-legged balancing. The balance strategies manipulate the ground reaction force vector (GRF) to control the horizontal center of mass (CoM) motion. For upright balance, the CoM is on average above the point of application of the GRF, i.e., the center of pressure (CoP). (1) Ankle torques manipulate the CoP location. The GRF still acts along the line between the CoP and CoM. (2) Upper-body motions (e.g., through hip torques) change the direction of the GRF, generating torque around the CoM and resulting in angular momentum around the CoM. The equivalence of this is a virtual GRF that passes through the CoM and acts at the Centroidal Moment Pivot point (Popovic et al., 2005). As a result, a forward lunge of the trunk results in a backward motion of the CoM with respect to the foot. (3) A step changes the base of support and offers new candidate CoP locations.

gets an indication of the control authority of each strategy. When standing on one foot, try to recover from a small push by taking a step. Now try to recover by using only ankle torques. Finally, try to recover using only upper body motions (by disabling the ankle strategy by standing only on the ball of the foot). The ability to control the CoM motion decreases for each subsequent balancing task. Though ankle torques and angular momentum around the CoM can manipulate the interaction forces to some extent, the interaction forces are to a large extent simply determined by the position of the foot on the ground relative to the CoM. Consequently, the evolution of the CoM over time is mostly dictated by the location and the timing of a step. Or, in other words, balance control during walking is dictated by foot placement. So to prevent a fall, adequate foot placement is essential and therefore the main topic of this research.

1.4 Goal

The goal of this thesis is to improve the performance of robotic bipedal locomotion in terms of robustness, versatility and energy-efficiency, by increasing the understanding of the mechanics and control of foot placement. Since versatility and energy-efficiency are only relevant when the robot can prevent a fall, a strong focus lies on the robustness of gait. This thesis aims to achieve this goal by answering the following three research questions (illustrated in the figures below).

1. Regarding robustness: how can we determine where and when the foot should be placed to prevent a fall?
2. Regarding versatility: how can we determine where and when the foot should be placed to enable the system to evolve from its initial state to any desired future state?
3. Regarding energy-efficiency: how do actuator limitations influence foot placement strategies?



1.5 Approach

Since foot placement is such an essential part of walking, control of foot placement has often been addressed in research on bipedal robots (see Chapter 2 for a thorough review).

However, generally, foot placement is interpreted inadequately. For 'Zero Moment Point'-based control (Vukobratovic and Borovac, 2004), foot placement locations are predefined and the robot's configuration is adapted to achieve these locations. Forcefully applying such precomputed foot placement locations constrains the robot's motion significantly and limits its robustness against unexpected disturbances. For Limit Cycle Walking (Hobbelen and Wisse, 2007b), foot placement is controlled implicitly and mostly the result of passive swing-leg dynamics with only low levels of control. Though the resulting foot placement is adapted to the robot's state and can even reject small disturbances (Wisse et al., 2005a), these implicit strategies do not indicate where and when to place the foot in case of larger disturbances.

To improve the robot's performance, foot placement control should be *adaptive* and *explicit*. Adaptive, so that the time and location of the step can be adjusted. Explicit, so that foot placement is a function of the current robot state and the desired future system state. We contend that adaptive and explicit foot placement strategies are the only way to remove the apparent trade-off between robustness, versatility and energy-efficiency in robotic bipedal locomotion.

To derive control strategies for foot placement, simple gait models will be used since they reduce the dimensionality of the problem. As demonstrated in the previous sections, looking only at the CoM dynamics relative to the CoP can be an insightful approach to study the fundamentals of balance control.

As mentioned earlier, walking is not only a matter of control but also a matter of mechanics. To gain insight in both these aspects of walking, a new humanoid robot named 'TULip' was developed. By trying to re-invent walking on a robot, insight is gained in which problems are most relevant in walking and which solutions are most effective. This may offer candidate hypotheses on the mechanics and control of human locomotion, since human-like robots obey the same laws of physics as humans do. Similarly, hypotheses on human walking can be made more plausible when shown effective on a real robot.

This thesis builds upon the work of other researchers and merges and extends different approaches on bipedal walking. The work is influenced by the Limit Cycle Walking approach (Hobbelen and Wisse, 2007b), as this approach successfully demonstrates that minor control actions on a step-to-step basis can already be sufficient to stabilize

walking. Also, the research in this thesis adopts a view on walking that is inspired by the work of Raibert (1986) and Pratt (2000b), where the walking problem is viewed as a collection of decoupled tasks, with an essential task being the control of the CoM motion through foot placement. Part of the research described in this thesis (Chapter 3 and 7) is the result of collaborative research with Pratt and colleagues.

1.6 Thesis outline

The thesis is structured as follows.

Chapter 2 gives a review of relevant research on foot placement in robotic bipedal walking.

Chapter 3 addresses the first question raised: how can we determine where and when the foot should be placed to prevent a fall? The chapter introduces the capturability framework to analyze or synthesize actions that can prevent a fall. Fall avoidance is analyzed by focusing on the system's ability to eventually come to a stop without falling by taking a given number of steps. The framework is applied to three simple gait models to approximate capturability for complex legged systems. The analysis can be used to construct capture regions on the ground to which a bipedal system can step and prevent a fall.

Chapter 4 addresses the second question raised: how can we determine where and when the foot should be placed to enable the system to evolve from its initial state to any desired future state? Foot placement strategies are derived using a simple gait model that accounts for the most essential dynamics of a legged system: the CoM dynamics with respect to the CoP. A controller is presented that outputs desired foot placement strategies to reach any desired state or gait within a finite number of steps. The performance is demonstrated in simulation.

Chapter 5 addresses the third question raised: how do actuator limitations influence foot placement strategies? The cost of foot placement is studied in human-like recovery strategies following a stumble. The hypothesis is that human-like stepping strategies are the result of a minimization of cost of recovery. A simulation study evaluates five hypothetical measures for cost of recovery.

Chapter 6, 7 and 8 present and evaluate the developed robotic prototype TULip, its control algorithm and the resulting performance.

Chapter 9 presents a discussion of the research presented in this thesis and gives suggestions for future work.

Note that Chapters 3-5, 7, and part of Chapter 6 are written as separate papers that have been submitted or accepted for international conferences or journals.

Note that Chapter 3 and 7 are the result of collaborative research. Chapter 3 is an extension of previous published work of Jerry Pratt and colleagues. The description of the capturability framework is the result of collaborative work of all authors. The author of this thesis contributed to the application of the capturability framework to the presented gait models and thereby enabling the syntheses of N -step capture regions for any $N > 0$. Also, he developed the Matlab GUI together with Twan Koolen which verified and fueled our theoretical insight in the instantaneous capture point dynamics. Chapter 7 bundles and extends previous control algorithms as published by Jerry Pratt and colleagues. The author of this thesis contributed mostly to the writing and structuring of the paper and to the verification of the developed control algorithms by implementing them on TULip.

Supplemental to Chapter 8 are three videos of robot experiments of which video stills are included in the chapter. The videos are available at the website of the Delft Biorobotics Laboratory: <http://dbl.tudelft.nl>. When entering the website, click on 'walking robots' and subsequently on 'TULip'. Also, supplemental to Chapter 3 and 4, two Matlab graphical user interfaces can be downloaded. These user interfaces allow the user to manipulate the control inputs of the gait models presented in Chapter 3 and 4 and reproduce all presented results.

2

Background

2.1 Introduction

This chapter reviews relevant research on control of robotic locomotion. A broad spectrum of control approaches exist, with many approaches having elements in common. An attempt is made to categorize these approaches into four fundamentally different control methodologies or paradigms. Each of these paradigms is reviewed and special attention is paid to the role of foot placement control within these paradigms. Following the approach described in Chapter 1, we review whether foot placement control is:

- adaptive, i.e., both step time and step location can be changed;
- explicit, i.e., a function of the initial robot state and the desired future robot state;
- computationally relatively inexpensive, so that it can be run online on the robot.

Sections 2.2 to 2.5 describe the four control paradigms: *Zero Moment Point*, *Limit Cycle Walking*, *Hybrid Zero Dynamics* and *Decoupled control*. Section 2.6 concludes that all the above-mentioned aspects are found in the *Decoupled control* approach.

2.2 Zero Moment Point

Probably the most popular control approach to achieve bipedal locomotion is based on the Zero Moment Point (ZMP) (Vukobratovic and Borovac, 2004; Vukobratovic and Juricic, 1969). The ZMP can be described as the location on the ground about which the sum of all the moments of the active forces between the foot and the robot equals zero. Thus, it effectively reduces the ground reaction force distribution for single support or for double support to a single point. For flat foot contact on a horizontal floor, the ZMP location is equal to the Center of Pressure (Popovic et al., 2005). Though the ZMP is just an indicator of the robot state and potentially useful in many bipedal control approaches, its name has become a byword for a specific control approach and also a specific group of bipedal robots.

Control approaches that are specifically based on the ZMP are characterized by satisfaction of the constraint that the ZMP is strictly within the support polygon (i.e., the convex hull that encloses the foot or feet that are on the ground). This ensures that the stance foot remains firmly planted on the floor. The distance between the ZMP and the edge of the support polygon is interpreted as a measure of stability (Goswami, 1999). Offline computations are used to synthesize a gait for

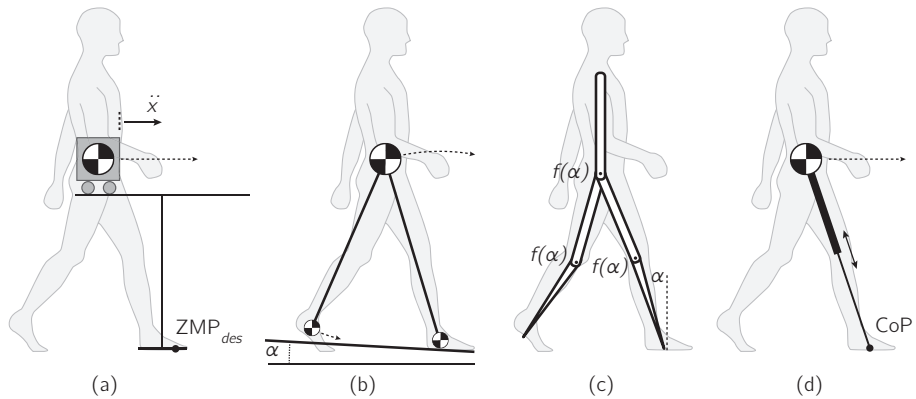


Figure 2.1: Four models that are frequently used in the analysis and control of bipedal gait and can be considered representative for the different control approaches described in this chapter. (a) Table-cart model (Kajita et al., 2003) used in ZMP-based control. CoM motion is calculated that prevents the table (i.e., robot) from tipping and ZMP at a prescribed desired location. (b) Simplest walking model (Garcia et al., 1998) used in Limit Cycle Walking control. Planar uncontrolled model that walks down a shallow slope, powered by gravity. (c) Planar 5-link model used in walking control based on Hybrid Zero Dynamics and Virtual Constraints (Westervelt et al., 2003). All link motions are a function of the state of the unactuated stance ankle. (d) Linear Inverted Pendulum Model (Kajita and Tanie, 1991). Used offline for generation of ZMP-based robotic gait (Kajita et al., 2002), but also used online for adaptive foot placement strategies (Pratt and Tedrake, 2006).

which the ZMP stays away from the support edges while satisfying predefined gait properties such as desired foot step locations or gait speed (Huang et al., 2001; Kajita et al., 2003). Complex multi-body dynamical models (Huang et al., 1999; Kagami et al., 2002; Nishtwaki et al., 1999) or more simple inverted pendulum type models (Figure 2.1a,d) can be used to generate joint trajectories and reference trajectories for the ZMP and the center of mass (CoM). These reference trajectories are then tracked by local joint controllers on the robot. ZMP-based gait typically results in a characteristic walk with flat feet and bent knees, which is required to keep full control authority over the evolution of the ZMP and CoM.

Bipedal robots that make use of the ZMP-based control approach are typically advanced robots with a humanlike morphology as depicted in Figure 2.2. The development of these robots is not solely focused on the ability to walk, but also on tasks that use machine vision, robotic manipulation and artificial intelligence. With the ZMP constraint satisfied, a designer can create a wide variety of desired robotic motions that can be tracked on the robot. This has led to an impressive variety of robotic tasks that were demonstrated, from climbing stairs (Hirai et al., 1998) to pushing a wheelchair along (Sakata et al., 2004).

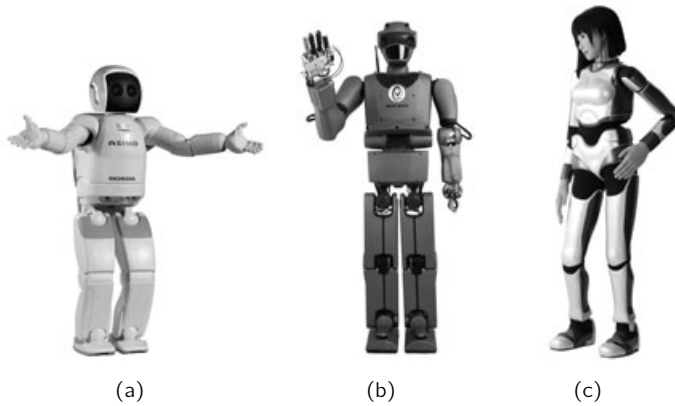


Figure 2.2: Three humanoid robots relying on ZMP-based control for walking. (a) ‘ASIMO’, from Honda (Honda, 2011). (b) ‘REEM-B’ from PAL-Robotics (Tellez et al., 2008). (c) ‘HRP-4C’ from the National Institute of Advanced Industrial Science and Technology (AIST) (Kaneko et al., 2009).

There are numerous drawbacks to ZMP-based control algorithms. Firstly, requiring flat foot contact and bent knees typically results in poor energy-efficiency. Secondly, using precomputed walking gaits results in poor robustness to unexpected disturbances. Consequently, recent research has been focused on the development of methods that enable real-time adaptation of the reference trajectories in response to large disturbances (Diedam et al., 2008; Morisawa et al., 2010; Nishiwaki and Kagami, 2010). Despite this effort, the fundamental limitation remains that ZMP-based control is not applicable in situations where the robot loses flat foot contact with the ground. In this situation, the ZMP is non-existent (or virtual as defined by Vukobratovic and Borovac, 2004) and gives no information on how to remain balanced. This makes it hard to synthesize motions where the foot can rotate with respect to the ground, which is required to achieve human-like gait or handle uneven terrain. We consider ZMP-based control too restrictive to achieve human-like walking performance.

Note that these limitations detract nothing from the fact that the ZMP (more precisely: the center of pressure) is a valuable reference point in bipedal locomotion. However, the robot’s configuration should not be adapted to achieve a predefined ZMP location. Instead, the ZMP location should be adapted to the robot’s configuration. The ZMP location is directly related to the forces that act on the center of mass (as will also be shown in Chapter 3) and is therefore an input (and not an output) of the most essential control problem of bipedal gait: controlling the center of mass dynamics.

2.3 Limit Cycle Walking

Opposite from the highly controlled ZMP-based approach is the Limit Cycle Walking approach (Hobbelen and Wisse, 2007b) that originates from the observation that a stable bipedal gait is not necessarily only a matter of actuation and control, but also a matter of the biped's mechanical design being naturally conducive to walking. This was the result of the work of McGeer (1990a), who demonstrated that a 2D (i.e., planar) unactuated mechanism could obtain a cyclic gait when walking on a gentle downhill slope. This cyclic motion is powered only by gravity and locally stable in the sense that small perturbations are rejected naturally: a change in walking velocity naturally results in a change in energy dissipation at the next step which can stabilize the gait. For a small set of initial states, the gait naturally converges to a stable limit cycle.

McGeer's work led to the development of a wide variety of passive mechanisms and models, for example with knees (Figure 2.3a), a torso (Wisse et al., 2004), or those that are capable of running (Owaki et al., 2010) or performing a three-dimensional stable walking gait (depicted in Figure 2.3b). The concept was also extended to actuated bipedal walking on level ground and robotic prototypes were developed of varying complexity, for example the bipeds depicted in Figure 2.3c-d. The general principle remained intact, being that the robot dynamics were still dominated by the

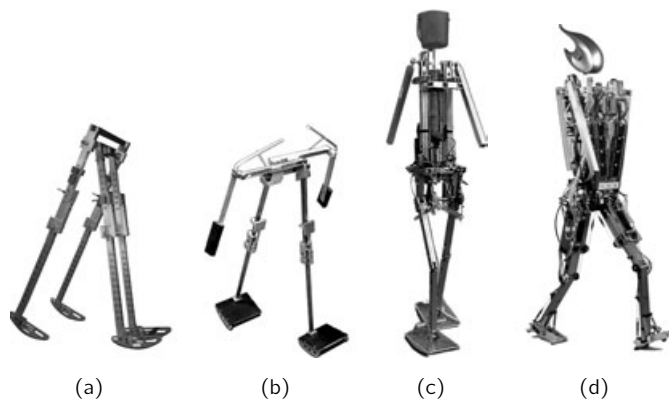


Figure 2.3: Four bipeds relying on passive system dynamics to achieve a cyclic walking gait. (a) Cornell University's copy of McGeer's planar passive dynamic walker with knees (McGeer, 1990b). (b) Passive walker with knees and counterswinging arms (Collins et al., 2001). (c) 'Denise' from Delft University of Technology (Wisse et al., 2007). (d) 'Flame' from Delft University of Technology (Hobbelen and De Boer, 2008).

natural dynamics of the limbs and only minimal actuation was applied to sustain a cyclic gait. Consequently, the resulting robotic gait can be characterized as energy-efficient and consists of smooth and life-like motions.

Analyzing or proving cyclic stability for a complex biped is not straightforward. Therefore, control algorithms for the actuated Limit Cycle Walkers are often based on dynamic principles of a simple planar gait model, typically the Simplest Walking Model (Figure 2.1b) or an extension thereof. This low-dimensional model allows for a stability analysis using Poincaré maps, where the model's cyclic motion is analyzed on a step-to-step basis. This modeling approach is used to gain insight in how a parameter variation at one point in the gait cycle affects the model's state over the course of one or multiple steps. Parameter studies on this model revealed fundamental principles in the dynamics of bipedal walking, for example concerning walking energetics (Kuo, 2007 and references therein) or walking robustness (Hobbelen and Wisse, 2008a,b).

However, transferring the insights gained from the simple planar walking model to control algorithms for a complex biped is challenging. There can be knowledge on how a parameter change in a certain direction can for example improve the walking robustness, but translating this knowledge to a control setting for a specific biped is not straightforward. The performance of these robots relies heavily on manual tuning of the control parameters and careful tuning of the mechanical structure of the robot (e.g., experimenting with joint stiffness or foot shape). The lack of state-dependent feedback mechanisms makes their performance typically unreliable. The most important stabilizing mechanism, foot placement, is left mostly as a result of the swing leg dynamics, which makes these robots very sensitive to changes in the environment or mechanical structure.

2.4 Hybrid Zero Dynamics

The Limit Cycle Walking approach demonstrates how challenging it can be to obtain a periodic gait for a complex bipedal robot, let alone prove that the robotic motions asymptotically converge to a stable limit cycle. A way of dealing with this is described by the method of Virtual Constraints and Hybrid Zero Dynamics (Grizzle et al., 2001; Westervelt et al., 2003). Feedback controllers that enforce virtual constraints on the system are designed. These virtual constraints effectively reduce the number of degrees of freedom of walking to one. The motions of all the robotic links become a function of this single degree of freedom: the stance leg ankle which is left unactuated. Using an accurate robot model (Figure 2.1c) and offline optimization techniques, joint motions are sought that ensure a provably stable periodic gait.

Walking was achieved for the planar bipeds (Chevallereau et al., 2003; Plestan et al., 2003; Sreenath et al., 2010) depicted in Figure 2.4 and in 3D for simulated bipeds (Chevallereau et al., 2009).

The resulting gait is time invariant, which means that the foot placement locations are fixed and the time of foot placement is a function of the forward progression of the robot with respect to its ankle. The robustness of the gait is determined by the ability of the actuators to keep enforcing the constraints through high-gain feedback, even in the presence of disturbances (Sreenath et al., 2010; Westervelt et al., 2004). It remains challenging to extend these methods to allow for actuated ankles, gaits with a non-instantaneous double stance phase and non periodic gaits with adaptive foot placement strategies (Grizzle et al., 2010; Sabourin and Bruneau, 2005).

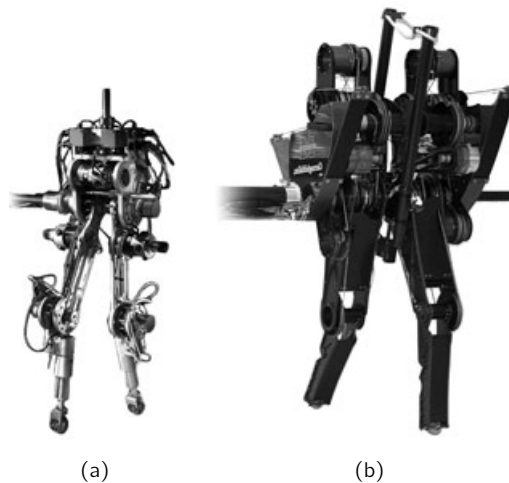


Figure 2.4: Two planar bipeds relying on Virtual Constraints and Hybrid Zero Dynamics for walking. (a) 'Rabbit' from Centre National de Recherche Scientifique (CNRS) and the University of Michigan (Chevallereau et al., 2003). (b) 'MABLE' from the University of Michigan in collaboration with Carnegie Mellon University (CMU) (Grizzle et al., 2009).

2.5 Decoupled control

Another way to reduce the dimensionality of bipedal walking is to view the walking task as a collection of several decoupled tasks, each of lower dimensionality. An overview is given of various decoupled control methods.

Raibert (1986) decoupled the control of running by control of hopping height, body orientation and velocity. The velocity was controlled using simple foot placement

strategies: the foot was placed forward, backward, or on the 'neutral point' to respectively decelerate, accelerate or maintain the same velocity. During the stance phase, hopping height was controlled by leg extension and the body orientation was controlled using hip torques.

Simple approximate rules proved effective for robust robotic running and resulted in impressive robot performance on several robots, for example for the one-legged hopper depicted in Figure 2.5a. Pratt and Pratt (1998) used a the same control decomposition for planar walking plus additional controllers for swing leg placement, and support transition (for the robot depicted in Figure 2.5b). Again, robust robotic performance was achieved using only approximate control rules for foot placement (e.g., 'increase the nominal stride length as the robot walks faster'). However, in this case the robot velocity was not solely regulated by foot placement but also by joint torques during single and double support.

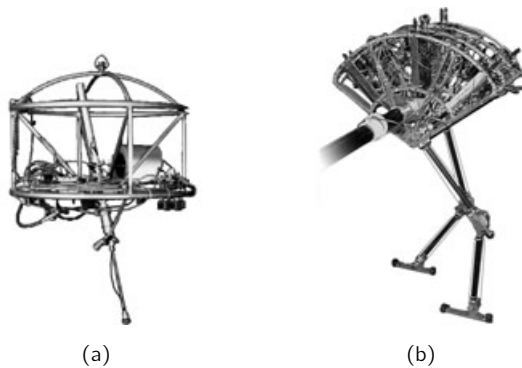


Figure 2.5: Robots relying on decoupled control for locomotion. (a) '3D One-Leg Hopper' by Massachusetts Institute of Technology (MIT) (Raibert, 1986). (b) 'Spring Flamingo' by Massachusetts Institute of Technology (MIT) (Pratt and Pratt, 1998).

The results of Raibert, Pratt and their colleagues illustrate that simple approximate foot placement strategies, typically derived from simple dynamic models, can result in robust bipedal locomotion. However, the effectiveness of these approximate rules is typically dependent on a few factors. They can require proper tuning of their control variables, either manual or automated. Also, approximate foot placement rules rely either on (a) the rapid succession of consecutive (stabilizing) steps to deal with the inherent mismatch between the approximated effect and true effect on the robot dynamics. This is typically only possible for well engineered, heavily actuated robots. Or, on (b) the existence of additional stabilizing mechanisms (e.g., ankle torques and hip torques) that together achieve the intended robotic motion.

Later, Pratt et al. (2006) introduced a more analytical and model-based approach to derive rules for velocity control through foot placement. The simple linear inverted pendulum model (introduced by Kajita and Tanie (1991) and depicted in Figure 2.1d) was used to derive the 'capture point'. This point gives an approximation of the step location for which stepping there will allow a robot to come to a stop. It uses only information about the position and velocity of the center of mass of the robot. Adjusting the foot placement location relative to the capture point yields a velocity control mechanism analogous to the neutral point of Raibert. The capture point was used in the control of foot placement in simulated robots to balance, recover from pushes and step on desired stepping stones (Pratt and Tedrake, 2006; Rebula et al., 2007). Recently, very similar reference points were derived using non-linear pendulum dynamics with impact dynamics (Stephens, 2007b; Wight et al., 2008) and were shown to be useful step indicators for push recovery on a humanoid robot (Stephens, 2007b; Stephens and Atkeson, 2010) or walking for a planar biped (Wight, 2008).

Recently, the decoupled control of bipedal locomotion has also been extended to the domain of physics-based character animations (Coros et al., 2010; De Lasa, 2010; Yin et al., 2007). Again, bipedal locomotion control is decoupled in separate control tasks of which explicit control of foot placement is one and derived from the same linear or non-linear inverted pendulum model. Decoupled control of walking was shown to be generalizable across a wide variety of gaits and simulated characters, and shown to work in the presence of disturbances or while the character is performing secondary tasks. Since the proposed walking controllers have a lot of common elements with controllers found in real robots, they can serve as an inspiration for (or indication of) future robotic behavior.

The above mentioned research suggests that simple gait models that model only the center of mass dynamics can be used effectively to synthesize robust foot placement strategies. This is also suggested by biomechanical studies, where such simple model-based laws were found to be good predictors of human foot placement (Hof, 2008; Hof et al., 2010; Millard et al., 2009; Townsend, 1985). These findings motivate the approach chosen in this research. A large part of the research builds upon the notion of the capture point and extends it to the concept of 'capturability'. This concept is applied to various 3D gait models of varying complexity. It is demonstrated how these models can be used in the control of various walking gaits, or upright balance with and without reactive stepping to remain balanced.

2.6 Conclusion

Four distinct control paradigms in the control of robotic bipedal locomotion were evaluated based on their potential to synthesize robust, versatile and energy-efficient gait. We contend that human-like walking performance can only be achieved by foot placement strategies that are adaptive, explicitly formulated and computationally achievable in real-time.

In the Zero Moment Point control paradigm, the center of mass dynamics with respect to the center of pressure (or ZMP) location is controlled explicitly. However, the CoM dynamics are typically adapted to achieve a precalculated ZMP trajectory instead of adapting the ZMP location to the current CoM dynamics. This limits the applicability of ZMP-based control to situations where the CoM motion deviates significantly from the nominal motion, for example in case of walking in the presence of perturbations or over uneven terrain. Foot placement in the Limit Cycle Walking paradigm is naturally adaptive to the robot's state, but controlled implicitly since it is mostly the result of the passive dynamics of the system. This approach is typically only practical for periodic unperturbed walking gaits.

The Hybrid Zero Dynamics paradigm relies heavily on complex offline optimizations to obtain a stable gait. For this gait, all the joints (and therefore also foot placement) are an explicit function of the stance ankle state. The step time is automatically adjusted based on the forward progression of the robot. However, this methodology only works if the robot can conform to the single prescribed gait and if the robot has pointy feet. Extending this method to more complex 3D bipeds and complex terrains is challenging.

For decoupled control, the high dimensional walking task is viewed as a collection of several decoupled tasks of lower dimensionality. Foot placement is considered a key task and controlled explicitly. The low dimensionality of each task allows for simple modeling and online adaptation to the robot state. We consider this approach currently the most promising for the synthesis of robust, versatile and energy-efficient gait.

3

Capturability-Based Analysis and Control of Legged Locomotion: Theory and Application to Three Simple Gait Models

T. Koolen, T. de Boer, J. Rebula, A. Goswami, J. Pratt
Submitted to *International Journal of Robotic research*, 2011

Abstract

This two-part paper (i.e., Chapter 3 and 7) discusses the analysis and control of legged locomotion in terms of N -step capturability: the ability of a legged system to come to a stop without falling by taking N or fewer steps. We consider this ability to be crucial to legged locomotion and a useful, yet not overly restrictive criterion for stability.

This chapter (Part 1) introduces a theoretical framework for assessing N -step capturability. This framework is used to analyze three simple models of legged locomotion. All three models are based on the 3D Linear Inverted Pendulum Model. The first model relies solely on a point foot step location to maintain balance, the second model adds a finite-sized foot, and the third model enables the use of centroidal angular momentum by adding a reaction mass. We analyze how these mechanisms influence N -step capturability, for any $N > 0$. Part 2 (i.e., Chapter 7) will show that these results can be used to control a humanoid robot.

3.1 Introduction

Preventing falls is essential in legged locomotion. A fall can be energetically costly and dangerous for both the legged system itself and other agents. Healthy humans are able to avoid falling in almost all conditions experienced in everyday life. While many legged robots can currently walk, run, and dance without falling, these tasks are usually performed in a controlled environment. Unexpected perturbations will easily topple most current bipedal robots. The ability of legged robots to avoid falling must be significantly improved before they can find utility in complex environments.

Measuring how close a legged system is to falling can provide useful insight and could be used for controller design. However, effectively quantifying closeness to falling is challenging. For traditional control systems, stability can be analyzed using measures such as eigenvalues, phase margins or loop gain margins. Legged locomotion on the other hand is generally characterized by nonlinear dynamics, under-actuation, and a combination of continuous and discrete dynamics. These properties limit the relevance of traditional analysis and control techniques to legged locomotion.

Existing stability measures for legged locomotion such as those based on the Zero Moment Point or a Poincaré map analysis may be readily computed but only apply to specific classes of controllers or robot motions (Hobbelen and Wisse, 2007a; Masani et al., 2006). More general techniques, such as the Viability Margin (Wieber, 2002), have been proposed but are difficult to compute, limiting their usefulness.

This leads us to propose the analysis of legged locomotion based on *N-step capturability*, which we informally define as the ability of a system to come to a stop without falling by taking N or fewer steps, given its dynamics and actuation limits. N -step capturability offers measures that are applicable to a large class of robot motions, including non-periodic locomotion over rough terrain with impassable regions, and it does not require a specific control system design. N -step capturability may be readily approximated, and it is useful in controller design.

Both preventing a fall and coming to a stop require adequate foot placement as a result of the ground reaction force constraints that are typical to legged locomotion. We will focus extensively on this aspect of legged locomotion using the *N-step capture region*, the set of points to which a legged system in a given state can step to become $(N - 1)$ -step capturable. A new measure of capturability in a given state, termed the *N-step capturability margin*, is then naturally defined as the size of the N -step capture region. Additionally, we will introduce the d_∞ *capturability level*, which allows a general, state-independent capturability comparison between simple gait models.

The remainder of this first part is structured as follows. Section 3.2 provides a survey of relevant literature. Section 3.3 contains definitions of the various concepts that constitute the N -step capturability framework. In Sections 3.4 through 3.7 we apply the capturability framework to three simple gait models based on the Linear Inverted Pendulum Model (Kajita et al., 2001; Kajita and Tanie, 1991). For these simple gait models, we can exactly compute capturability. Section 3.8 introduces the two capturability measures and compares the simple gait models in terms of these measures. A discussion is provided in Section 3.9, and we conclude the part in Section 7.10.

In Part 2 (i.e., Chapter 7) of the paper, we demonstrate the utility of the capturability framework by using the results of the simple gait models to control and analyze balancing and walking motions of a 3D bipedal robot with two 6-degree-of-freedom legs.

3.2 Background

The question "how stable is a given legged system?" has been the subject of much research and debate, in both robotics and biomechanics. We will now present previous work attempting to answer this question, including previous work on capturability.

The Zero Moment Point (ZMP) is often used as an aid in control development, with the constraint that it must remain in the interior of the base of support of

a legged robot. A common ZMP control method is to maintain the ZMP along a precomputed reference trajectory (Vukobratovic and Stepanenko, 1972). During walking, the error between the actual and desired ZMP can be used as a measure of the error between the current and desired state of the robot (Okumura et al., 2003). The repeatability of the gait can also be used as an error measure (Vukobratovic and Stepanenko, 1972). One drawback to following a precomputed trajectory is the inability of the robot to recover from a large unexpected push. Further work has expanded the ZMP method to include step placement adjustment in reaction to disturbances (Morisawa et al., 2009; Nishiwaki and Kagami, 2010), but there is no measure of the ability of the robot to reactively avoid a fall when following a given preplanned ZMP trajectory. In addition, the ZMP requires significant modification to apply to non-flat terrain (Wieber, 2002) or dynamic gait with a foot that rotates on the ground.

Poincaré maps have been used to measure the local stability of periodic gaits, and to induce periodic gaits of real robots using reference trajectories (Morimoto et al., 2005). Based on Poincaré Map analysis, the Gait Sensitivity Norm (Hobbelen and Wisse, 2007a) provides a measure of robustness for limit cycle walkers (Hobbelen and Wisse, 2007b) and has been shown to correlate well with the disturbance rejection capabilities of simulated planar walkers. The Gait Sensitivity Norm is calculated as the sensitivity of a given gait measure, such as step time, to a given disturbance type, such as a step-down in terrain, using a simulated model or experimental data. Another Poincaré map method based on Floquet multipliers has been used to analyze the stability of human walking gaits (Dingwell et al., 2001). However, Poincaré map analysis assumes cyclic gait to yield a measure of stability. In addition, it requires a linearization at a given point in the gait cycle, which limits the applicability of the method to large disturbances between steps where the linearization fails to capture essential dynamics of the motion (Dingwell et al., 2001).

Poincaré map analysis has also been applied to the case of passive limit cycle walkers under stochastic environmental perturbations (Byl and Tedrake, 2008), without linearizing the system around the fixed point, yielding a probabilistic basin of attraction. The stability of a walker is described with a mean first passage time, which is the expected number of steps before failure, given a set of statistics for the stochastic environmental disturbance. However, this method assumes an approximately periodic gait, and does not apply to large general disturbances such as a significant push. Poincaré map analysis has been extended to control a walker in acyclic desired gaits, by applying linear control based on a continuous family of Poincaré maps along the entire trajectory (Manchester et al., 2009). This control method can provide a measure of robustness about the desired trajectory, but it does not consider the

robustness of the desired trajectory itself.

The concepts of Virtual Constraints and Hybrid Zero Dynamics have been used to obtain and prove asymptotic stability of periodic motions for walking robots (Chevallereau et al., 2003). Introducing Virtual Constraints reduces the dimensionality of the walking system under consideration by choosing a single desired gait, allowing a tractable stability analysis. However, if actuator limitations render the robot incapable of maintaining the Virtual Constraints after a large perturbation, it is possible a fall could be avoided only by changing the desired trajectory to alter foot placement and use of angular momentum.

The Foot Placement Estimator, like the present work, considers the footstep location to be of primary importance and can be used both to control and to analyze bipedal systems (Wight et al., 2008). For a simple planar biped that maintains a rigid A-frame configuration, the Foot Placement Estimator demarcates the range of foot placement locations that will result in a statically standing system. This approach is quite similar to ours, though it is unclear how to extend this method to more general systems.

Wieber (2002) uses the concept of viability theory (Aubin, 1991) to reason about the subset of state space in which the legged system must be maintained to avoid falling. He shows a Lyapunov stability analysis for standing on non-flat terrain given a balance control law. However, the standing assumption precludes the use of this method in walking, and it provides no information on choosing step locations to avoid falling. Capturability is closely related to viability theory, but focuses on states which are most relevant to normal walking and also provides a method to explicitly compute acceptable regions to step.

In previous work, we have implicitly used the concept of capturability to develop the notion of capture points, the places on the ground to step that will allow a legged robot to come to a stop. We have used capture points based on simple models to control complex models, including a simulation of M2V2, a 12 degree of freedom humanoid robot. We have designed controllers that balance, recover from pushes, and walk across randomly placed stepping stones (Pratt and Tedrake, 2006; Rebula et al., 2007). Some of these capture point-based control methods were also implemented on the physical M2V2 (Pratt et al., 2009). We will extend the concept of capture points, applying the theory to general legged systems, considering multiple steps and providing a more complete analysis of the ability of a legged system to come to a stop.

3.3 Capturability framework

Consider a class of hybrid dynamic systems that have dynamics described by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad \text{if } h_i(\mathbf{x}) \neq 0 \quad (3.1a)$$

$$\mathbf{x} \leftarrow \mathbf{g}_i(\mathbf{x}) \quad \text{if } h_i(\mathbf{x}) = 0 \quad (3.1b)$$

$$\mathbf{u} \in \mathbf{U}(\mathbf{x}) \quad (3.1c)$$

for $i \in I \subset \mathbb{N}$. Here, \mathbf{x} is the state of the system and \mathbf{u} is the system's control input, which is confined to the state-dependent set of allowable control inputs $\mathbf{U}(\mathbf{x})$. When the system state lies on a switching surface, such that $h_i(\mathbf{x}) = 0$, the discrete jump dynamics reset the state to $\mathbf{g}_i(\mathbf{x})$ instantaneously. An *evolution* of this system is a solution to (3.1a) and (3.1b) for some input satisfying (3.1c).

For this analysis, we assume that some part of state space must be avoided at all cost – a set of *failed states*. For a bipedal robot, this set could comprise all states for which the robot has fallen. The *viability kernel*, described in (Aubin, 1991; Aubin et al., 2002) and introduced into the field of legged locomotion in (Wieber, 2000, 2002), is the set of all states from which these failed states can be avoided. That is, for every initial state in the viability kernel, there exists at least one evolution that never ends up in a failed state. As long as the system state remains within the viability kernel, the system is viable.

The viability concept arises quite naturally and can be seen as a very generic and unrestrictive definition of 'stability' for a dynamic system. However, determining the viability kernel is generally analytically intractable, and approximation is computationally expensive (Wieber, 2002). In addition, it is not trivial to synthesize a controller based solely on the viability kernel, even if it were given. This motivates the use of more restrictive definitions of stability. N -step capturability adds the restriction that the system should be able to come to a stop by taking N or fewer steps, resulting in the following definition:

Definition 1 (N -step capturable). *Let $\mathbf{X}_{\text{failed}}$ denote a set of failed states associated with a hybrid dynamic system defined by (3.1). A state \mathbf{x}_0 of this system is N -step capturable with respect to $\mathbf{X}_{\text{failed}}$, for $N \in \mathbb{N}$, if and only if there exists at least one evolution starting at \mathbf{x}_0 that contains N or fewer crossings of switching surfaces (steps), and never reaches $\mathbf{X}_{\text{failed}}$.*

Similar to the viability kernel and the *viable-capture basin* (Aubin et al., 2002), we define an N -step *viable-capture basin* as the set of all N -step capturable states. The

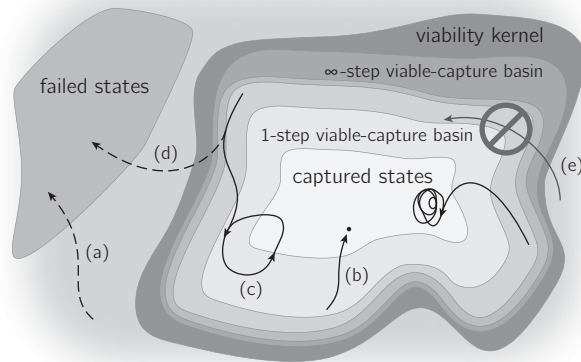


Figure 3.1: Conceptual view of the state space of a hybrid dynamic system. Several N -step viable-capture basins are shown. The boundary between two N -step viable-capture basins is part of a step surface. The ∞ -step viable-capture basin approximates the viability kernel. Several evolutions are shown: (a) an evolution starting outside the viability kernel inevitably ends up in the set of failed states; (b) the system starts in the 1-step viable-capture basin, takes a step, and comes to a rest at a fixed point inside the set of captured states (*i.e.* the 0-step viable-capture basin); (c) an evolution that eventually converges to a limit cycle; (d) an evolution that has the same initial state as (c), but ends up in the set of failed states because the input $u(\cdot)$ was different; (e) impossible evolution: by definition, it is impossible to enter the viability kernel if the initial state is outside the viability kernel.

0-step viable-capture basin will also be referred to as the set of *captured states*, and if a system's state is within the 0-step viable-capture basin, the system will be referred to as *captured*.

N -step viable-capture basins, shown schematically in Figure 3.1, describe the subsets of state space in which a controller should maintain the system so that the system *is able to* reach a captured state ('come to a stop') by taking N or fewer steps. For $N > 0$, the N -step viable-capture basin is equivalent to the set containing every initial state x_0 for which at least one evolution containing a single step and starting at x_0 reaches the $(N - 1)$ -step viable-capture basin in finite time, while never reaching a failed state. This property allows the use of recursive methods to derive or approximate N -step viable-capture basins.

The ∞ -step viable-capture basin is generally a strict subset of the viability kernel because having the ability to eventually come to a stop is not a necessary condition for avoiding the set of failed states. However, for human locomotion, the difference between the ∞ -step viable-capture basin and the viability kernel is 'small', as it is hard

to imagine a state in which a human can avoid falling, but cannot eventually come to a captured state. A notable exception is a purely passive walker (McGeer, 1990a), for which walking persists in an infinite limit cycle with no possibility of coming to a stop. In fact, an infinitely repeatable gait has been found for a simulated 3D passive walking model that has no captured states (Coleman et al., 2001).

A problem that N -step viable-capture basins share with the viability kernel is that they do not provide a direct means of controller design. This motivates the introduction of N -step capture points and N -step capture regions. While viable-capture basins specify capturability in terms of state space, capture points and capture regions are defined in Euclidean space, and describe the places where the system can step to reach a captured state. This information can for example be used to determine future step locations, to be used in a control algorithm for a bipedal robot.

We encode step locations using *contact reference points*. Each body that is allowed to come in contact with the environment during normal operation is assigned a single contact reference point, which is fixed with respect to the contacting body. Contact reference points provide a convenient, low-dimensional way of referring to the position of a contacting body, and allow us to define the N -step capture points and N -step capture regions as follows:

Definition 2 (N -step capture point, region). *Let \mathbf{x}_0 be the state of a hybrid dynamic system defined by (3.1), with an associated set of failed states \mathbf{X}_{failed} . A point \mathbf{r} is an N -step capture point for this system, for $N > 0$, if and only if there exists at least one evolution starting at \mathbf{x}_0 that contains one step, never reaches \mathbf{X}_{failed} , reaches an $(N - 1)$ -step capturable state, and places a contact reference point at \mathbf{r} at the time of the step. The N -step capture region is the set of all N -step capture points.*

A conceptual visualization of N -step capture regions is shown in Figure 3.2.

3.4 Three simple gait models

Legged locomotion can be difficult to analyze and control due to the dynamic complexity of a legged system. Simple gait models permit tractable and insightful analysis and control of walking. We present three models for which it is possible to determine N -step viable-capture basins and capture regions in closed form. The results can be used as approximations for more complex legged systems and prove useful in their control.

To illustrate the results obtained in this research, a Matlab graphical user interface (GUI) was created that allows the user to manipulate the control inputs for all models

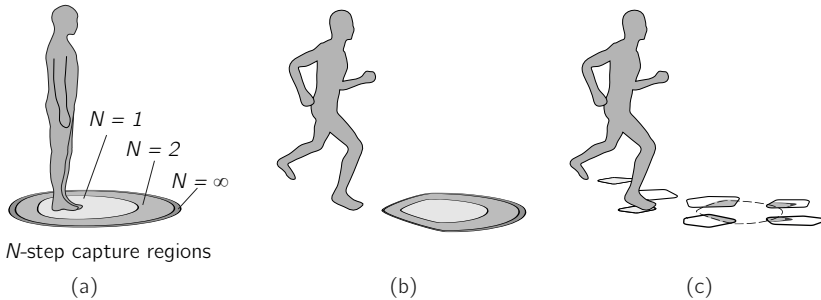


Figure 3.2: (a) A conceptual representation of the N -step capture regions for a human in a captured state (standing at rest). (b) N -step capture regions for a running human. The capture regions have decreased in size and have shifted, as compared to (a). (c) N -step capture regions for the same state as b), but with sparse footholds (e.g., stepping stones in a pond). The set of failed states has changed, which is reflected in the capture regions.

described in this paper, while the N -step capture regions are dynamically updated. This GUI is included as Multimedia Extension 1.

All three models are based on the 3D Linear Inverted Pendulum Model (3D-LIPM) (Kajita et al., 2001; Kajita and Tanie, 1991), which comprises a single point mass maintained on a plane by a variable-length leg link. The complexity of the presented models increases incrementally. To each subsequent model, another stabilizing mechanism is added. These mechanisms are generally considered fundamental in dealing with disturbances, both in the biomechanics and robotics literature (Abdallah and Goswami, 2005; Guihard and Gorce, 2002; Horak and Nashner, 1986; Hyon et al., 2007; Nenchev and Nishio, 2008; Stephens, 2007b).

The first model (Section 3.5) relies solely on point foot placement to come to a stop. The second model (Section 3.6) is obtained by adding a finite-sized foot and ankle actuation to the first model, enabling modulation of the Center of Pressure (CoP). The third model (Section 3.7) extends the second by the addition of a reaction mass and hip actuation, enabling the human-like use of rapid trunk (van der Burg et al., 2005; Horak and Nashner, 1986) or arm motions (Pijnappels et al., 2010; Roos et al., 2008).

3.5 3D-LIPM with point foot

The 3D Linear Inverted Pendulum Model, described by Kajita et al. (Kajita et al., 2001; Kajita and Tanie, 1991) and depicted in Figure 3.3, comprises a point mass

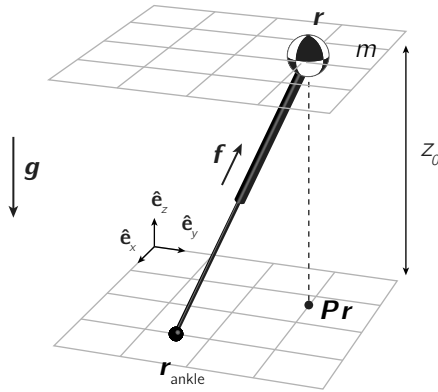


Figure 3.3: Schematic representation of the 3D-LIPM with point foot. The model comprises a point foot at position $\mathbf{r}_{\text{ankle}}$, a point mass at position \mathbf{r} with mass m and a massless telescoping leg link with an actuator that exerts a force \mathbf{f} on the point mass that keeps it at constant height z_0 . The projection matrix \mathbf{P} projects the point mass location onto the xy -plane. The gravitational acceleration vector is \mathbf{g} .

with position \mathbf{r} at the end of a telescoping massless mechanism (representing the leg), which is in contact with the flat ground. The point mass is kept on a horizontal plane by suitable generalized forces in the mechanism. Torques may be exerted at the base of the pendulum. For this first model, however, we set all torques at the base to zero. Hence, the base of the pendulum can be seen as a point foot, with position $\mathbf{r}_{\text{ankle}}$. Foot position changes, which occur when a step is taken, are assumed instantaneous, and have no instantaneous effect on the position and velocity of the point mass.

Following the capturability framework introduced in Section 3.3, we treat the 3D-LIPM with point foot as a hybrid dynamic system, with dynamics that will be derived in Section 3.5.1. Its control input is the point foot position. We define a set of allowable values for this control input, described in Section 3.5.2. The point $\mathbf{r}_{\text{ankle}}$ will be the contact reference point for all models in this paper. Changing the location of the point foot is considered crossing a step surface. The set of failed states for all simple models presented in this paper comprises all states for which $\|\mathbf{r} - \mathbf{r}_{\text{ankle}}\| \rightarrow \infty$ as $t \rightarrow \infty$, for any allowable control input.

3.5.1 Equations of motion

The equations of motion for the body mass are

$$m\ddot{\mathbf{r}} = \mathbf{f} + m\mathbf{g} \quad (3.2)$$

where m is the mass, $\mathbf{r} = (x \ y \ z)^T$ is the position of the center of mass (CoM), expressed in an inertial frame, $\mathbf{f} = (f_x \ f_y \ f_z)^T$ is the actuator force acting on the point mass and $\mathbf{g} = (0 \ 0 \ -g)^T$ is the gravitational acceleration vector.

A moment balance for the massless link shows that

$$-(\mathbf{r} - \mathbf{r}_{\text{ankle}}) \times \mathbf{f} = \mathbf{0} \quad (3.3)$$

where $\mathbf{r}_{\text{ankle}} = (x_{\text{ankle}} \ y_{\text{ankle}} \ 0)^T$ is the location of the ankle.

If $\dot{z} = 0$ initially, the point mass will stay at $z = z_0$ if $\ddot{z} = 0$. Using (3.2), we find $f_z = mg$. This can be substituted into (3.3) to find the forces f_x and f_y ,

$$\begin{aligned} f_x &= m\omega_0^2(x - x_{\text{ankle}}) \\ f_y &= m\omega_0^2(y - y_{\text{ankle}}) \end{aligned}$$

where $\omega_0 = \sqrt{\frac{g}{z_0}}$ is the reciprocal of the time constant for the 3D-LIPM.

The equations of motion, (3.2), can now be rewritten as

$$\ddot{\mathbf{r}} = \omega_0^2(\mathbf{P}\mathbf{r} - \mathbf{r}_{\text{ankle}}) \quad (3.4)$$

where $\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ projects \mathbf{r} onto the xy -plane.

Note that the equations of motion are linear. This linearity is what makes the model valuable as an analysis and design tool, as it allows us to make closed form predictions. In addition, the equations are decoupled and represent the same dynamics in the x - and y -directions. Each of the first two rows of (3.4) describes a separate 2D-LIPM with point foot. Therefore, results obtained for the 2D model can readily be extended to the 3D model.

3.5.2 Allowable control inputs

We introduce two constraints on the stepping capabilities of the model. First, we introduce an upper limit on step length, i.e., the distance between subsequent ankle locations. This maximum step length is denoted l_{max} and is assumed to be constant; it does not depend on the CoM location \mathbf{r} . Second, we introduce a lower limit to the time between steps (ankle location changes), Δt_s , which models swing leg dynamics.

3.5.3 Dimensional analysis

We perform a dimensional analysis to reduce the number of variables involved and to simplify subsequent derivations. Let us define dimensionless point mass position r' , ankle (point foot) position r'_{ankle} and time t' as¹

$$r' = \frac{r}{z_0} \quad r'_{\text{ankle}} = \frac{r_{\text{ankle}}}{z_0} \quad t' = \omega_0 t.$$

Throughout this paper, the dimensionless counterparts of all positions and lengths will be obtained by dividing by z_0 , and times and time intervals will be nondimensionalized by multiplying by ω_0 .

The dimensionless point mass position can be differentiated with respect to dimensionless time to obtain dimensionless velocity \dot{r}' and acceleration \ddot{r}' :

$$\begin{aligned} \dot{r}' &= \frac{d}{dt'} r' = \frac{\dot{r}}{\omega_0 z_0} \\ \ddot{r}' &= \frac{d}{dt'} \dot{r}' = \frac{\ddot{r}}{\omega_0^2 z_0} = \frac{\ddot{r}}{g}. \end{aligned}$$

Using these dimensionless quantities, the equations of motion, (3.4), become

$$\ddot{r}' = P r' - r'_{\text{ankle}}. \quad (3.5)$$

Further derivations will be simplified by the absence of ω_0 in this equation, as compared to (3.4).

3.5.4 Instantaneous capture point

As a first step toward examining N -step capturability, we now introduce the *instantaneous capture point*. For the 3D-LIPM with point foot, it is the point on the ground that enables the system to come to a stop if it were to instantaneously place and maintain its point foot there. Although its definition is motivated by the current model, it will also be useful in the analysis of the other models presented in this part, and we consider it an important quantity to monitor even for more complex, physical, legged systems.

Note that the instantaneous capture point is not necessarily a capture point. According to the definitions given in Section 3.3, capture points must be reachable, considering the dynamics and actuation limits, while the instantaneous capture point

¹All dimensionless quantities will be marked with a prime.

does not take into account the step time or step length constraints as defined in Section 3.5.2.

The location of the instantaneous capture point can be computed from energy considerations. For a given constant foot position, we can interpret the first two rows of (3.5) as the descriptions of two decoupled mass-spring systems, each with unit mass and negative unit stiffness. Dimensionless *orbital energies* (Kajita et al., 2001; Kajita and Tanie, 1991), $E'_{\text{LIP},x}$ and $E'_{\text{LIP},y}$, are then defined as the Hamiltonians of these systems:

$$E'_{\text{LIP},x} = \frac{1}{2}\dot{x}'^2 - \frac{1}{2}(x' - x'_{\text{ankle}})^2 \quad (3.6a)$$

$$E'_{\text{LIP},y} = \frac{1}{2}\dot{y}'^2 - \frac{1}{2}(y' - y'_{\text{ankle}})^2. \quad (3.6b)$$

Since Hamiltonians are conserved quantities, so are the orbital energies.

The orbital energy for a direction determines the behavior of the 3D-LIPM in that direction when the CoM is moving toward the foot. Considering the x' -direction for example, three cases of interest arise:

1. $E'_{\text{LIP},x} > 0$. The orbital energy is sufficient to let x' reach x'_{ankle} , after which x' continues to accelerate away from x'_{ankle} .
2. $E'_{\text{LIP},x} < 0$. x' reverses direction before x' reaches x'_{ankle} .
3. $E'_{\text{LIP},x} = 0$. x' comes to a rest exactly at x'_{ankle} .

We can solve for a foot location that results in either desired orbital energies or, equivalently, a desired velocity vector at a given value of \mathbf{r}' (Kajita et al., 2001; Kajita and Tanie, 1991). To determine the instantaneous capture point, we are interested in the foot placement required to obtain zero orbital energy in each direction. Solving (3.6) for r'_{ankle} and choosing the solution for which the point mass moves toward the point foot shows that the dimensionless version of the instantaneous capture point (Pratt and Tedrake, 2006) is

$$\mathbf{r}'_{\text{ic}} = \mathbf{P}\mathbf{r}' + \dot{\mathbf{r}}' \quad (3.7)$$

or, in terms of the original physical quantities:

$$\mathbf{r}_{\text{ic}} = \mathbf{P}\mathbf{r} + \frac{\dot{\mathbf{r}}}{\omega_0}. \quad (3.8)$$

This quantity was independently described by Hof et al. (Hof, 2008; Hof et al., 2007, 2005) and named the Extrapolated Center of Mass. It was shown to have significant ties to balancing and walking in human test subjects.

3.5.5 Instantaneous capture point dynamics

If the point foot is not instantaneously placed at the instantaneous capture point, the instantaneous capture point will move. We will now analyze this motion. The results of this analysis are depicted graphically in Figure 3.4. The dynamics that describe the motion of the instantaneous capture point on the ground can be derived by reformulating the dimensionless equations of motion in state space form. The state space model is based on the x' -dynamics only (i.e., the first row of (3.5), a 2D-LIPM), but the derivations can readily be extended to both directions, as noted in Section 3.5.1. The first row of (3.5) is rewritten in state space form as

$$\begin{pmatrix} \dot{x}' \\ \dot{x}' \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} x' \\ \dot{x}' \end{pmatrix} + \underbrace{\begin{pmatrix} 0 \\ -1 \end{pmatrix}}_{\mathbf{B}} x'_{\text{ankle}}. \quad (3.9)$$

The state matrix \mathbf{A} has eigenvalues $\lambda_{1,2} = \pm 1$ and corresponding eigenvectors

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

The eigendata show that there is a saddle point with one stable and one unstable eigenvector. The state matrix can be diagonalized using the similarity transformation $\mathbf{T} = \mathbf{V}^{-1}$, which results in the new state vector

$$\begin{pmatrix} \dot{x}'_1 \\ \dot{x}'_2 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}}_{\mathbf{T}} \begin{pmatrix} \dot{x}' \\ \dot{x}' \end{pmatrix}. \quad (3.10)$$

The new state x'_1 is identical to the instantaneous capture point x'_{ic} , and x'_2 is the point reflection of the instantaneous capture point across the projection of the point mass onto the ground. The diagonalized state space model is

$$\begin{pmatrix} \dot{x}'_1 \\ \dot{x}'_2 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}_{\mathbf{TAT}^{-1}} \begin{pmatrix} x'_1 \\ x'_2 \end{pmatrix} + \underbrace{\begin{pmatrix} -1 \\ 1 \end{pmatrix}}_{\mathbf{TB}} x'_{\text{ankle}}. \quad (3.11)$$

The diagonal state matrix \mathbf{TAT}^{-1} shows that the model's instantaneous capture point dynamics are first order. State $x'_1 = x'_{ic}$ corresponds to the unstable eigenvalue +1 and is thus of primary interest in stabilizing the system.

These derivations can be repeated for the y' -direction, so that the first row of (3.11) can be extended to

$$\dot{r}'_{ic} = r'_{ic} - r'_{\text{ankle}}. \quad (3.12)$$

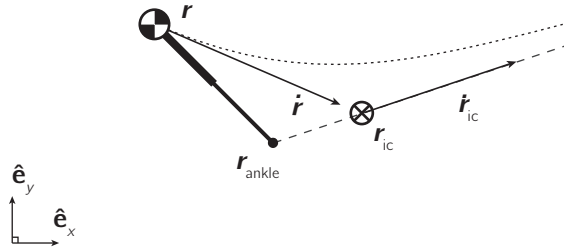


Figure 3.4: Top view of the 3D-LIPM with point foot for a given initial state at time t . By adding the CoM velocity vector \dot{r} (divided by ω_0 , see (3.8)) to the projected CoM position \mathbf{Pr} , we find the instantaneous capture point location r_{ic} . The future trajectories of the point mass and the instantaneous capture point are along the dotted lines for a constant foot location r_{ankle} . For this figure, $\mathbf{Pr} = [-0.4, 0.4, 0]$, $\dot{r} = [0.7, -0.3, 0]$, $r_{ankle} = [0, 0, 0]$, and model parameters z_0 , m and g are all set to unit magnitude.

This derivation proves the following theorem:

Theorem 1. *For the 3D-LIPM with point foot, the instantaneous capture point moves on the line through the point foot and itself, away from the point foot, at a velocity proportional to its distance from the point foot.*

As the instantaneous capture point moves away from the foot, its velocity increases exponentially. Figure 3.4 shows the motion of both the instantaneous capture point and the point mass when the point foot is kept fixed. Note that the projection of the point mass onto the xy -plane describes a hyperbolic curve, as shown in (Kajita et al., 2001).

An explicit formulation of the instantaneous capture point trajectory for a fixed foot position is found by solving (3.12):

$$r'_{ic}(\Delta t') = [r'_{ic}(0) - r'_{ankle}]e^{\Delta t'} + r'_{ankle}. \quad (3.13)$$

This equation will prove useful, both in determining whether a state is N -step capturable and in computing N -step capture regions.

3.5.6 Capturability

The instantaneous capture point is now used to determine N -step capturability for the 3D-LIPM with point foot. Although computing complete N -step viable-capture basins is possible for this model, we choose to only examine N -step capturability for a part of state space that we consider interesting. The reasons for this choice are brevity and clarity of presentation and because only those parts of the state

space need to be considered to compute the N -step capture regions and related capturability measures. For the current model in particular, we will only consider those states for which the model has just taken a step. Denoting the time at which the previous step has been taking $t'_{s,\text{prev}}$, we set $t' = t'_{s,\text{prev}} = 0$.

For the 3D-LIPM with point foot, N -step capturability for these states can be fully described in terms of the initial distance between the contact reference point and the instantaneous capture point, $\|\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{ankle}}\|$. The maximum distance for which the state is still N -step capturable will be denoted d'_N . Figure 3.5 shows an evolution that captures the model in the minimum number of steps and the values of d'_N for five values of N . We now proceed to determine these d'_N , first for $N = 0$ and then for the general case.

0-step capturability

The requirement for 0-step capturability follows directly from the definition of the instantaneous capture point, which shows that the model is 0-step capturable if and only if the instantaneous capture point coincides with the point foot location. The requirement for 0-step capturability is thus $\|\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{ankle}}\| \leq d'_0$, with $d'_0 = 0$ for this model. If this requirement is not met, then $\|\mathbf{r} - \mathbf{r}_{\text{ankle}}\| \rightarrow \infty$ as $t \rightarrow \infty$ for any evolution that contains no steps.

N -step capturability

For higher N , N -step capturability requires being able to reach an $(N - 1)$ -step capturable state using an evolution that contains only a single step. This is possible if and only if the distance between the foot and the instantaneous capture point, evaluated at the earliest possible step time, $\Delta t'_s$, is such that there exists a step of allowable length that makes the model $(N - 1)$ -step capturable:

$$\|\mathbf{r}'_{\text{ic}}(\Delta t'_s) - \mathbf{r}'_{\text{ankle}}\| \leq d'_{N-1} + l'_{\text{max}}. \quad (3.14)$$

Using (3.13), this can be rewritten as

$$\|\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{ankle}}\| \leq (d'_{N-1} + l'_{\text{max}})e^{-\Delta t'_s} = d'_N \quad (3.15)$$

which leads to a recursive expression for d'_N :

$$d'_N = (d'_{N-1} + l'_{\text{max}})e^{-\Delta t'_s}, \quad d'_0 = 0. \quad (3.16)$$

The maximum distance for N -step capturability, d'_N , follows a converging geometric series, since

$$d'_{N+1} - d'_N = (d'_N - d'_{N-1})e^{-\Delta t'_s}, \quad \forall N \geq 1.$$

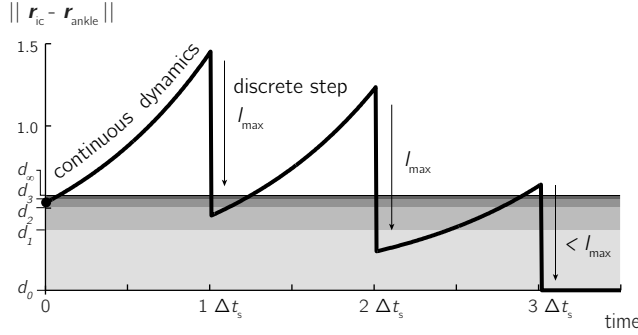


Figure 3.5: N -step capturability for the 3D-LIPM with point foot, characterized using the values of d_N (shown for $N \in \{0 \dots 3, \infty\}$). The quantity d_N is the maximum distance between the instantaneous capture point and the ankle, evaluated at step time, for which the model is N -step capturable. An example initial state is shown, which is 3-step capturable. Note that this state is different from the state depicted in Figure 3.4. Because the initial state is not 1-step capturable, the distance between the ankle and instantaneous capture point at Δt_s is larger than l_{\max} . A first step of length l_{\max} towards the instantaneous capture point results in a discrete jump in the distance between the instantaneous capture point and the ankle. A second step is required to make the state 1-step capturable, and a third step is required to reach a captured state. Note that the d_N levels only describe capturability just after a step has been taken: capturability is not in any way reduced during the continuous evolution of the dynamics. For this figure, Δt_s and l_{\max} are set to unit magnitude.

The ratio of the geometric series, $\exp(-\Delta t'_s) = \exp(-\sqrt{g/z_0}\Delta t_s)$, can be interpreted as a measure of the dynamic mobility of the legged system. The ratio is a dimensionless quantity that takes a value in the interval $[0, 1)$ if the minimum step time is strictly positive. Hence, the series d'_N converges. Moreover, notice that being allowed to take more steps to come to a stop suffers from diminishing returns. The nature of the series allows the requirement for ∞ -step capturability to be computed in closed form:

$$d'_\infty = d'_0 + \sum_{N=0}^{\infty} [d'_{N+1} - d'_N] \quad (3.17a)$$

$$= l'_{\max} \frac{e^{-\Delta t'_s}}{1 - e^{-\Delta t'_s}} \quad (3.17b)$$

since $d'_0 = 0$ for the 3D-LIPM with point foot.

3.5.7 Capture regions

The N -step capture regions for the 3D-LIPM with point foot are shown in Figure 3.6 for an example state. The values of d'_N obtained in the previous section will be used to determine these N -step capture regions in three steps:

1. determine the instantaneous capture point location at the minimum step time;
2. determine the set of possible instantaneous capture point locations before the first step is taken;
3. construct a series of nested regions around this set of possible instantaneous capture point locations.

Instantaneous capture point location after the earliest possible step time

The legged system will come to a stop if it steps to the instantaneous capture point. However, stepping is only possible after the minimum step time has passed. Hence, we first determine where the (future) instantaneous capture point will be at the first possible time at which a step can be taken. This point is readily found by substituting $\Delta t' = \Delta t'_s$ into (3.13).

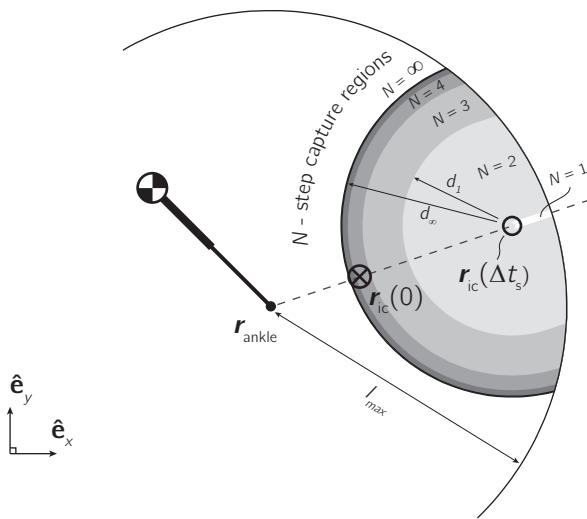


Figure 3.6: Top view of the 3D-LIPM with point foot and N -step capture regions, for the same state as shown in Figure 3.4. Additional to the information in Figure 3.4, this figure gives a schematic representation of the N -step capture regions for $N \in \{1 \dots 4, \infty\}$. Before the first step, the instantaneous capture point r_{ic} will move away from the point foot, r_{ankle} , on the dashed line. The set of possible future instantaneous capture point locations for which the minimum step time has passed is the ray starting at $r_{ic}(\Delta t_s)$ and pointing along the dashed line, away from the point foot. N -step capture regions are then found as the sets of points within a distance of d_{N-1} to the ray, as long as they lie inside the maximum step length circle. For this figure, model parameters Δt_s and l_{max} are set to unit magnitude.

Possible instantaneous capture point locations before the first step is taken

If a step is not taken at the earliest possible time, the instantaneous capture point will just keep moving farther away from the point foot, as shown by Theorem 1. Therefore, the set of possible future instantaneous capture point locations at $t' \geq \Delta t'_s$ is a ray starting at $\mathbf{r}'_{ic}(\Delta t'_s)$ which points away from \mathbf{r}'_{ankle} .

Nested regions

N -step capture regions for $N \in [1, \infty]$ can be found using this ray and the expression for d'_N in (3.16). After taking a single step to an N -step capture point, the legged system's state should be $(N - 1)$ -step capturable. Step locations that put the legged system in such a state are readily found using (3.16): all points within a distance of d'_{N-1} to a possible instantaneous capture point at $t' \geq \Delta t'_s$ are N -step capture points, provided that the legged system can reach those points given the maximum step length constraint.² This results in the nested regions depicted in Figure 3.6.

Note that finding the 1-step capture region is especially simple. Since $d'_0 = 0$, the step of finding points with distance d'_{N-1} to the ray simply results in the ray itself. The 1-step capture region is then the part of the ray that is inside the maximum step length circle.

3.6 3D-LIPM with finite-sized foot

In this section, we extend the 3D-LIPM with point foot by making the foot size finite. The finite-sized foot articulates with the leg at a 2-DoF ankle joint, and is assumed massless. At the ankle, torques may be applied in the pitch and roll directions. However, the torques are limited in such a way that the foot does not start to rotate with respect to the ground. The foot orientation (about the z -axis, *i.e.* the yaw direction) may be chosen arbitrarily when a step is taken. The model is shown in Figure 3.7.

3.6.1 Equations of motion

Only slight modifications to the derivation of the equations of motion for the 3D-LIPM are necessary. Equation (3.2) also applies to this model. Adding controllable ankle torques $\tau_{ankle,x}$ and $\tau_{ankle,y}$ and a reaction torque $\tau_{ankle,z}$ changes the moment balance of the massless leg link, (3.3), to

$$-(\mathbf{r} - \mathbf{r}_{ankle}) \times \mathbf{f} + \boldsymbol{\tau}_{ankle} = \mathbf{0} \quad (3.18)$$

²A point that cannot be reached can never be an N -step capture point.

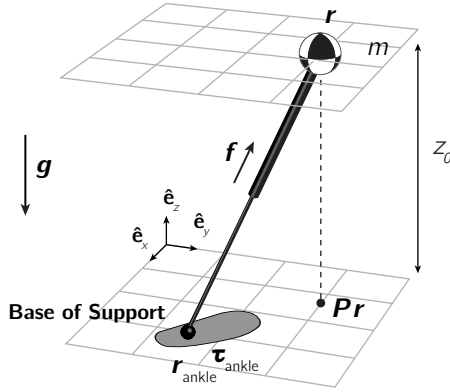


Figure 3.7: The 3D-LIPM with finite-sized foot, obtained by extending the 3D-LIPM with point foot (Figure 3.3) by a finite-sized foot and the ability to apply ankle torques τ_{ankle} .

where $\tau_{\text{ankle}} = (\tau_{\text{ankle},x} \ \tau_{\text{ankle},y} \ \tau_{\text{ankle},z})^T$ is the ankle torque and r_{ankle} is now the projection of the ankle joint onto the ground.

As before, $f_z = mg$ due to the model constraint $\dot{z} = 0$, and we find the actuator forces f_x, f_y and the reaction torque $\tau_{\text{ankle},z}$ from (3.18):

$$\begin{aligned} f_x &= m\omega_0^2(x - x_{\text{ankle}}) + \frac{\tau_{\text{ankle},y}}{z_0} \\ f_y &= m\omega_0^2(y - y_{\text{ankle}}) - \frac{\tau_{\text{ankle},x}}{z_0} \\ \tau_{\text{ankle},z} &= -\frac{\tau_{\text{ankle},x}}{z_0}(x - x_{\text{ankle}}) \\ &\quad - \frac{\tau_{\text{ankle},y}}{z_0}(y - y_{\text{ankle}}). \end{aligned}$$

The equations of motion can then be derived by substituting this into (3.2), resulting in

$$\dot{r} = \omega_0^2(\mathbf{Pr} - r_{\text{CoP}}) \tag{3.19}$$

where r_{CoP} is the location of the CoP, given by

$$\begin{aligned} r_{\text{CoP}} &= r_{\text{ankle}} + \Delta r_{\text{CoP}}, \\ \Delta r_{\text{CoP}} &= -\frac{1}{mg} \begin{pmatrix} \tau_{\text{ankle},y} \\ -\tau_{\text{ankle},x} \\ 0 \end{pmatrix} = -\frac{\tau_{\text{ankle}} \times \hat{e}_z}{mg}. \end{aligned}$$

The fact that this is the CoP for this model follows readily from a moment balance for the foot, considering that the ankle torques are such that the foot does not rotate with respect to the ground, by model definition.

Comparing (3.19) to (3.4) clearly shows that the dynamics are essentially unchanged. The only difference is that it is now possible to displace the CoP without taking a step. Hence, the results of Section 3.5.5 are still valid if r_{ankle} is replaced by r_{CoP} .

The dynamics of our 3D-LIPM with finite-sized foot are the same as those of the original 3D-LIPM by Kajita et al. (2001), where the virtual inputs are interpreted as components of an ankle torque vector, expressed in a ground-fixed frame.

3.6.2 Allowable control inputs

The step length and step time limits as defined for the 3D-LIPM with point foot in Section 3.5.2 also apply to the 3D-LIPM with finite-sized foot.³ We augment these allowable control inputs by specifying limits on the ankle torques. The allowable ankle torques are easiest to describe in terms of their resulting CoP location. To fulfill the requirement that the foot must not rotate about its edge, r_{CoP} must be kept inside the base of support.⁴

When a step is taken, the foot orientation may be chosen without restriction.

3.6.3 Dimensional analysis

In addition to the dimensionless quantities defined for the 3D-LIPM with point foot in Section 3.5.3, we define dimensionless ankle torque τ'_{ankle} as

$$\tau'_{\text{ankle}} = \frac{\tau_{\text{ankle}}}{m\omega_0^2 z_0^2}.$$

The dimensionless counterpart of the CoP is then

$$r'_{\text{CoP}} = \frac{r_{\text{CoP}}}{z_0} = r'_{\text{ankle}} - \tau'_{\text{ankle}} \times \hat{e}_z$$

and the equations of motion reduce to

$$\ddot{r}' = P r' - r'_{\text{CoP}}. \quad (3.20)$$

Replacing r'_{ankle} by r'_{CoP} , (3.12) becomes

$$\dot{r}'_{\text{ic}} = r'_{\text{ic}} - r'_{\text{CoP}} \quad (3.21)$$

³Note that the ankle location is still used as the reference point for determining step length.

⁴To be precise, r_{CoP} is the *foot rotation indicator* (Goswami, 1999), which must be kept inside the base of support to prevent foot rotation. If it is inside the base of support, then the CoP coincides with the foot rotation indicator; hence we have chosen the notation r_{CoP} .

and for a constant CoP, (3.13) becomes

$$r'_{ic}(\Delta t') = [r'_{ic}(0) - r'_{CoP}]e^{\Delta t'} + r'_{CoP}. \quad (3.22)$$

3.6.4 Equivalent constant center of pressure

To find the capture region for this model, the effect of a time-varying CoP must be investigated.

Suppose a time-varying CoP causes the instantaneous capture point to move from an initial position to a final position in a certain time interval. The *equivalent constant CoP* is the point where the CoP could have been held constant, while it would still move the instantaneous capture point from the initial position to the final position in the same time interval.⁵

We can use (3.22) to compute the equivalent constant CoP as

$$r'_{CoP,eq} = \frac{r'_{ic}(\Delta t') - r'_{ic}(0)e^{\Delta t'}}{1 - e^{\Delta t'}} \quad (3.23)$$

Let us now examine the equivalent constant CoP for a piecewise constant CoP trajectory. Suppose the CoP is initially located at $r'_{CoP,0}$, and is kept there for $\Delta t'_0$. Subsequently, it is changed to $r'_{CoP,1}$ and kept there for $\Delta t'_1$. The final instantaneous capture point position is found by applying (3.22) twice:

$$\begin{aligned} r'_{ic}(\Delta t'_0) &= [r'_{ic}(0) - r'_{CoP,0}]e^{\Delta t'_0} + r'_{CoP,0} \\ r'_{ic}(\Delta t'_0 + \Delta t'_1) &= [r'_{ic}(\Delta t'_0) - r'_{CoP,1}]e^{\Delta t'_1} + r'_{CoP,1} \end{aligned} \quad (3.24)$$

Solving (3.23) and (3.24) for $r'_{CoP,eq}$ (with $\Delta t' = \Delta t'_0 + \Delta t'_1$), we find

$$r'_{CoP,eq} = (1 - w')r'_{CoP,0} + w'r'_{CoP,1} \quad (3.25)$$

where

$$w' = \frac{e^{\Delta t'_1} - 1}{e^{\Delta t'_0 + \Delta t'_1} - 1}$$

The dimensionless scalar w' lies in the interval $[0, 1]$ because both $\Delta t'_0$ and $\Delta t'_1$ are nonnegative. The equivalent constant CoP is thus a weighted average of the two individual CoPs, where the weighting factors depend only on the time intervals. This statement can be generalized to any number of CoP changes and, in the limit, even to continuously varying CoPs, thus proving the following theorem:

⁵The equivalent constant CoP is only equivalent in terms of instantaneous capture point motion and not necessarily in terms of other parts of the state.

Theorem 2. *For the 3D-LIPM with finite-sized foot, the equivalent constant CoP is a weighted average of the CoP as a function of time.*

The time-varying CoP must always be inside the base of support, which is a convex set. By definition, a weighted average of elements of a convex set must also be in the convex set. Therefore:

Corollary 1. *If the base of support of the 3D-LIPM with finite-sized foot is constant, then the equivalent constant CoP for any realizable instantaneous capture point trajectory lies within the base of support.*

Theorem 2 and Corollary 1 greatly simplify the analysis of capturability and capture regions, since only constant CoP positions within the base of support have to be considered in our subsequent derivations.

Equation (3.25) reveals some interesting properties of computing the equivalent constant CoP for a piecewise constant CoP trajectory:

- distributivity over addition: adding a constant offset to the individual CoP locations results in an equivalent constant CoP that is offset by the same amount;
- associativity: when computing the equivalent constant CoP for a sequence of three individual CoP locations, the order of evaluation of the composition does not matter;
- non-commutativity: when computing an equivalent constant CoP for a sequence of individual CoP locations, the order of the sequence being composed does matter.

3.6.5 Capturability

The instantaneous capture point and equivalent constant CoP concepts are now used to determine capturability for the 3D-LIPM with finite-sized foot.

0-step capturability

We first analyze 0-step capturability. We can replace the point foot position by the CoP in Theorem 1 because the model dynamics are equivalent if the ankle position is replaced by CoP. Hence, the instantaneous capture point diverges away from the CoP. Since the base of support is a convex set and cannot change if no step is taken, a corollary of that theorem is:

Corollary 2. *Once the instantaneous capture point of the 3D-LIPM with finite-sized foot is outside the base of support, it is impossible to move it back inside without taking a step.*

Since a captured state can only be reached when the CoP can be made to coincide with the instantaneous capture point, Corollary 2 shows that the 3D-LIPM with finite-sized foot is 0-step capturable if and only if the instantaneous capture point is inside the base of support.

***N*-step capturability**

For higher N , capturability is analyzed in much the same way as for the 3D-LIPM with point foot. For the same reasons as mentioned in Section 3.5.6, we will not compute complete N -step viable-capture basins. For this model we restrict the analysis to states at which a step has just been taken and for which the foot is optimally oriented, in the sense that the distance between the border of the base of support and the instantaneous capture point is minimized, given a fixed ankle location.⁶ For these states, capturability can again be expressed in terms of the distance $\|r'_{ic}(0) - r'_{ankle}\|$.

The strategy that brings the model to a halt in as few steps as possible comprises stepping as soon as possible in the direction of the instantaneous capture point and always maintaining the CoP as close to the instantaneous capture point as possible.

The CoP should be placed at the point on the edge of the base of support that is closest to the instantaneous capture point. Due to the assumption of optimal orientation, this point also has the greatest distance to the ankle. This greatest distance will be denoted r_{max} and is normalized as $r'_{max} = r_{max}/z_0$. The requirement for 0-step capturability thus becomes

$$\|r'_{ic}(0) - r'_{ankle}\| \leq r'_{max} = d'_0.$$

Similar to Section 3.5.6, we can now start at (3.14) and arrive at formulas for d'_N and d'_∞ :

$$d'_N = (l'_{max} - r'_{max} + d'_{N-1})e^{-\Delta t'_s} + r'_{max}, \quad N \geq 1 \quad (3.26a)$$

$$d'_\infty = l'_{max} \frac{e^{-\Delta t'_s}}{1 - e^{-\Delta t'_s}} + r'_{max}. \quad (3.26b)$$

It is seen that the difference between d'_∞ for the model with point foot and d'_∞ for the model with finite-sized foot is simply the normalized maximum distance between the contact reference point and the edge of the foot, r'_{max} .

⁶Informally speaking: the foot is optimally oriented when the toes point in the direction of the instantaneous capture point. See Figure 3.8.

3.6.6 Capture regions

The N -step capture regions for the 3D-LIPM with finite-sized foot are shown in Figure 3.8. The analysis follows the same steps as in Section 3.5.7.

Possible instantaneous capture point locations at earliest possible step time

Due to the possibility of placing the CoP at any location in the base of support, there is now more than one location where the instantaneous capture point can be at the first time that a step can be taken, *i.e.* at $\Delta t'_s$. Theorem 2 and Corollary 1 reveal that only constant CoP positions within the base of support need to be considered in this analysis. To find the set of possible instantaneous capture point locations at $\Delta t'_s$, we apply (3.22) and scan through all possible CoP locations in the base of support. Examining (3.22) shows that this set of possible instantaneous

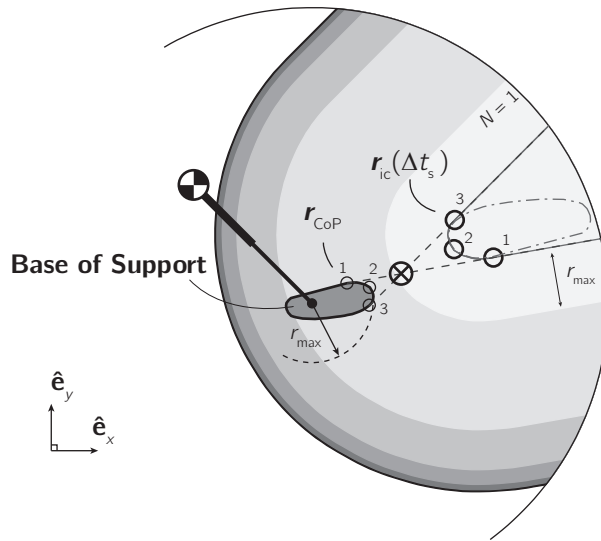


Figure 3.8: Top view of the 3D-LIPM with finite-sized foot, showing the N -step capture regions. The figure is an extension of Figure 3.6: r_{ankle} , r and \dot{r} are identical. We have omitted the labels shown in Figure 3.6 to avoid cluttering. CoP locations 1 and 3 are just in line of sight of $r_{ic}(t)$ and determine to which locations the instantaneous capture point may be directed (dashed lines). CoP location 2 is closest to $r_{ic}(t)$ and results in the closest possible location of $r_{ic}(\Delta t'_s)$. The set of all possible instantaneous capture point locations at $\Delta t'_s$ is a scaled point reflection of the base of support across the instantaneous capture point (dash-dotted lines), as demonstrated by example CoP locations 1 to 3 and corresponding capture point locations 1 to 3. To obtain the N -step capture regions, the region of possible instantaneous capture point locations before the first step is taken is surrounded by bands of width d'_N , given by (3.26a). For this figure, $r_{\text{max}} = 0.2$.

capture point locations is a scaled point reflection of the base of support across the instantaneous capture point as shown in Figure 3.8.

Possible instantaneous capture point locations before the first step is taken

If a step is not taken at the earliest possible time, the instantaneous capture point will be pushed farther and farther away by the CoP. Since the CoP can only lie within the base of support, the instantaneous capture point can only be pushed in the directions allowed by Theorem 1 (with point foot replaced by CoP), resulting in the wedge-shaped region of possible instantaneous capture point locations shown in Figure 3.8. Note that this region is bounded by the 'lines of sight' from the instantaneous capture point to the base of support (dashed lines in Figure 3.8).

Nested regions

To find the N -step capture regions, we follow the same procedure as in Section 3.5.7, that is, we create nested regions around the region of possible instantaneous capture point locations. This time, the greatest allowed distance to the possible instantaneous capture point locations is computed using (3.26a) instead of (3.16). This method assumes that the foot orientation will be chosen optimally when the step is taken. Note that for this model, $d'_0 = r'_{\max} > 0$, as opposed to the previous model. Discarding points that are outside the maximum step length circle results in the final N -step capture regions for this model.

3.7 3D-LIPM with finite-sized foot and reaction mass

We now extend the 3D-LIPM with finite-sized foot by modeling not just a point 'body' mass at the end of the leg, but a rigid body possessing a non-zero mass moment of inertia. Actuators in the hip can exert torques on this reaction mass in all directions, enabling lunging motions in 3D. The model, depicted in Figure 3.9, is a 3D version of the Linear Inverted Pendulum plus Flywheel Model presented in (Pratt et al., 2006). It can also be considered a linear version of the Reaction Mass Pendulum (Lee and Goswami, 2007) with a constant mass moment of inertia.

To make the analysis tractable, we specify several constraints. We place limits on the allowable angle of the reaction mass with respect to the vertical axis. At the start of our analysis, we assume that both this angle and the angular velocity of the body are zero. Hip torques can be used to accelerate the reaction mass, but must be followed by decelerating torques to prevent the reaction mass from exceeding its angle limit. Furthermore, we assume that the robot can only lunge once, in only one direction, similar to a human using a single impulsive lunging response in an attempt

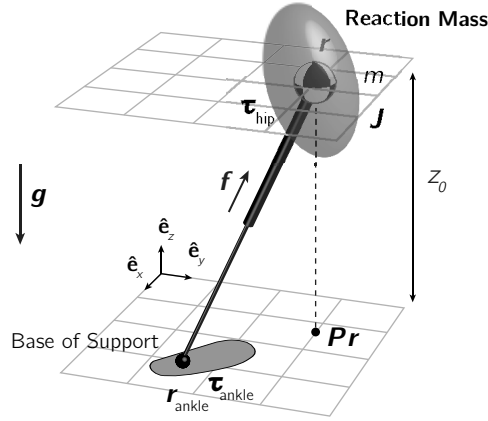


Figure 3.9: The 3D-LIPM with finite-sized foot and reaction mass. The 3D-LIPM with finite-sized foot (Figure 3.7) is extended with a non-zero mass moment of inertia tensor \mathbf{J} and the ability to apply hip torques $\boldsymbol{\tau}_{\text{hip}}$ to obtain the 3D-LIPM with finite-sized foot and reaction mass.

to regain balance after a severe perturbation. Besides angle limits, we place limits on the allowable hip torque. The hip torque component around the z -axis is determined by the requirement of no yaw of the reaction mass. This requirement makes the equations of motion linear. For the horizontal torque components, we assume a bang-bang input profile, as used in (Stephens, 2007a,b). We assume in the analysis that the execution time of the profile is less than the minimum step time and that the CoP is held constant while the torque profile is executed.

3.7.1 Equations of motion

The equations of motion for the reaction mass are

$$m\ddot{\mathbf{r}} = \mathbf{f} + m\mathbf{g} \quad (3.27a)$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}_{\text{hip}} - \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega}) \quad (3.27b)$$

where $\boldsymbol{\omega} = (\omega_x \ \omega_y \ \omega_z)^T$ is the angular velocity vector of the upper body, expressed in the inertial reference frame, $\boldsymbol{\tau}_{\text{hip}} = (\tau_{\text{hip},x} \ \tau_{\text{hip},y} \ \tau_{\text{hip},z})^T$ is the hip torque vector, \mathbf{J} is the mass moment of inertia in the body-fixed frame, and m , \mathbf{r} , \mathbf{f} and \mathbf{g} are as defined in Section 3.5.1.

Assuming that $\omega_z = 0$, that $\tau_{\text{hip},z}$ is such that $\dot{\omega}_z = 0$, and that \mathbf{J} is diagonal,

(3.27b) can be rewritten as

$$\begin{aligned} J_{xx}\dot{\omega}_x &= \tau_{\text{hip},x} \\ J_{yy}\dot{\omega}_y &= \tau_{\text{hip},y} \\ 0 &= \tau_{\text{hip},z} + (J_{xx} - J_{yy})\omega_x\omega_y. \end{aligned}$$

This last equation specifies the hip torque about the z-axis that is required to keep the reaction mass from yawing. Note that no hip torque about the z-axis is required if $J_{xx} = J_{yy}$.

The moment balance for the massless leg link is

$$-(\mathbf{r} - \mathbf{r}_{\text{ankle}}) \times \mathbf{f} - \boldsymbol{\tau}_{\text{hip}} + \boldsymbol{\tau}_{\text{ankle}} = \mathbf{0}. \quad (3.28)$$

Keeping the mass at $z = z_0$ means that $f_z = mg$, as before. This fact and (3.28) can be used to find the reaction forces f_x and f_y , and the ankle torque $\tau_{\text{ankle},z}$:

$$\begin{aligned} f_x &= m\omega_0^2(x - x_{\text{CoP}}) - \frac{\tau_{\text{hip},y}}{z_0} \\ f_y &= m\omega_0^2(y - y_{\text{CoP}}) + \frac{\tau_{\text{hip},x}}{z_0} \\ \tau_{\text{ankle},z} &= \frac{\tau_{\text{hip},x} - \tau_{\text{ankle},x}}{z_0}(x - x_{\text{ankle}}) + \frac{\tau_{\text{hip},y} - \tau_{\text{ankle},y}}{z_0}(y - y_{\text{ankle}}) + \tau_{\text{hip},z}. \end{aligned}$$

We can now rewrite (3.27) to obtain the equations of motion,

$$\ddot{\mathbf{r}} = \omega_0^2(\mathbf{P}\mathbf{r} - \mathbf{r}_{\text{CMP}}) \quad (3.29a)$$

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}\mathbf{P}\boldsymbol{\tau}_{\text{hip}} \quad (3.29b)$$

where \mathbf{r} , ω_0 and \mathbf{P} are as defined in Section 3.5.1, and

$$\mathbf{r}_{\text{CMP}} = \mathbf{r}_{\text{CoP}} + \Delta\mathbf{r}_{\text{CMP}}, \quad (3.30a)$$

$$\Delta\mathbf{r}_{\text{CMP}} = \frac{1}{mg} \begin{pmatrix} \tau_{\text{hip},y} \\ -\tau_{\text{hip},x} \\ 0 \end{pmatrix} = \frac{\boldsymbol{\tau}_{\text{hip}} \times \hat{\mathbf{e}}_z}{mg}. \quad (3.30b)$$

The point \mathbf{r}_{CMP} is the Centroidal Moment Pivot (CMP) as defined in (Popovic et al., 2005). Here we have used the fact that the CoP is equal to the Zero Moment Point when the ground is flat and horizontal (Popovic et al., 2005).

The equations of motion are again linear. Note the similarity to the equations of motion for the previous models, which allows us to reuse most results obtained for those models if the CoP is replaced by the CMP.

3.7.2 Allowable control inputs

The actuation limits of the 3D-LIPM with finite-sized foot are extended to include the hip torque profile. The set of allowable hip torque profiles is the set of bang-bang torque profiles for which the torque and angle limits are not exceeded at any time. Any allowable torque profile can be written as

$$P\boldsymbol{\tau}_{\text{hip}} = \tau_{\text{hip}}\hat{\mathbf{e}}_{\tau}[u(t) - 2u(t - \Delta t_{\text{RM}}) + u(t - 2\Delta t_{\text{RM}})] \quad (3.31)$$

where τ_{hip} is the torque magnitude, $\hat{\mathbf{e}}_{\tau}$ is the torque direction, $u(\cdot)$ is the Heaviside step function, and Δt_{RM} is the duration of each torque 'bang'. The hip torque magnitude is limited as $\tau_{\text{hip}} \leq \tau_{\text{hip,max}}$.

To comply with model assumptions, the angular velocity of the reaction mass must be zero both before and after the application of the hip torque profile, so both bangs must have equal duration. For a 2D version of the presented model, (Stephens, 2007b) and (Pratt et al., 2006) have shown that this duration has a maximum value

$$\Delta t_{\text{RM,max}} = \sqrt{J\theta_{\text{max}}/\tau_{\text{hip}}} \quad (3.32)$$

given the scalar mass moment of inertia J , the angle limit θ_{max} with respect to vertical, and the hip torque τ_{hip} . The appropriate scalar inertia value for the model presented here can be obtained from the mass moment of inertia tensor and the torque direction as $J = \hat{\mathbf{e}}_{\tau}^T \mathbf{J} \hat{\mathbf{e}}_{\tau}$.

3.7.3 Dimensional analysis

Additional dimensionless quantities are needed to nondimensionalize the equations of motion. We define the dimensionless mass moment of inertia \mathbf{J}' , angular velocity $\boldsymbol{\omega}'$, and hip torque $\boldsymbol{\tau}'_{\text{hip}}$ as

$$\mathbf{J}' = \frac{\mathbf{J}}{mz_0^2} \quad \boldsymbol{\omega}' = \frac{\mathbf{J}'\boldsymbol{\omega}}{\omega_0} \quad \boldsymbol{\tau}'_{\text{hip}} = \frac{\boldsymbol{\tau}_{\text{hip}}}{m\omega_0^2 z_0^2}$$

The dimensionless angular velocity $\boldsymbol{\omega}'$ is differentiated with respect to dimensionless time t' to obtain dimensionless angular acceleration:

$$\dot{\boldsymbol{\omega}}' = \frac{d}{dt'}\boldsymbol{\omega}' = \frac{\mathbf{J}'\dot{\boldsymbol{\omega}}}{\omega_0^2}$$

The dimensionless version of the CMP is

$$\mathbf{r}'_{\text{CMP}} = \frac{\mathbf{r}_{\text{CMP}}}{z_0} = \mathbf{r}'_{\text{CoP}} + \Delta\mathbf{r}'_{\text{CMP}} \quad (3.33a)$$

$$\Delta\mathbf{r}'_{\text{CMP}} = \boldsymbol{\tau}'_{\text{hip}} \times \hat{\mathbf{e}}_z. \quad (3.33b)$$

These quantities can be used to rewrite the equations of motion, (3.29), as

$$\ddot{\mathbf{r}}' = \mathbf{P}\mathbf{r}' - \mathbf{r}'_{\text{CMP}} \quad (3.34a)$$

$$\dot{\boldsymbol{\omega}}' = \mathbf{P}\boldsymbol{\tau}'_{\text{hip}}. \quad (3.34b)$$

Replacing \mathbf{r}'_{CoP} by \mathbf{r}'_{CMP} , (3.21) becomes

$$\mathbf{r}'_{\text{ic}} = \mathbf{r}'_{\text{ic}} - \mathbf{r}'_{\text{CMP}} \quad (3.35)$$

and for a constant CMP, (3.22) becomes

$$\mathbf{r}'_{\text{ic}}(\Delta t') = [\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{CMP}}]e^{\Delta t'} + \mathbf{r}'_{\text{CMP}}. \quad (3.36)$$

3.7.4 Effect of the hip torque profile

To analyze capturability for this model, we first examine how the hip torque profile influences the instantaneous capture point motion.

Since the CoM dynamics of the current model, (3.34a), are the same as those of the previous model, (3.19), with the CoP replaced by the CMP, we can reuse the equivalent constant CoP concept from Section 3.6.4. During the application of the hip torque profile, the CMP will first be held constant at $\mathbf{r}'_{\text{CoP}} + \tau'_{\text{hip}} \hat{\mathbf{e}}_{\tau} \times \hat{\mathbf{e}}_z$ for $\Delta t'_{\text{RM}}$, after which it moves to $\mathbf{r}'_{\text{CoP}} - \tau'_{\text{hip}} \hat{\mathbf{e}}_{\tau} \times \hat{\mathbf{e}}_z$ when the torque direction is reversed, and stays there for another $\Delta t'_{\text{RM}}$. The equivalent constant CMP is found using (3.25), with $\Delta t'_0 = \Delta t'_1 = \Delta t'_{\text{RM}}$:

$$\mathbf{r}'_{\text{CMP,eq}} = \mathbf{r}'_{\text{CoP}} + (1 - 2w')\tau'_{\text{hip}} \hat{\mathbf{e}}_{\tau} \times \hat{\mathbf{e}}_z \quad (3.37)$$

The final location of the instantaneous capture point can then be computed using (3.36):

$$\mathbf{r}'_{\text{ic}}(2\Delta t'_{\text{RM}}) = [\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{CMP,eq}}]e^{2\Delta t'_{\text{RM}}} + \mathbf{r}'_{\text{CMP,eq}}.$$

Using (3.37), this can be rewritten as

$$\mathbf{r}'_{\text{ic}}(2\Delta t'_{\text{RM}}) = [\mathbf{r}'_{\text{ic}}(0) - \mathbf{r}'_{\text{CMP}*}]e^{2\Delta t'_{\text{RM}}} + \mathbf{r}'_{\text{CoP}} \quad (3.38)$$

where

$$\mathbf{r}'_{\text{CMP}*} = \mathbf{r}'_{\text{CoP}} + \Delta \mathbf{r}'_{\text{CMP}*}, \quad (3.39a)$$

$$\Delta \mathbf{r}'_{\text{CMP}*} = v' \tau'_{\text{hip}} \hat{\mathbf{e}}_{\tau} \times \hat{\mathbf{e}}_z, \quad (3.39b)$$

$$v' = (1 - 2w')(1 - e^{-2\Delta t'_{\text{RM}}}) \quad (3.39c)$$

$$= 1 - 2e^{-\Delta t'_{\text{RM}}} + e^{-2\Delta t'_{\text{RM}}}. \quad (3.39d)$$

The vector $\Delta r'_{\text{CMP}^*}$ expresses the influence of the hip torque profile on the instantaneous capture point motion. The scalar v' can be shown to monotonically increase from 0 to 1 for $\Delta t'_{\text{RM}} > 0$.

Equation (3.38) shows that the norm of $\Delta r'_{\text{CMP}^*}$ must be maximized to gain a maximal effect of the hip torque profile on the final instantaneous capture point location. It can be shown that

$$\|\Delta r'_{\text{CMP}^*}\|_{\max} = \|\Delta r'_{\text{CMP}^*}\|_{\tau'_{\text{hip}} = \tau'_{\text{hip}, \max}}. \quad (3.40)$$

That is, even though increasing hip torque τ'_{hip} reduces the allowed torque duration $\Delta t'_{\text{RM}}$ according to (3.32), the linear term in (3.39b) outweighs the reduced value of v' .

3.7.5 Capturability

The results from Section 3.7.4 will now be used to investigate capturability.

0-step capturability

With a reaction mass, the model can be 0-step capturable even if the instantaneous capture point is not initially located inside the base of support. Rather, the requirement for 0-step capturability is that the instantaneous capture point should be inside the base of support *after the application of the torque profile*. To determine which states are 0-step capturable, we examine a boundary case for which the instantaneous capture point can only just be pushed from outside the base of support back to its edge (see Figure 3.10).

For this boundary case, it is best to place the CoP as close to the initial instantaneous capture point as possible, thus minimizing its rate of divergence. As the instantaneous capture point needs to be pushed back to the boundary of the base of support and the optimal CoP location is the closest point on that boundary, we have $r'_{\text{ic}}(2\Delta t'_{\text{RM}}) = r'_{\text{CoP}}$. The hip torque profile should always be applied as soon as possible to be most effective, since waiting longer simply results in an initial instantaneous capture point location that is farther removed from the foot. Using this information together with (3.39a) and (3.38), we obtain

$$\begin{aligned} r'_{\text{CoP}} &= [r'_{\text{ic}}(0) - r'_{\text{CoP}} - \Delta r'_{\text{CMP}^*}]e^{2\Delta t'_{\text{RM}}} + r'_{\text{CoP}} \\ \therefore \|r'_{\text{ic}}(0) - r'_{\text{CoP}}\|_{\max} &= \|\Delta r'_{\text{CMP}^*}\|_{\max} \end{aligned}$$

for the boundary case. Since the hip torque may be exerted in any direction and the CoP may be anywhere inside the base of support, the system is 0-step capturable in the general case if and only if the initial distance between the instantaneous capture point and the base of support is smaller than or equal to $\|\Delta r'_{\text{CMP}^*}\|_{\max}$.

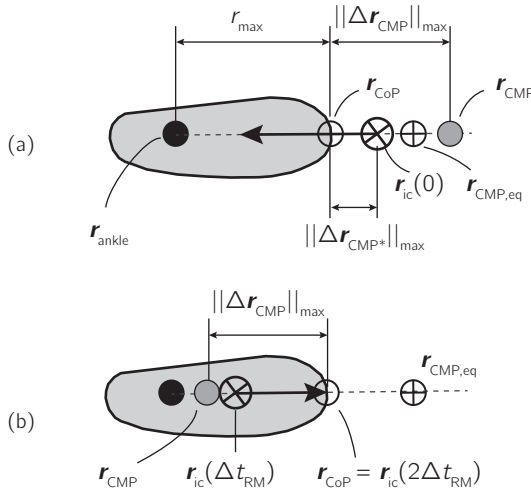


Figure 3.10: Instantaneous capture point motion during the hip torque profile, for the boundary case described in Section 3.7.5. (a) $t \in [0, \Delta t_{\text{RM}}]$: during the first half of the hip torque profile, the CMP maximally pushes the instantaneous capture point inside the base of support. (b) $t \in [\Delta t_{\text{RM}}, 2\Delta t_{\text{RM}}]$: during the ‘payback phase’, the CMP must be placed in the opposite direction to stop the spinning motion of the reaction mass. The net effect is that the instantaneous capture point ends up on the boundary of the base of support at $2\Delta t_{\text{RM}}$, exactly at the CoP. Note that the figure shows a special case where the foot is optimally oriented.

***N*-step capturability**

For *N*-step capturability, we restrict the analysis to states in which a step has just been taken, the foot is optimally oriented, and the reaction mass starts in the upright position.

For these states, the strategy that brings the model to a stop in as few steps as possible consists of stepping as soon as possible, choosing the CoP location as close as possible to the initial instantaneous capture point and lunging as soon and as hard as possible in the direction of the initial instantaneous capture point.

While this strategy is being executed, r'_{ic} , r'_{CoP} , r'_{CMP} , and r'_{ankle} are all on the same line due to optimal orientation of the foot (as in Figure 3.10). The requirement for 0-step capturability can thus be simplified for these states and written in terms of the distance to the contact reference point r'_{ankle} as

$$\|r'_{\text{ic}}(0) - r'_{\text{ankle}}\| \leq r'_{\text{max}} + \|\Delta r'_{\text{CMP*}}\|_{\text{max}} = d'_0. \quad (3.41)$$

The limit of capturability for $N = 1$ is calculated as follows. At the end of the

torque profile, the instantaneous capture point location is determined by (3.38). The motion of the instantaneous capture point between the end of the torque profile and the minimum swing time is governed by (3.36). Composing these equations results in the instantaneous capture point location at the minimum step time:

$$\mathbf{r}'_{ic}(\Delta t'_s) - \mathbf{r}'_{CoP} = (\mathbf{r}'_{ic}(0) - \mathbf{r}'_{CMP*})e^{\Delta t'_s} \quad (3.42)$$

Using the definitions of \mathbf{r}'_{CoP} and \mathbf{r}'_{CMP*} , the fact that all points involved lie on the same line, and maximizing the influence of the hip torque profile by using $\|\Delta \mathbf{r}'_{CMP*}\| = \|\Delta \mathbf{r}'_{CMP*}\|_{\max}$, we have

$$\begin{aligned} \|\mathbf{r}'_{ic}(\Delta t'_s) - \mathbf{r}'_{CoP}\| &= \|\mathbf{r}'_{ic}(\Delta t'_s) - \mathbf{r}'_{ankle}\| - r'_{\max}, \\ \|\mathbf{r}'_{ic}(0) - \mathbf{r}'_{CMP*}\| &= \|\mathbf{r}'_{ic}(0) - \mathbf{r}'_{ankle}\| - (r'_{\max} + \|\Delta \mathbf{r}'_{CMP*}\|_{\max}). \end{aligned}$$

We can use this in combination with (3.42) to find

$$\|\mathbf{r}'_{ic}(\Delta t'_s) - \mathbf{r}'_{ankle}\| = \|\mathbf{r}'_{ic}(0) - \mathbf{r}'_{ankle}\| e^{\Delta t'_s} - (r'_{\max} + \|\Delta \mathbf{r}'_{CMP*}\|_{\max})e^{\Delta t'_s} + r'_{\max} \quad (3.43)$$

Since usage of the reaction mass is no longer available after the first step is taken, the model is essentially reduced to the model presented in Section 3.6, so the requirement for 1-step capturability is that the instantaneous capture point is located inside the base of support right after the first step is taken. Stepping in the direction of the instantaneous capture point reduces its distance to the ankle by at most l'_{\max} , and after the step the instantaneous capture point should be at most r'_{\max} away from the ankle to be 0-step capturable. The criterion for 1-step capturability is therefore

$$\|\mathbf{r}'_{ic}(\Delta t'_s) - \mathbf{r}'_{ankle}\| \leq l'_{\max} + r'_{\max}.$$

Using (3.43), this becomes

$$\|\mathbf{r}'_{ic}(0) - \mathbf{r}'_{ankle}\| \leq l'_{\max} e^{-\Delta t'_s} + r'_{\max} + \|\Delta \mathbf{r}'_{CMP*}\|_{\max} = d'_1 \quad (3.44)$$

For both 0-step and 1-step capturability, we see that the margin that is gained by the addition of the reaction mass is $\|\Delta \mathbf{r}'_{CMP*}\|_{\max}$, compared to (3.26a). Recursively applying the above derivations shows that this trend continues for all N , so that

$$d'_N = (l'_{\max} - r'_{\max} + d'_{N-1})e^{-\Delta t'_s} + r'_{\max} + \|\Delta \mathbf{r}'_{CMP*}\|_{\max}, \quad N \geq 1 \quad (3.45a)$$

$$d'_\infty = l'_{\max} \frac{e^{-\Delta t'_s}}{1 - e^{-\Delta t'_s}} + r'_{\max} + \|\Delta \mathbf{r}'_{CMP*}\|_{\max}. \quad (3.45b)$$

3.7.6 Capture regions

The N -step capture regions for the 3D-LIPM with finite-sized foot and reaction mass are shown in Figure 3.11 and are derived as follows.

Possible instantaneous capture point locations at earliest possible step time

Similar to the previous models, the first step to finding the capture regions is to find the set of possible future instantaneous capture point locations at time $\Delta t'_s$. The difference that the reaction mass makes is found by rewriting (3.38) as

$$r'_{ic}(\Delta t'_s) = r'_{ic}(\Delta t'_s)|_{\tau'_{hip}=0} - \Delta r'_{CMP*} e^{\Delta t'_s} \quad (3.46)$$

where $r'_{ic}(\Delta t'_s)|_{\tau'_{hip}=0}$ is the instantaneous capture point location at $\Delta t'_s$ when no hip torque is applied, that is, when $\Delta r'_{CMP*} = 0$, for which the model reduces to the

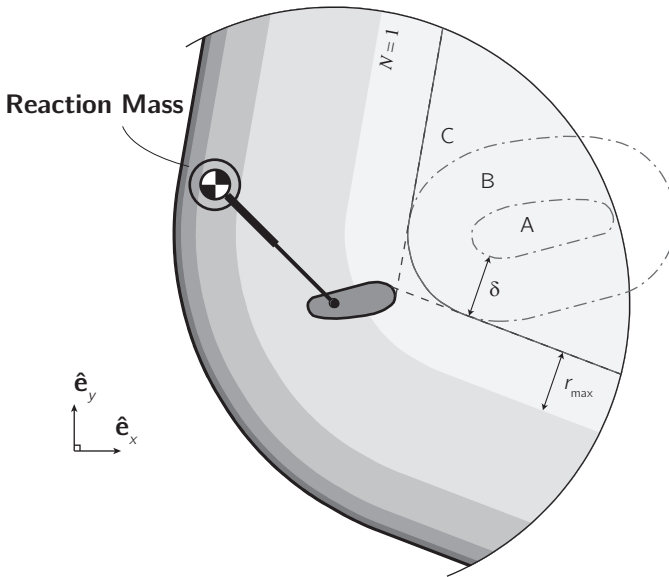


Figure 3.11: Top view of the 3D-LIPM with finite-sized foot and reaction mass, with a schematic representation of the N -step capture regions. The figure is an extension of Figure 3.8: state parameters r_{ankle} , r and \dot{r} are identical. We have omitted labels that were already shown in Figure 3.8 to avoid cluttering. The geometric construction is as follows. 1) find region A as described in Figure 3.8 and find region B by offsetting region A by δ ; 2) use the lines of sight from the base of support to find region C; 3) find the capture regions by offsetting region C by the values of d_N from (3.26a). For this figure, $\tau_{hip,max}$ is set to 0.5 and $\theta_{max} = 1/8$, which results in a total lunge time ($2\Delta t_{RM,max}$) of 1.

model without reaction mass. Taking the hip torque limit into account results in

$$\left\| r'_{ic}(\Delta t'_s)|_{\tau'_{hip}=0} - r'_{ic}(\Delta t'_s) \right\| \leq \delta$$

where

$$\delta = \|\Delta r'_{CMP*}\|_{\max} e^{\Delta t'_s}.$$

Therefore, the set of possible instantaneous capture point locations at time $\Delta t'_s$ consists of all points that lie at most δ away from possible instantaneous capture point locations at $\Delta t'_s$ for the model without reaction mass (see Section 3.6.6).

Possible instantaneous capture point locations after the earliest possible step time

After the application of the hip torque profile, the CMP will coincide with the CoP, and the instantaneous capture point will move on a line through itself and the CoP. Bounds on reachable instantaneous capture point locations are therefore found exactly as in Section 3.6.6, by constructing lines of sight (shown as the dashed lines in Figure 3.11) from the base of support to the set of possible instantaneous capture point locations at $\Delta t'_s$.

Nested regions

Finally, we can construct capture regions exactly as in Section 3.6.6. After the first step is taken, no hip torque is applied anymore and the model essentially reduces to the 3D-LIPM with finite-sized foot. We should hence construct nested regions around the set of possible future instantaneous capture point locations using the values of d'_N for the LIPM *without* reaction mass, i.e., those calculated using (3.26a), not the ones from (3.45a). The effect of the reaction mass is already incorporated in the set of possible instantaneous capture point locations at the earliest possible step time.

3.8 Capturability comparison

For all three models, we determined which states in a subset of state space are N -step capturable, and derived descriptions of the N -step capture regions. The N -step capture regions of Figure 3.6, 3.8 and 3.11 clearly showed that an increase in the number of possible stabilizing mechanisms leads to an increase in capture region size. This result implies that there is more freedom to choose foot placements that keep the model capturable, or we could say that the 'level of capturability' increases.

For a specific state, the area of the N -step capture region can be used as a measure of capturability. We call this the *N -step capturability margin*. This metric expresses

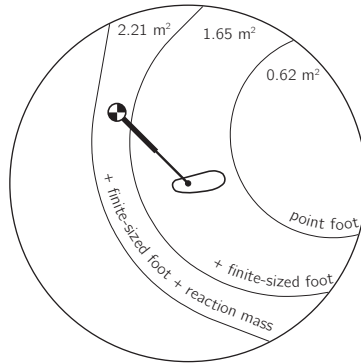


Figure 3.12: Superimposed ∞ -step capture regions of all three models, as previously presented in Figure 3.6, 3.8 and 3.11. The sizes of the ∞ -step capture regions, *i.e.* the ∞ -step capturability margins, are shown.

how close a specific state of a system is to not being N -step capturable. It also gives an indication of the input deviations and disturbances that are allowed while executing a given evolution. A small size of the ∞ -step capture region, for example, indicates that a small disturbance will likely make the legged system fall.

In the previous sections, we graphically depicted the influence of the various model parameters on the ∞ -step capture region for a given initial state. Figure 3.12 combines these results and displays the size of the three ∞ -step capture regions. Parameters were set to estimated anthropomorphic values, as presented in Appendix 3.10. For the selected initial state, the addition of a finite-sized foot caused the ∞ -step capturability margin to increase by 160%. Another increase of 30% was found for the addition of the reaction mass.

Instead of considering a specific state, we can also consider the capturability of a model in general. The d_∞ *capturability level*, which was computed for all three models, gives an indication of the overall legged-system stability and allows a comparison. In terms of the original physical quantities, d_∞ for the 3D-LIPM with finite-sized foot and reaction mass is expressed as

$$d_\infty = \underbrace{l_{\max} \frac{e^{-\omega_0 \Delta t_s}}{1 - e^{-\omega_0 \Delta t_s}}}_{\text{3D-LIPM, Section 3.5}} + \underbrace{r_{\max}}_{\text{3D-LIPM, Section 3.6}} + \underbrace{\frac{\tau_{\text{hip,max}}}{m\omega_0^2 z_0} [1 - 2e^{-\omega_0 \Delta t_{\text{RM,max}}} + e^{-2\omega_0 \Delta t_{\text{RM,max}}}]}_{\text{3D-LIPM, Section 3.7}}. \quad (3.47)$$

For the model with point foot, $d_\infty = 0.431$ using anthropometric parameters. Adding a finite-sized base of support results in $d_\infty = 0.631$, and an additional reaction mass results in $d_\infty = 0.664$.

3.9 Discussion

3.9.1 Simple models

To analyze capturability for the three presented walking models, we made extensive use of the instantaneous capture point, which is determined only by the CoM position and velocity. This gave us a dimensionally-reduced description of the dynamics of the three models. We showed how this resulted in relatively simple and comprehensible expressions, and enabled calculation and visualization of capture regions and viable-capture basins.

The three models revealed the relation between the location of the point foot, the CoP and the CMP in the analysis of capturability. Despite time variant inputs, the dynamics of the instantaneous capture point remains easy to predict for all three models: the instantaneous capture point diverges away from the CMP along a straight line at a velocity proportional to the distance to the CMP. The CMP reduces to the CoP if no reaction mass is present or actuated. The CoP reduces to the point foot location if the base of support is infinitesimally small.

The LIPM with point foot suggests that in order to remain capturable, the foot should be placed sufficiently quickly in the direction of the instantaneous capture point. This simple stepping strategy was used to create a variety of stable locomotion patterns in simulation (Pratt and Tedrake, 2006; Yin et al., 2007) and was also found to be a good predictor of stable foot placement locations in the analysis of human walking (Hof, 2008; Hof et al., 2010; Millard et al., 2009; Townsend, 1985).

The analysis for the LIPM with finite-sized foot introduced the equivalent constant CoP, which greatly simplifies the analysis of the presented models. This equivalent constant CoP is a useful analysis tool and can also be applied to robot control, as demonstrated recently (Englsberger et al., 2011).

The LIPM with finite-sized foot and reaction mass showed that lunging as soon as possible in the direction of the instantaneous capture point maximizes the level of capturability. We conjecture that bang-bang control achieves the maximal influence on instantaneous capture point motion if lunging is constrained by angle and torque limits. Note that in general it is not straightforward to relate the effect of the angular momentum generated by the simple reaction mass to the effect generated

by all individual links of a complex multibody system (Lee and Goswami, 2007; Orin and Goswami, 2008). However, this simple model still demonstrates the conceptual contribution of angular momentum to the stability of locomotion.

The influence of each stabilizing mechanism on the capturability of each model was demonstrated by (3.47). The values of d_∞ obtained for human parameters suggest that, not surprisingly, the ability to perform rapid steps is most important to remain capturable. This suggestion is also expressed by the metric being most sensitive to changes in minimum step time. A variation in minimum step time can be compensated by another stabilizing mechanism to retain the same level of capturability. However, a 10% increase in step duration already requires a 17% longer step or a 30% longer foot. For humans, selection of the appropriate step speed and length may be a trade-off between the required muscle strength to perform a quick step (Smeesters et al., 2001; Thelen et al., 1997) and the perceived level of stability or safety of the selected step length (Hsiao-Wecksler and Robinovitch, 2007; Maki and McIlroy, 1999; Weerdesteyn et al., 2005).

The use of the three presented simple models as a representation of legged locomotion has a number of limitations. The models discard many aspects of legged locomotion. Height variations of the CoM during legged locomotion were not considered. Internal forces generated by lunging or swing leg dynamics were discarded. Slippage or losses at the change of support were not considered. The existence of a double support phase in case of walking was also not taken into account. Consequently, using these simple models to approximate the capturability of a real robot will lead to discrepancies between the approximated and true values.

Furthermore, the limitations on the stabilizing control inputs were modeled simplistically. For example, consider the limitations on the stepping performance of the model. Stepping speed was constrained by enforcing a constant minimum step time, independent of the step location. Step location was constrained by limiting the maximum step length, irrespective of the current CoM position or direction of motion. The expressions for capturability in this paper rely strongly on these simplistically modeled limitations.

We see an advantage in the simplicity of the presented models however. Comparable studies demonstrated that making the models even slightly more complex can result in expressions that are less comprehensible and require numerical methods to be solved (Pratt and Drakunov, 2007; Wight et al., 2008). This decreases understanding and increases the computational burden. Although the models are very elementary, they are still useful for the analysis and control of legged locomotion.

3.9.2 Robustness metrics

We introduced the N -step capturability margin, which expresses the level of capturability for a single state and takes both the position and velocity of the CoM into account. Human subject studies already demonstrated that the CoM position and velocity in relation to the base of support is a good indicator of the ability to maintain balance and the number of steps required to do so (Aftab et al., 2010; Hsiao and Robinovitch, 2001; Maki and McIlroy, 1999; Pai and Patton, 1997; Pai et al., 1998). Hof et al. (2005) were the first to formally define the distance between the instantaneous capture point (which these authors call the 'extrapolated CoM') and the base of support as a 'margin of stability'. We see an advantage to using our metrics, since they take the effects and limits of the stabilizing control inputs (foot placement, ankle torque and hip torque) into account.

3.9.3 More complex models

Although we were able to perform a complete capturability analysis for three simple models, it remains an open issue to find a more generally applicable analytical method, or even a numerical algorithm. A possible numerical algorithm could start with a small set of states that are known to be captured, such as default standing positions. This set can then be expanded by finding initial states in its neighborhood for which there exist evolutions that reach the set and contain no steps. Subsequently, sets of N -step capturable states can be found recursively by searching for states from which it is possible to reach an $(N - 1)$ -step capturable state in a single step. While this algorithm is conceptually simple, it is likely computationally prohibitive for a complex system. In addition, including the full state of the system requires knowledge of all relevant environment information, such as the ground profile and contact characteristics. Encoding the entire environment for all time is prohibitive in general. Also note that for a system with regions of chaotic dynamics, the capturability may be uncomputable, as determining whether a state is in an N -step viable-capture basin may be undecidable (Sipser, 2005).

3.9.4 Capturability for a specific control system

The capturability analysis presented in this part took both the dynamics and actuation limits of the legged system into account, while no specific control law was assumed *a priori*. This approach allows us to make some strong conclusions concerning capturability. For example, if there is no ∞ -step capture region, then it is impossible to make the legged system come to a stop without falling, no matter what control law is used. Another approach could be to assume an existing controller and determine capturability given that controller. We can also assume a partial controller, such as

one that provides balance and swing leg control and takes a target step location as an input. Such a controller might have internal state, which must be incorporated into the robot state \mathbf{x} , but the range of actuator inputs to consider can be reduced, simplifying capturability analysis of the partially controlled system. We have used this approach to greatly reduce the actuation dimensionality of a lower body humanoid, admitting a machine learning solution for finding 1-step capture regions in simulation (Rebula et al., 2007). We also use such a parameterized controller for the robot in Part 2 (i.e., Chapter 7).

3.9.5 Capturability and Viability

Preventing a fall is important for legged locomotion. A maximally robust control system would prevent falls for all states in which preventing a fall is possible. However, designing such a control system may be impractical. Instead, we design stabilizing control systems using techniques and analysis tools which prevent falls for a subset of the theoretically possible states. We believe that focusing on preventing falls over a set of N -step capturable states will lead to robust control systems and that as N increases the states which are subsequently considered become less and less common and relevant. In addition, it is likely that analysis and control is computationally less complex for small N than for large N .

We hypothesize that nearly all human legged locomotion takes place in a 3-step viable capture basin and that all 3D bipedal robot locomotion demonstrated to date likely falls in a 2-step viable capture basin. Therefore, considering N -step capturability instead of viability focuses on the states from which it is the least difficult to avoid a fall. For large N , it may be best to just take the fall and switch to an emergency falling controller to protect the legged system and surrounding environment.

3.9.6 Future work

While the simple gait models presented in this part all pertain to bipedal walking, the concepts introduced in this paper can be applied to a wide range of walking and running legged systems, with any number of legs. One main area of future work is to generate walking and running models of increased complexity, develop algorithms for determining their capturability, and use the results to improve the robustness of legged robots. For humanoid walking robots, we are currently investigating models that incorporate uneven terrain, and consider the use of arms for pushing on walls and grabbing handrails in order to increase robustness.

3.10 Conclusion

In this paper we introduced and defined N -step capturability, and demonstrated capturability analysis on three simple gait models. The main strength of capturability analysis lies in the explicit focus on avoiding a fall in a global sense, while considering the computationally simpler issue of the ability to come to a stop in a given number of steps.

By projecting N -step capturable states to the ground using contact reference points, we can generate capture regions which define appropriate foot placement, explicitly providing practical control information and leading to the N -step capturability margin, a useful robustness metric.

In Part 2 (i.e., Chapter 7) we will show that the exact solutions to the simple models in this part can be successfully used as approximations for control of a lower body humanoid.

Acknowledgements

The authors gratefully acknowledge helpful discussions with D. Karssen, whose ideas contributed to the work. We would like to thank E. Westervelt for his helpful comments.

Appendix

Graphical user interface

A Matlab graphical user interface is created that illustrates N -step capture regions. This GUI can be downloaded from <http://dbl.tudelft.nl>.

Anthropomorphic model parameters

We estimated anthropomorphic model parameters for the 3D-LIPM with finite size foot and reaction mass, see Table 3.1. Mass and length parameters are based on a typical human 1.75 m tall and with a mass of 70 kg. Gait parameters are based on experimental studies on human trip recovery.

Table 3.1: Estimates of anthropomorphic model parameters.

	Symbol	Value	Units	Ref.
Step length	l_{\max}	0.7	m	[1, 2]
Time between steps	$\Delta t_{s,\min}$	0.3	s	[1, 2]
Ankle to toe length	r_{\max}	0.2	m	[3]
CoM height	z_0	0.95	m	[3]
HAT segment max. angle	θ_{\max}	0.5	rad	[1, 4]
Moment of inertia of HAT w.r.t. body CoM	J	8	kg m ²	[3]
Body mass	m	70	kg	[3]
Hip torque	τ_{\max}	100	Nm	[5]

References:

- [1]. Pavol et al. (2001)
- [2]. Forner Cordero et al. (2003)
- [3]. Winter (1990)
- [4]. van der Burg et al. (2005)
- [5]. Wojcik et al. (2001)

4

Foot Placement control: Step Location and Step Time as a Function of the Desired Walking Gait

T. de Boer, J.G.D. Karszen, M. Wisse
Submitted to *Robotica*, 2011

Abstract

We present a foot placement controller for walking bipedal robots, which outputs the appropriate step location and step time to obtain a desired walking gait from every feasible system state. The step location and step time are determined by approximating the robot dynamics with the 3D Linear Inverted Pendulum Model and analytically solving the constraint equations. The performance of the controller is tested in a simulation study. The results of this study show that the controller can get the system to a desired gait cycle from every feasible state within a finite number of steps.

4.1 Introduction

Generating a robust bipedal gait causes a difficult control problem. The center of mass velocity can be controlled by the center of pressure location, but the support polygon imposes a constraint on this location. Especially during single support, the support polygon is small, which results in a system that resembles a highly unstable inverted pendulum. For the system to remain stable, an appropriate new step has to be selected. This step determines possible future center of pressure locations. Consequently, deciding when and where to place a step is essential to control the center of mass and thereby obtain a robust walking gait.

Biomechanical studies found that stable step locations can be predicted based on simple inverted pendulum dynamics, and center of mass position and velocity information (Hof et al., 2005; Millard et al., 2009; Townsend, 1985). Pratt et al. (2006) expanded this finding to robotic control and introduced the *capture point*. The essence of this step indicator (or very similar ones (Hof et al., 2005; Wight et al., 2008)) is that stepping on the indicated point will allow the robot to come to a stop. The concept of the capture point is powerful and can be used to readily predict step regions for which the system has the ability to come to a stop or will inevitably fall (see Chapter 3).

Using the capture point to generate walking gait is less straightforward, since this obviously requires a non-zero forward speed. Walking controllers that step a constant offset away from the capture point have been suggested (Hof, 2008; Kajita et al., 2001; Pratt et al., 2011; Wight, 2008). This “constant offset control” approach seems logical since stepping ahead or behind the indicated point will in general cause the legged system to decelerate or accelerate respectively. Several implementations

of this control approach were demonstrated (Coros et al., 2010; Kajita et al., 2001; Pratt et al., 2011; Wight et al., 2008).

However, the constant offset control approach has three significant shortcomings, which we will demonstrate in this paper. These are: (1) The desired gait will only be reached in infinite time. This can be problematic because the system may not have recovered from a perturbation before a new perturbation occurs. (2) The controller will only work for a limited set of all feasible system states. This limits the ability to maintain stable in the presence of large perturbations. (3) The controller assumes a constant step time. This reduces the possible step strategies and therefore limits the control performance.

In this paper, we will present a novel method that overcomes these shortcomings. We introduce a controller, which we will refer to as *Dynamic Foot Placement* controller, which gives a direct relation between the desired system state and the required step time and step location to reach that state within a finite number of steps. The controller is derived using the linear inverted pendulum model. The simple nature of this model results in simple control rules which allow for real-time implementation on a real robot. We will demonstrate the effectiveness of our proposed foot placement controller and show that the controller can be used to obtain a desired cyclic walking gait from any feasible system state.

To illustrate the performance of the Dynamic Foot Placement controller, a Matlab graphical user interface (GUI) was created. It allows the user to manipulate the state of the linear inverted pendulum model and the desired walking gait that should be obtained by the foot placement controller. The GUI can be downloaded on <http://dbl.tudelft.nl>.

This paper is organized as follows. Section 4.2 presents the model, followed by a description of the walking gait in Section 4.3. A formal description of the intended control problem is given in Section 4.4. Section 4.5 describes the principle of operation of the Dynamic Foot Placement controller, followed by a comparison between the constant offset control approach and our proposed controller in Section 4.6.

4.2 Model description

To obtain simple control strategies, we use a simple model to approximate the dynamics of a bipedal robot: the 3D Linear Inverted Pendulum Model as depicted in Figure 4.1 (Kajita and Tanie, 1991). This model is widely used to synthesize control strategies for bipedal robots (Kajita et al., 2001; Kajita and Tanie, 1991) and to analyze human locomotion (Hof, 2008; Hof et al., 2005).

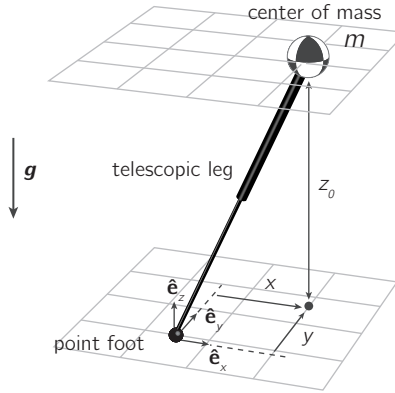


Figure 4.1: Schematic representation of the 3D-LIPM (Kajita and Tanié, 1991). The model comprises a point foot, a point mass with mass m at position (x, y, z) with respect to a local reference frame located at the point foot and a massless telescoping leg that keeps the point mass at the constant height $z = z_0$. The gravitational acceleration vector is $\mathbf{g} = (0, 0, -g)^T$. The state of the model is uniquely described by the set of state variables $\mathbf{q} = (x, y, \dot{x}, \dot{y})^T$.

The model comprises a point mass and a telescoping massless leg that is in contact with the ground. The extensible leg applies an appropriate force at the point mass to keep it at a constant height from the ground. The leg has a point foot, which can not apply torques on the ground.

4.2.1 Equations of motion

The state of the model is uniquely described by the set of state variables

$$\mathbf{q} = \begin{pmatrix} \mathbf{p} \\ \dot{\mathbf{p}} \end{pmatrix}, \quad \text{with } \mathbf{p} = \begin{pmatrix} x \\ y \end{pmatrix},$$

with x and y representing the position of the point mass with respect to a local reference frame located at the point foot. The dynamics of the model consist of one part that describes the stance phase and one part that describes the transition from one stance phase to the next. Figure 4.2 shows the dynamics of the model in x -direction for an example walking motion that consists of two stance phases and two steps.

The stance phase dynamics are continuous and described by:

$$\ddot{\mathbf{p}} = \omega_0^2 \mathbf{p}, \quad (4.1)$$

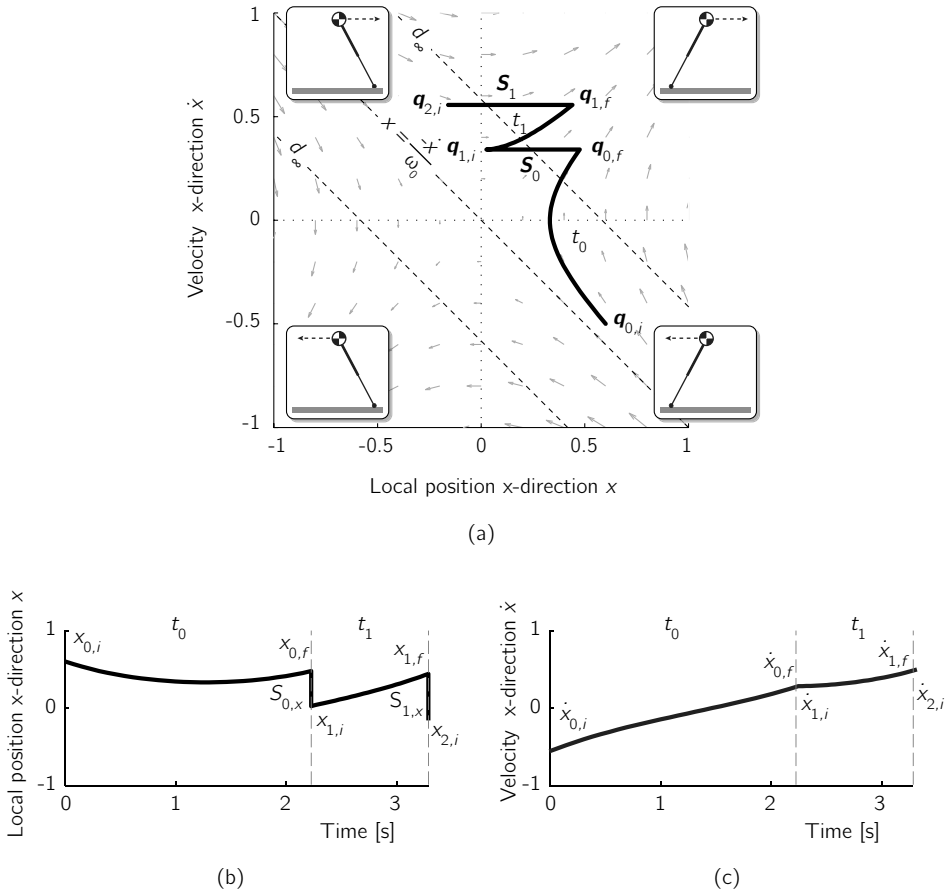


Figure 4.2: Dynamics of the 3D-LIPM in x -direction (with g and z_0 of unit magnitude) for an example initial state $\mathbf{q} = (0.6, y, 0.5, \dot{y})^T$. The model's state is shown in state space in (a) and in the time domain in (b) and (c). For the state space plot, the boxed model pictures illustrate the model's configuration for the four quadrants of the plot. Continuous dynamics describe the evolution of initial state $\mathbf{q}_{0,i}$ to state $\mathbf{q}_{0,f}$ over time t_0 . The transition dynamics describe the instantaneous change of state $\mathbf{q}_{0,f}$ to state $\mathbf{q}_{1,i}$ due to step S_0 . This sequence is repeated for consecutive stance phases and steps. Any state that lies on the line $x = -\frac{\dot{x}}{\omega_0}$ will make the model come to a stop in x -direction, i.e., $\lim_{t \rightarrow \infty} \dot{x} = 0$ (i.e., the instantaneous capture point defined by Pratt et al., 2006). The model can only come to a stop for initial states $\mathbf{q}_{n,i}$ that lie inside the ∞ -step viable-capture basin (dashed lines labeled d_∞ , see Chapter 3) due to the imposed stepping constraints, as described in Section 4.2.2.

with $\omega_0 = \sqrt{\frac{g}{z_0}}$, for which g is the gravitational constant and z_0 the height of the point mass. Solving (4.1) results in a closed form solution for the stance phase dynamics,

$$\mathbf{p}_{n,f} = \mathbf{p}_{n,i} \cosh(\omega_0 t_n) + \frac{\dot{\mathbf{p}}_{n,i}}{\omega_0} \sinh(\omega_0 t_n) \quad (4.2a)$$

$$\dot{\mathbf{p}}_{n,f} = \mathbf{p}_{n,i} \omega_0 \sinh(\omega_0 t_n) + \dot{\mathbf{p}}_{n,i} \cosh(\omega_0 t_n). \quad (4.2b)$$

where t_n is the duration of the stance phase and subscript i and f refer to the initial and final model state respectively of a stance phase, which is indexed with subscript n , with $n \in \mathbb{N}_0$.

The transition dynamics are instantaneous and describe the changes in foot positions that occur when taking a step. It is assumed that a step has no instantaneous effect on the velocity of the point mass. The instantaneous state change is given by:

$$\mathbf{p}_{n+1,i} = \mathbf{p}_{n,f} - \mathbf{S}_n \quad (4.3a)$$

$$\dot{\mathbf{p}}_{n+1,i} = \dot{\mathbf{p}}_{n,f}, \quad (4.3b)$$

where $\mathbf{S}_n = (S_{n,x}, S_{n,y})^T$ describes the step size in x - and y -direction.

4.2.2 Stepping constraints

To model the limitations of a real robot, we impose two stepping constraints on the model as introduced in Chapter 3. First, we model actuator saturation on a real robot by introducing a lower limit on the time between foot location changes:

$$t_n \geq t_{\min}. \quad (4.4)$$

Second, we model the limited kinematic workspace of a real robot by introducing an upper limit on step length, i.e., the distance between subsequent point foot locations:

$$\|\mathbf{S}_n\| \leq l_{\max}. \quad (4.5)$$

Due to these constraints, the model can only operate in a subset of the state space. The subset consist of all states for which the model has the ability to come to a stop. For states outside of this subset the model will accelerate without the possibility to decelerate. The subset is spanned by model states $\mathbf{q}_{n,i}$ that lie within the ∞ -step viable-capture basin as introduced in Chapter 3 and is given by

$$\left\| \mathbf{p}_s + \frac{\dot{\mathbf{p}}_i}{\omega_0} \right\| < \frac{l_{\max}}{e^{\omega_0 t_{\min}} - 1} = d_{\infty}. \quad (4.6)$$

In Figure 4.2, the basin boundary is indicated by the dashed lines labeled d_{∞} .

4.3 Walking gait

Our proposed foot placement controller has the ability to bring the model to any desired feasible state. In this paper, we select a more practical application of the controller by bringing the model to a state that is part of a desired walking gait. This requires that the model should not only arrive at the desired state, but should also be capable of maintaining the desired gait cycle. This requires a passed minimum step time when the gait cycle requires another step to be taken.

The desired gait cycle is set to a humanlike two-step cyclic gait with an alternating left and right step. If the model has obtained the desired gait cycle, then

$$\mathbf{q}^* = \mathbf{q}_{n+2} = \mathbf{q}_n, \quad (4.7)$$

with corresponding step time $t^* = t_{n+2} = t_n$. The step size of the desired gait cycle is,

$$\mathbf{S}_n = \begin{pmatrix} S_x^* \\ S_y^*(-1)^n \end{pmatrix}, \quad (4.8)$$

with x and y being the forward and lateral direction of movement respectively.

The control objective is a desired state within the gait cycle. For this paper, we select the state at the end of the stance phase as a reference state within the gait cycle:

$$\mathbf{q}_f^* = \begin{pmatrix} X_f^* \\ Y_f^* \\ \dot{X}_f^* \\ \dot{Y}_f^* \end{pmatrix} = \begin{pmatrix} \frac{1}{2}S_x^* \\ \frac{1}{2}S_y^*(-1)^n \\ \frac{1}{2}\omega_0 S_x^* \coth(\frac{1}{2}\omega_0 t^*) \\ \frac{1}{2}\omega_0 S_y^* \tanh(\frac{1}{2}\omega_0 t^*)(-1)^n \end{pmatrix}. \quad (4.9)$$

For all figures throughout this paper, we use an example desired gait cycle, which is depicted in Figure 4.3. For this gait cycle $t^* = 1.5$, $S_x^* = 0.75$ and $S_y^* = 0.3$. These three parameters uniquely describe the gait cycle and set the average forward velocity to $S_x^*/t^* = 0.5$. We set the model parameters g , z_0 , l_{\max} and t_{\min} to unit magnitude.

4.4 Control problem

To let the model reach the desired state \mathbf{q}_f^* within the gait cycle, the controller can adjust the step time t_n as well as the step location $S_{n,x}$ and $S_{n,y}$ over a finite number

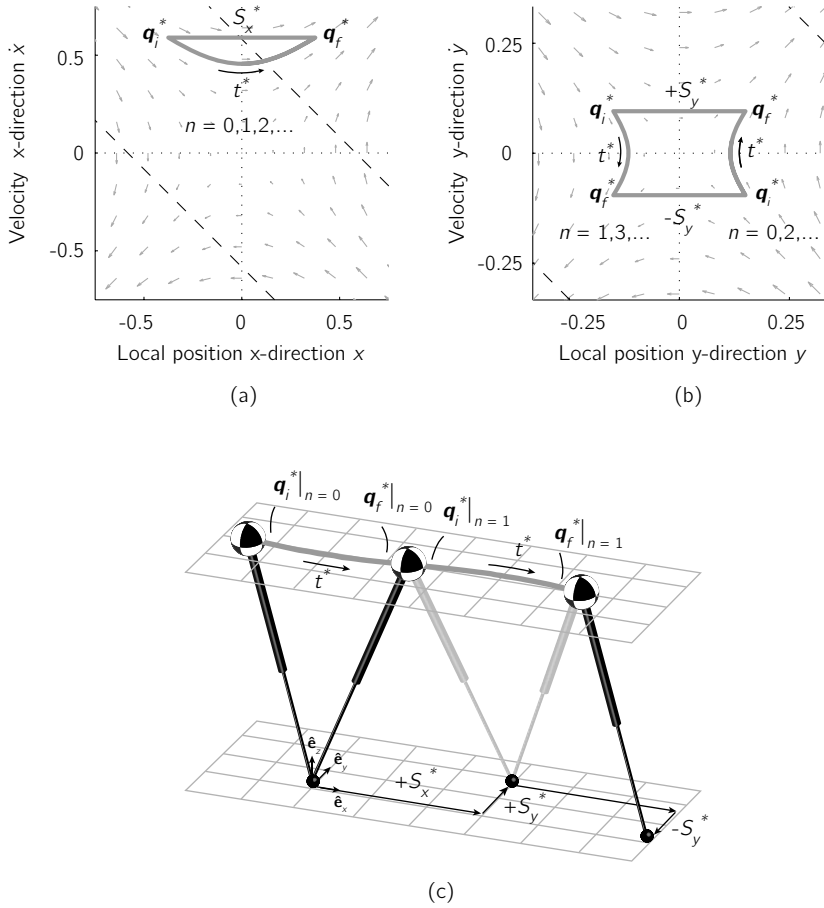


Figure 4.3: Example two-step gait cycle for the 3D-LIPM. The gait cycle is described by $\mathbf{S}^* = (0.75, 0.3)^T$, $t^* = 1.5$. (a) State space in x . (b) State space in y (c) 3D global representation of the evolution of the model state.

of N steps, while satisfying the stepping constraints (Section 4.2.2). This results in the following formal description of the control problem:

$$f(\mathbf{q}_{0,i}, t_0, \dots, t_N, \mathbf{S}_0, \dots, \mathbf{S}_{N-1}) = \mathbf{q}_f^* \quad (4.10a)$$

$$t_n \geq t_{\min} \quad \text{for } n = 0 \dots N \quad (4.10b)$$

$$\|\mathbf{S}_n\| \leq l_{\max} \quad \text{for } n = 0 \dots N - 1 \quad (4.10c)$$

in which function f gives the state of the system after the N th step. This function can be derived from (4.2) and (4.3) and is given by:

$$f(\mathbf{q}_{0,i}, t_0, \dots, t_N, \mathbf{S}_0, \dots, \mathbf{S}_{N-1}) = \begin{pmatrix} \mathbf{p}_{0,i} \cosh(\omega_0 T(N)) + \frac{\dot{\mathbf{p}}_{0,i}}{\omega_0} \sinh(\omega_0 T(N)) \\ - \sum_{i=0}^{N-1} \mathbf{S}_i \cosh(\omega_0(T(N) - T(i))) \\ \mathbf{p}_{0,i} \omega_0 \sinh(\omega_0 T(N)) + \dot{\mathbf{p}}_{0,i} \cosh(\omega_0 T(N)) \\ - \sum_{i=0}^{N-1} \mathbf{S}_i \omega_0 \sinh(\omega_0(T(N) - T(i))) \end{pmatrix} \quad \text{with } T(i) = \sum_{n=0}^i t_n. \quad (4.11)$$

There are many combinations of step locations, step sizes, and total number of steps for which the model will reach the desired gait cycle. We have chosen an approach where we minimize the required number of steps, for two reasons. First, because in case of a disturbance, it is beneficial to quickly return to the desired gait cycle in a small number of steps in order to withstand a possible new disturbance. Second, making a step is potentially costly. Using many steps may be disadvantageous from an energetic point of view and always involves risk: any step can be misplaced.

Finding a controller that satisfies the control problem (4.10) for a minimal number of steps N can be done using numerical optimization methods. However, the non-deterministic nature of these optimization methods makes it unattractive for real-time implementation on a robot.

Therefore, we adopt a different approach to synthesize a foot placement controller. We start by checking if there is a solution to the control problem for $N = 0$. If there is no solution, we repeat the procedure after an incremental increase of N . For each N , the problem that needs to be solved is different, since N determines the number of free (control) parameters and constraints (see Table 4.1). For example, a 1-step strategy has four free parameters and four equality constraints making this a fully constrained problem.

In the next section, we will show that our approach leads to an overall foot placement controller that consists of multiple simple closed-form solutions. The proposed controller solves the control problem for any number of steps.

Table 4.1: Parameters and constraints for different number of steps.

	0-step	1-step	2-step	N-step
Free parameters				
total number	1	4	7	$3N + 1$
in t	t_0	t_0, t_1	t_0, t_1, t_2	t_0, \dots, t_N
in S	-	S_0	S_0, S_1	S_0, \dots, S_{N-1}
Equality constraints				
total number	4	4	4	4
in f	$f() = \mathbf{q}_f^*$	$f() = \mathbf{q}_f^*$	$f() = \mathbf{q}_f^*$	$f() = \mathbf{q}_f^*$
Inequality constraints				
number	1	3	5	$2N + 1$
in t	$t_n \geq t_{\min}$	$t_n \geq t_{\min}$	$t_n \geq t_{\min}$	$t_n \geq t_{\min}$
in S	-	$\ S_n\ \leq l_{\max}$	$\ S_n\ \leq l_{\max}$	$\ S_n\ \leq l_{\max}$
Problem				
type	over-constrained	fully constrained	under-constrained	-

4.5 Dynamic Foot Placement controller

The Dynamic Foot Placement controller outputs the next step location and step time, expressed in terms of the control parameters $S_{x,n}$, $S_{y,n}$ and t_n . We will derive expressions for these control parameters for each initial model state. The foot placement controller consists of four N -step strategies with $N = 0, 1, 2, (N > 2)$. Each N -step strategy will result in an $(N - 1)$ -step strategy up to a 0-step strategy, as demonstrated for an example 3-step strategy in Figure 4.4.

4.5.1 0-step strategy

If the initial state is already on the gait cycle or will arrive at the gait cycle without taking a step, then the following should hold:

$$\mathbf{q}_{0,f} = \mathbf{q}_f^* \quad \wedge \quad t_0 \geq t_{\min}. \quad (4.12)$$

The control parameters are then given by $S_{0,x} = S_x^*$, $S_{0,y} = S_y^*$ and

$$t_0 = \omega_0^{-1} \log \left(\frac{\dot{x}_f^* + \sqrt{\dot{x}_f^{*2} + \omega_0^2 x_{0,i}^2 - \dot{x}_{0,i}^2}}{\omega_0 x_{0,i} + \dot{x}_{0,i}} \right), \quad (4.13)$$

which is found by solving (4.12) for t in which $\mathbf{q}_{0,f}$ is given by (4.2).

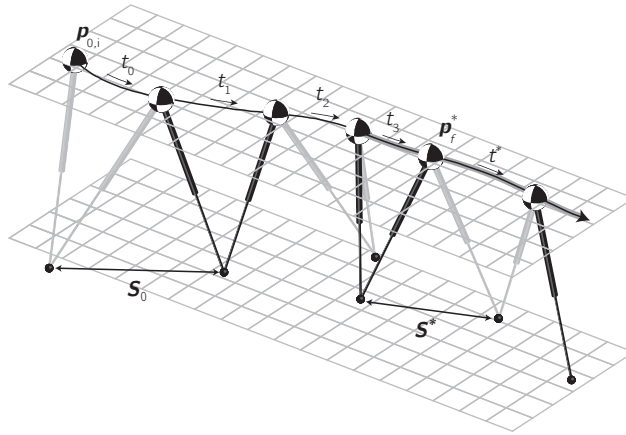
4.5.2 1-step strategy

The 1-step strategy is fully constrained (Table 4.1), which means that a unique solution exists for the control parameters $S_{x,0}$, $S_{y,0}$, t_0 and t_1 . To derive an expression for these parameters it is important to acknowledge the following: since a step has no instantaneous effect on the model velocities, we know that the velocities in the x - and y -direction at the moment of the step should match the velocities of the desired gait cycle at some instant of the gait.

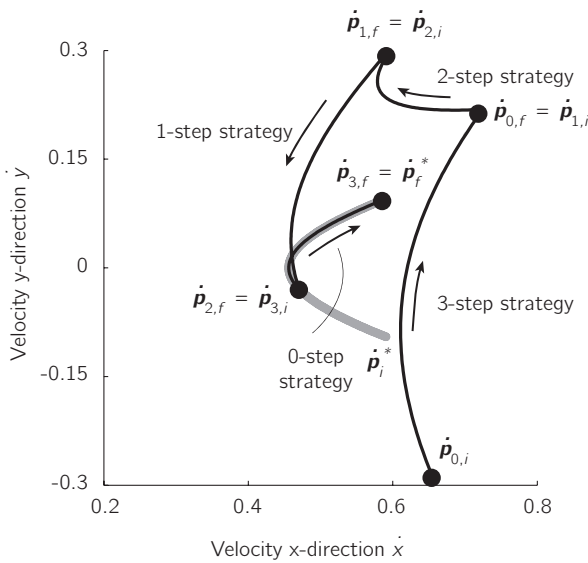
To further clarify this fact, let us consider the model velocities $\dot{\mathbf{p}}$ shown in the phase space diagram depicted in Figure 4.4(b). Here, we see a 1-step strategy that is part of a 3-step strategy. We see that, going from a 1-step strategy to a 0-step strategy, the phase space trajectory of the model velocities intersects the phase space trajectory of the possible gait-cycle velocities. At the instant that the velocities match, a step is taken and the model evolves to the state \mathbf{q}_f^* .

Consequently, for a 1-step strategy to be feasible, an intersection $\dot{\mathbf{p}}_{0,f} = \dot{\mathbf{p}}_{1,i}$ should occur. This means that the following expression, based on (4.2b), should hold,

$$\mathbf{p}_{0,i} \omega_0 \sinh(\omega_0 t_0) + \dot{\mathbf{p}}_{0,i} \cosh(\omega_0 t_0) = \mathbf{p}_f^* \omega_0 \sinh(\omega_0(-t_1)) + \dot{\mathbf{p}}_f^* \cosh(\omega_0(-t_1)). \quad (4.14)$$



(a)



(b)

Figure 4.4: Example 3-step strategy of the model in initial state $\dot{q}_{0,i}$ to the desired state \dot{q}_f^* part of the desired gait cycle (highlighted by the thick grey line). The initial state is $\dot{q}_{0,i} = (0.25, 0.58, 0.65, 0.30)^T$. All step constraints described in Section 4.2.2 are satisfied. (a) 3D global representation of the evolution of state component p . (b) The evolution of the velocities in x - and y -direction, i.e., state component \dot{p} . Black dots indicate the states just after a step.

This expression can be solved to find the stance phase durations t_0 and t_1 . We omit the expressions for t_0 and t_1 because of their length.

With the known stance phase durations t_0 and t_1 , we can also derive the expression for the step size \mathbf{S}_0 from (4.2a) and (4.3a),

$$\begin{aligned} \mathbf{S}_0 &= \mathbf{p}_{0,f} - \mathbf{p}_{1,i}, & \text{with} & & (4.15) \\ \mathbf{p}_{0,f} &= \mathbf{p}_{0,i} \cosh(\omega_0 t_0) + \frac{\dot{\mathbf{p}}_{0,i}}{\omega_0} \sinh(\omega_0 t_0) \\ \mathbf{p}_{1,i} &= \mathbf{p}_f^* \cosh(-\omega_0 t_1) + \frac{\dot{\mathbf{p}}_f^*}{\omega_0} \sinh(-\omega_0 t_1). \end{aligned}$$

The last part of the 1-step strategy is to check if the found step durations t_0 and t_1 and step size \mathbf{S}_0 are within the stepping constraints, (4.4) and (4.5). If this is not the case, the 2-step strategy should be considered.

4.5.3 2-step strategy

A 2-step strategy is underconstrained (Table 4.1), meaning that multiple solutions can exist for the available control parameters \mathbf{S}_0 , \mathbf{S}_1 and stance durations t_0 , t_1 and t_2 . The constraint equation (4.11) in case of a 2-step strategy becomes

$$f(q_{0,i}, t_0, t_1, t_2, \mathbf{S}_0, \mathbf{S}_1) = q_f^*. \quad (4.16)$$

Solving (4.16) for \mathbf{S}_0 , \mathbf{S}_1 results in

$$\begin{aligned} \mathbf{S}_0 &= -\left(\frac{\dot{\mathbf{p}}_{0,i}}{\omega_0} \cosh(\omega_0 (t_0 + t_1)) + \mathbf{p}_{0,i} \sinh(\omega_0 (t_0 + t_1)) \right. \\ &\quad \left. - \frac{\dot{\mathbf{p}}_f^*}{\omega_0} \cosh(\omega_0 t_2) + \mathbf{p}_f^* \sinh(\omega_0 t_2) \right) \frac{1}{\sinh(\omega_0 t_1)} \end{aligned} \quad (4.17)$$

$$\begin{aligned} \mathbf{S}_1 &= \left(-\frac{\dot{\mathbf{p}}_f^*}{\omega_0} \cosh(\omega_0 (t_1 + t_2)) + \mathbf{p}_f^* \sinh(\omega_0 (t_1 + t_2)) \right. \\ &\quad \left. + \frac{\dot{\mathbf{p}}_{0,i}}{\omega_0} \cosh(\omega_0 t_0) + \mathbf{p}_{0,i} \sinh(\omega_0 t_0) \right) \frac{1}{\sinh(\omega_0 t_1)}. \end{aligned} \quad (4.18)$$

We can set three of the seven free control parameters (Table 4.1) and thereby derive a value for all parameters. An example option would be to set all stance durations t_0 , t_1 and t_2 to t_{\min} , which we have done for the visualization of the control performance in Section 4.6. Though this approach might result in a stepping strategy that does not use the absolute minimum number of steps for some model states, Section 4.6 demonstrates that the overall performance of the controller is still close to optimal.

4.5.4 ($N > 2$)-step strategy

For all model states that cannot evolve to the gait cycle using N -step strategies for $N \leq 2$, we adopt a single step strategy. Though more suited stepping strategies for $N > 2$ can be derived, these were found to be complex and result in a negligible increase in stepping performance.

The model state should quickly be brought to a region in state space for which N -step strategies for $N \leq 2$ do apply, consequently t_n is set to t_{\min} . A corresponding effective step location is a point at an offset from the instantaneous capture point (Pratt et al., 2006) and was introduced as part of the constant offset controller, explained in (Hof, 2008):

$$\mathbf{s}_n = \mathbf{p} + \frac{\dot{\mathbf{p}}}{\omega_0} + \begin{pmatrix} \frac{-S_y^*}{e^{\omega_0 t^*} - 1} \\ \frac{S_y^* (-1)^n}{e^{\omega_0 t^*} + 1} \end{pmatrix}. \quad (4.19)$$

If this point is not within reach due to stepping constraints, we step on the instantaneous capture point or maximally towards this point if it is also out of reach.

4.6 Control performance

In the previous section, we introduced the Dynamic Foot Placement controller. In this section, we discuss the performance of this controller. The performance is discussed in terms of the numbers of steps it takes to reach the desired gait cycle (Section 4.6.1) and the time response (Section 4.6.2).

4.6.1 Number of steps

The goal of the Dynamic Foot Placement controller is to get the model to the desired gait cycle from every possible initial state with the least number of steps. We check how well the controller reaches this goal by comparing the required number of steps to the absolute minimal number of steps required. The numerical method used to determine the absolute minimal number of steps is explained in the appendix. We show the results for the special case of 2D walking, in which $S_y^* = 0$, $y = 0$ and $\dot{y} = 0$. We use this special case, because for this case the results can easily be visualized in a graph, since the state space is only 2-dimensional instead of 4-dimensional for the case of 3D walking.

Figure 4.5(c) shows the absolute minimal number of steps needed to get to the desired gait cycle for a set of initial states. It shows that it is possible to get to the desired gait cycle from all initial states within the ∞ -step viable-capture basin.

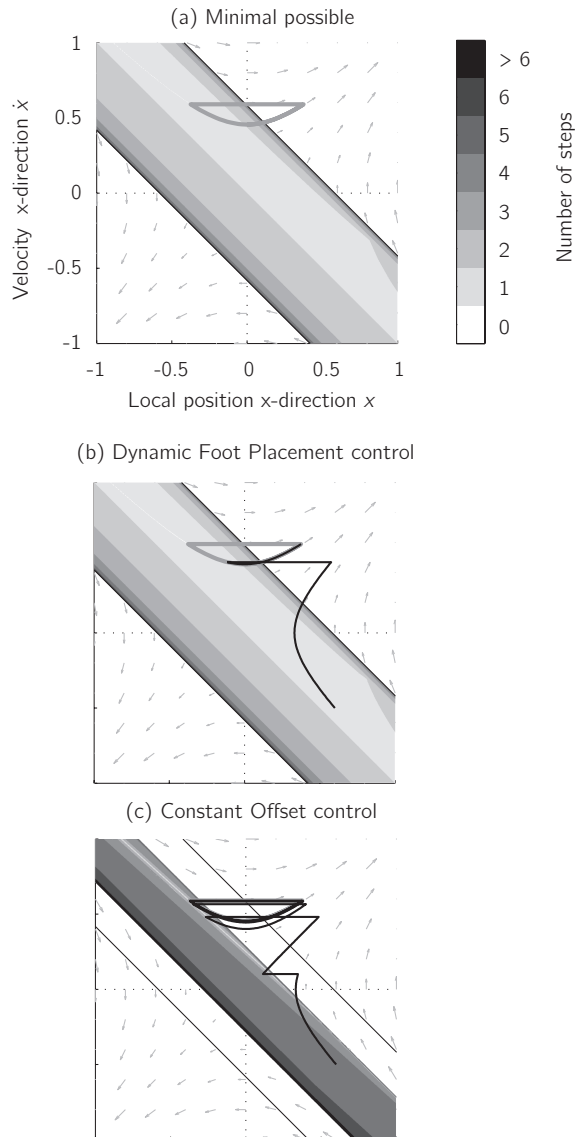


Figure 4.5: State space in the x-direction with the number of steps it takes to reach the desired gait cycle as a function of the initial state. The desired gait cycle is highlighted by the thick grey lines. (a) shows the absolute minimal number of steps it takes to reach the desired gait cycle. (b) and (c) show the number of steps takes for respectively the Dynamic Foot Placement controller and the constant offset controller. For constant offset controller, we define that the gait cycle is reached as the difference with the gait cycle is less than 10^{-3} . The black line in (b) and (c) is an example path for both controllers with an initial state $(x_s, \dot{x}_s) = (0.6, -0.5)$.

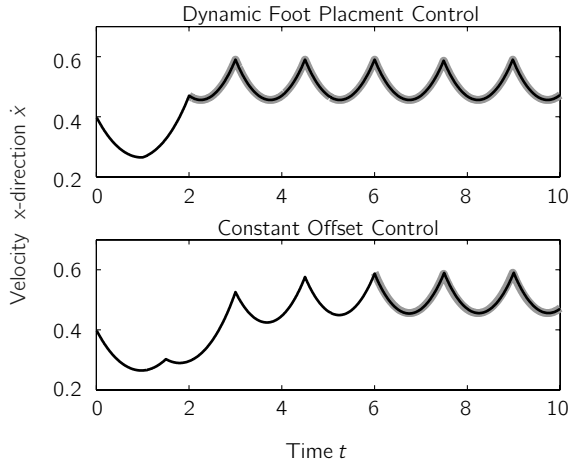


Figure 4.6: The convergence of the velocity in x -direction to the desired limit cycle for the Dynamic Foot Placement controller and the constant offset controller. The desired gait cycle is indicated by the thick grey lines.

Figure 4.5(b-c) show the number of steps that it takes to get to the desired gait cycle for respectively the Dynamic Foot Placement controller and the constant offset controller. The number of steps for the Dynamic Foot Placement controller is almost equal to the absolute minimal number of steps. Only for initial states close to the lower d_∞ boundary, the number of steps is one higher for some initial conditions. The number of steps of the constant offset controller is in theory infinite since the convergence is asymptotical. However for this test, we assume that the gait cycle is reached when the difference with the gait cycle is less than 10^{-3} . With this assumption, the number of steps for the constant offset controller is still much higher than the minimal number of steps. In addition, there is a large range of initial states that are within the ∞ -step viable-capture basin, but do not reach the desired gait cycle using the constant offset controller.

4.6.2 Time response

Figure 4.6 shows an example of how the velocity in x -direction evolves to the desired gait cycle from a disturbed state for both the Dynamic Foot Placement controller and the constant offset controller. The velocity in x -direction is representative for all the four state variables in \mathbf{q} . For the Dynamic Foot Placement controller, the state evolves to the desired gait cycle within a finite number of steps. For the constant offset controller, the state converges asymptotically to the desired gait cycle. This means that the gait cycle will only be reached if time goes to infinity.

4.7 Discussion

The goal of this study is to synthesize a general foot placement controller, which can predict an appropriate step location and step time that will bring a bipedal robot to any desired gait cycle in a minimal number of steps. To arrive at comprehensible control strategies, we used a simple gait model and adopted an incremental approach to solve the control problem. The use of this approach and the model has its benefits but also introduces some limitations.

Incrementally solving the control problem for different N allowed us to derive closed-form expressions for the control parameters. These expressions give direct insight in the relation between the walking objective (in this case a desired walking gait in a minimal number of steps) and the foot placement control (step time and step location) that is required to reach this objective. We demonstrated that a total of only four step strategies was required to effectively bring the model to the desired gait cycle from any feasible model state.

Another benefit from this approach is that the walking objective does not have to be a gait cycle per se. A high-level controller can use the Dynamic Foot Placement controller to track a desired gait pattern, which could include reaching a standstill, changing the walking speed (Figure 4.7), making a turn (Figure 4.8) or reaching any

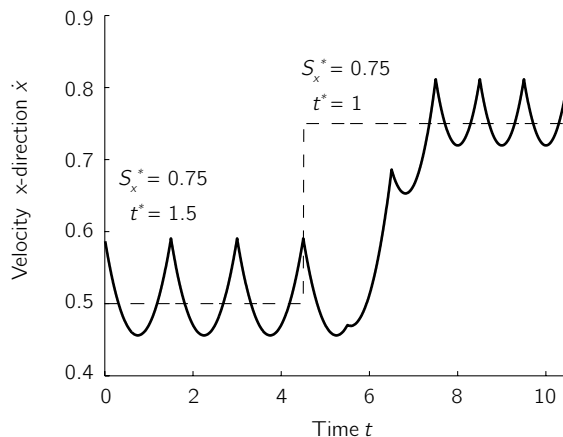


Figure 4.7: Simulation results of a change in walking speed using the Dynamic Foot Placement controller. A speed change is achieved by changing the desired gait parameters. For the depicted speed change, the desired speed S_x^*/t^* is changed from 0.5 to 0.75. The new desired speed is reached within three steps after the desired speed change.

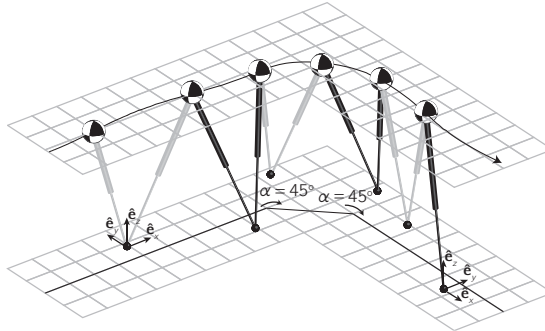


Figure 4.8: Simulation results of a change in walking direction using the Dynamic Foot Placement controller. A change in walking direction is achieved by rotating the local coordinate frame in the x - y plane over an angle α . The depicted 90° turn is in this case achieved by rotating the coordinate frame over 45° after the first step and again 45° after the second step.

other desired system state.

A limitation of the approach is that we optimized our controller to a minimum number of steps, though other optimization criteria might also be relevant. Other relevant optimization criteria could be, for example, minimization of the time required to reach the state or minimization of the amount of mechanical work required by the actuators. In the near future, we plan to study how the foot placement strategies are influenced by the selected optimization criterion or by combinations of criteria.

Though the 3D-LIPM enabled us to derive these simple stepping strategies, it is a very simplistic representation of a real robot. By modeling only center of mass dynamics and not taking impact dynamics into account, we neglect many dynamic aspects of true dynamic walking. However, we believe it can be successfully applied to control a complex biped robot, similar to the successful application of the 3D-LIPM in the walking and balancing algorithms of the twelve-degree-of-freedom biped robot M2V2 (Chapter 7). It should not be attempted to forcefully mimic the 3D-LIPM dynamics on the complex biped, as is typically done for 'Zero Moment Point'-based control approaches (Kajita et al., 2003). The evolution of the complex biped will inevitably deviate from the evolution that is predicted by the simple model. The simple nature of Dynamic Foot Placement controller allows for real-time execution on the robot, which allows the controller to continuously adapt to the current state of the robot. The output of the Dynamic Foot Placement controller can be used as an approximation for an adequate step time and step location for the real robot. Biomechanical studies (Hof et al., 2005, 2010) have shown that such simple foot placement approximations, based on simple inverted pendulum models, show good

correlation with human foot placement strategies.

4.8 Conclusion

In this paper, we presented the Dynamic Foot Placement controller: a controller that outputs the next step location and step time based on the desired gait cycle and the current state of the system. We investigated the performance of this controller in a simulation study and shown that it can get the model to a desired gait cycle from every feasible state within a finite number of steps. In addition, we compared the performance of the Dynamic Foot Placement controller to the frequently used constant offset controller. We showed that the Dynamic Foot Placement controller outperforms the constant offset controller, in both the number of steps it takes to reach the desired gait as well as the size of the region in state space for which it can reach the desired gait.

Appendix

Minimal number of steps

Figure 4.5a shows the absolute minimal number of steps it takes to reach the desired cyclic gait. In this section, we describe the numerical method used to determine the absolute minimal number of steps. The method determines iteratively the set of states A_N for which it is possible to reach the desired gait cycle within N steps based on the set A_{N-1} . Two steps are taken to get A_N from A_{N-1} . First, set A_{N-1} is expanded to B_N by adding all states that can reach A_{N-1} by making a step with a step size smaller or equal than the maximal step size l_{\max} . Next, A_N is determined as all states that will reach a state in set B_N with $t \geq t_{\min}$. The first set B_0 is equal to the final state in gait cycle q_f^* . In pseudocode this process looks as follows:

```

for  $N = 0$  to maximal number of steps do
  if  $N = 0$  then
     $B_N \leftarrow \{q_f^*\}$ 
  else
     $B_N \leftarrow \{q : q - (\mathbf{S}^T, 0, 0)^T \in A_{N-1} \text{ with } \|\mathbf{S}\| \leq l_{\max}\}$ 
  end if
   $A_N \leftarrow \{q : g(q, t) \in B_N \text{ with } t \geq t_{\min}\}$ 
end for

```

in which $g()$ represents the stance phase dynamics as given in (4.2).

5

Mechanical analysis of the preferred strategy selection in human stumble recovery.

T. de Boer, M. Wisse, F.C.T. van der Helm
Journal of Biomechanical Engineering, 2010

Abstract

We use simple walking models, based on mechanical principles, to study the preferred strategy selection in human stumble recovery. Humans typically apply an elevating strategy in response to a stumble in early and midswing, for which the perturbed step is lengthened in a continuation of the original step. A lowering strategy is executed for stumbles occurring at mid or late swing, for which the perturbed swing foot is immediately placed on the ground and the recovery is executed in the subsequent step. There is no clear understanding of why either strategy is preferred over the other. We hypothesize that the human strategy preference is the result of an attempt to minimize the cost of successful recovery. We evaluate five hypothesized measures for recovery cost, focusing on the energetic cost of active recovery limb placement. We determine all hypothesized cost measures as a function of the chosen recovery strategy and the timing of the stumble during gait. Minimization of the cost measures based on the required torque, impulse, power and torque/time, results in a humanlike strategy preference. The cost measure based on swing work does not predict a favorable strategy as a function of the gait phase.

5.1 Introduction

Humans show a remarkable ability to regain balance after large perturbations during gait, such as a stumble due to swing leg blockage. Experimental studies (Dietz et al., 1986; Eng et al., 1994; Forner Cordero et al., 2003; Grabiner et al., 1993; Schillings et al., 1996) have induced stumbles during gait to study the recovery actions that humans employ to regain balance. These studies have shown that the recovery actions following the induced stumble can be categorized into the following strategies: an elevating (or reaching) strategy, where the perturbed foot is lifted and the step lengthened in a continuation of the original step, or a lowering strategy, where the perturbed swing foot is immediately placed on the ground and the recovery step is executed by the contra lateral limb during the next step (Eng et al., 1994).

The kinematic and neuromotor responses for the lowering and elevating recovery strategy are well characterized (Dietz et al., 1986; Grabiner et al., 1993; Pavol et al., 2001; Schillings et al., 1999). It was found that the strategy preference depends on the timing of the induced stumble during gait (Pavol et al., 2001). For early swing perturbations (occurring at 5% to 25% of the step time) subjects show an elevating strategy. For perturbations in midswing (30% to 55%), both lowering and elevating

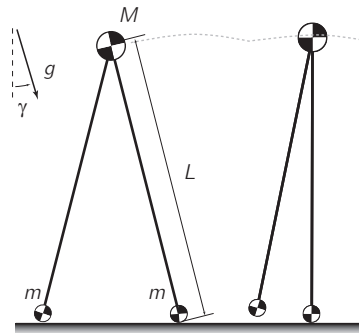
strategies occur. In late swing (55% to 75%), subjects perform a lowering strategy (Schillings et al., 2000). However, there is very little understanding of why either strategy could be favorable at a certain gait phase (Forner Cordero et al., 2005). It is not clear which factors influence a subject's decision to prefer one recovery over the other (Schillings et al., 2000; Weerdesteyn et al., 2005). We believe that the primary reason to prefer one strategy over the other is based on mechanical principles.

When considering the mechanics of stumble recovery, the role of the recovery limb is crucial. Independent of the employed strategy, balance recovery relies on the ability to rapidly perform a recovery step. Surely, postural responses such as control of the arms (Roos et al., 2008), trunk (van der Burg et al., 2005), or the support limb (Pijnappels et al., 2004) also contribute to balance recovery. But the timing, speed and length of the recovery step determine if the newly created base of support is effective in stopping the forward acceleration of the body (van den Bogert et al., 2002; Hsiao-Wecksler and Robinovitch, 2007; Maki and McIlroy, 1999). The ability to perform such a rapid recovery is dependent on muscular performance. Slower and weaker muscular responses, associated with muscular aging effects, can significantly decrease the recovery ability (Hsiao-Wecksler, 2008; Pijnappels et al., 2008; Smeesters et al., 2001; Thelen et al., 1997; Wojcik et al., 2001). Older human subjects rely more often than young subjects on multiple steps to recover (Luchies et al., 1994; McIlroy and Maki, 1996). These findings demonstrate that the limb dynamics and the required muscular performance strongly influence the success of the recovery. We hypothesize that the human strategy preference is based on an attempt to minimize the cost of successful recovery. Since this success depends on the muscular strength required to place the recovery leg, we hypothesize that the recovery cost is proportional to the required peak torque for rapid swing leg placement.

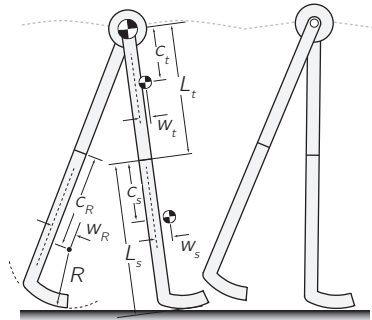
To study if strategy selection can be based on recovery cost minimization, we adopt a simulation approach where we use a simple, inverted pendulum type, walking model. We use this walking model to assess how the employed strategy and the timing of the stumble influence the required peak torque to perform a rapid recovery step. Furthermore, we validate if the mechanical principles of the simple model can be representative to human stumble recovery by using more physically realistic walking models, multiple recovery steps, and different hypothesized measures to describe the cost of recovery.

5.2 Models and simulation methods

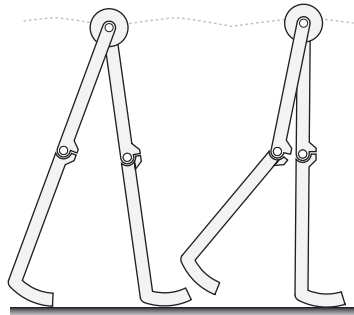
We simulated human stumble recovery in the following straightforward way. We subjected the simple walking model to a stumble at all phases of the gait cycle, from



(a) Simplest walking model



(b) Straight-legged anthropomorphic model



(c) Kneed anthropomorphic model

Figure 5.1: Three dynamic walking models capable of showing passive cyclic gait on slope γ . All models are extended with hip and knee (if applicable) actuation to recover from a stumble. Hip actuation produces a torque that acts on the swing and stance leg. Knee actuation produces a torque that acts on the swing leg thigh and shank. (a) The simplest walking model by Garcia et al. (1998). All variables are described in non-dimensional terms, with the leg length L , hip mass M , and gravitational constant g as base units. Foot mass m is infinitesimally small compared to the hip mass. (b) The straight-legged anthropomorphic equivalent of the kneed anthropomorphic model in (c), adopted from McGeer (1990a). Both models have anthropomorphically distributed mass and a point mass at the hip that represents the torso. See Table 5.1 for the model parameter description and values.

0% to 100% of the step. The stumble was induced by instantaneously blocking and releasing the forward swing foot motion. The stumble triggered simplified lowering and elevating strategies that used a constant hip flexion torque to power the recovery leg. We recorded the required recovery torque as a function of the gait phase and the employed strategy.

Furthermore, we tested if the mechanical principles revealed by the simple model changed when we used more complex simulation methods. We simulated stumble recovery using two anthropomorphic walking models, a more physically realistic stumble implementation and multiple recovery steps. Finally, we evaluated four other hypothesized measures to describe the cost of recovery.

5.2.1 The walking models

A total of three walking models were used throughout this simulation study. All models showed a cyclic passive dynamic gait when powered only by gravity using a slight downhill slope. We extended these passive walking models with actuation in the hip joint and knee joint (if present), but this addition was only used when a recovery action was required in response to a stumble.

To reveal the very fundamental principles of stumble recovery, we primarily used the “simplest walking model” (Garcia et al., 1998) (Figure 5.1a). This irreducibly simple walking model was based purely on mechanical principles. Despite its simple nature, the dynamic principles of this model have proven to provide useful insight into principles of human gait (Hobbelen and Wisse, 2008a; Kuo, 2001, 2002). Furthermore, similar simple models were found to adequately describe the dynamics of the human body during balance recovery (van den Bogert et al., 2002; Hsiao and Robinovitch, 1999; Maki and Mclroy, 1999). It consisted of two rigid links, connected by a frictionless rotational joint at the hip. Point masses were located at the hip and at the feet. The masses at the feet were infinitesimally small compared to the mass at the hip, which resulted in a dynamic motion of the hip that is not influenced by the swing leg dynamics. The dynamics of this model were very simple, comprehensible and required little computational power.

To test if this simple model could be representative for human stumble recovery, we validated the results on two complex models that were more physically realistic. We used the straight-legged and kneed anthropomorphic models adopted from McGeer (1990a) (Figure 5.1(b,c)). These models had identical geometry and anthropomorphically distributed mass, but the kneed anthropomorphic model also included a knee joint and knee latch. This latch prevented knee hyperextension and locked the knee when fully extended during the end of the swing phase and throughout stance. The

models comprised a point mass at the hip that represented the upper body, legs consisting of a thigh and shank and curved feet that were offset forward of the shank.

For all models, we used dimensionless variables scaled with the overall mass M , extended leg length L and gravitational acceleration g . This resulted in time displayed in $\sqrt{L/g}$ and torque as a fraction of MgL . The model parameters for both anthropomorphic models are given in Table 5.1.

The segment angles for a single step of the cyclic gait of all three walking models are shown in Figure 5.2. The simulation of a step consisted of continuous motion of both legs in between heel strikes, followed by a discrete velocity change due to the knee stop collision at full extension (for the kneed model) and the heelstrike collision at the end of the swing phase. Both events were modeled as a fully plastic (no slip, no bounce) instantaneous impact. After the knee stop collision, the simulation progressed with a straight swing leg. Heelstrike collision was triggered when the

Table 5.1: Model parameter description and values for both anthropomorphic models as shown in Figure 5.1(b,c) (identical to McGeer (1990a)). Mass parameters are presented as a fraction of the total body mass M , length parameters as a fraction of the extended total leg length L .

Model Parameter	Symbol	Value
Length of thigh	m_t	0.46
Length of shank	m_s	0.54
Mass at hip	m_h	0.676
Mass of thigh	m_t	0.100
Mass of shank	m_s	0.062
Vert. offset thigh center of mass (CoM)	c_t	0.2000
Vert. offset shank CoM	c_s	0.2372
Vert. offset foot center	c_r	0.3332
Hor. offset thigh CoM	w_t	0
Hor. offset shank CoM	w_s	0.0379
Hor. offset foot center	w_r	0.0675
Foot radius	R	0.2

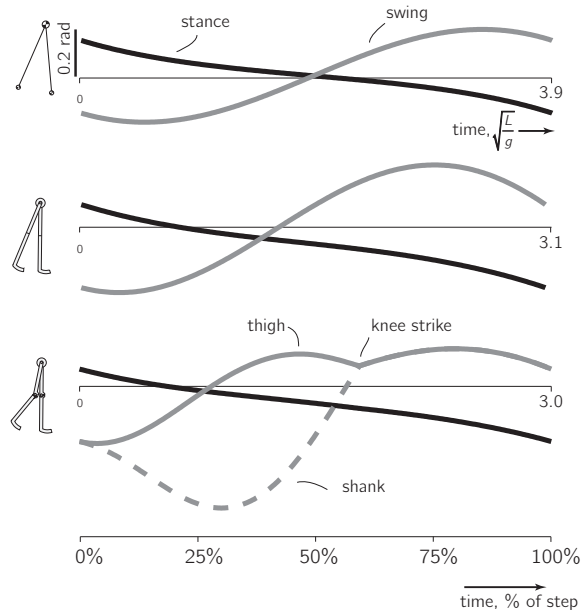


Figure 5.2: For all three models, the segment angles are shown for one step as a function of the step time. The angles are measured counterclockwise with respect to the slope normal. During a step, only one foot is in contact with the ground at any time; double support occurs instantaneously at heelstrike. The gait of the simplest walking model is a result of a slope angle γ of 0.004, following the work of (Wisse et al., 2005b). For the anthropomorphic models, a slope angle γ of 0.0075 is selected that results in stable gait for both models. The graphs are scaled with respect to the gait phase, expressed as a percentage of the total step time.

swing foot touched the ground at the end of the swing phase, which set the initial conditions for the next step. We ignored ground contact of the swing foot prior to midswing for the straight-legged models, following (Garcia et al., 1998; McGeer, 1990a). This concession was made to avoid the inevitable foot-scuffing for the two dimensional models without a knee.

5.2.2 The induced stumble

To induce a stumble, we simulated an instantaneous collision that blocked the forward velocity of the swing foot. At the initiation of the recovery action, the foot was free to move forward again. This obstruction implementation resembles the method used in human stumble experiments to block the swing leg motion with a long cord attached to the heel (Dietz et al., 1986; Forner Cordero et al., 2003) or a stumble

that occurs due to a small size obstacle or foot-scuffing. Note that for the kneed anthropomorphic model, we allowed the kneelock to disengage at the occurrence of a stumble.

To simulate the effect of a more physically realistic and more severe stumble, we applied the swing foot blockage over a longer period of time. This introduced a delay between the perturbation onset and the recovery initiation, simulating the effect of a response time.

5.2.3 The recovery

Each model executed a powered recovery step to recover from a stumble. During the recovery step, the actuator(s) applied a constant torque to the joint(s), simulating the case that the actuators used their maximum strength throughout the recovery in their attempt to accelerate the leg as fast as possible (as also observed in human balance recovery (Wojcik et al., 2001)). For the elevating strategy, the recovery step was the remainder of the perturbed step. For the lowering strategy, the recovery step was the step that followed after the lowering phase, which was the placement of the obstructed foot. The different strategies are illustrated in Figure 5.3 for the kneed model.

To simulate a multiple-step strategy, we allowed the model to perform multiple powered recovery steps after the employed lowering or elevating strategy. The hip torque magnitude was constant during the first and subsequent recovery steps of this multiple-step strategy.

The models could not employ a lowering strategy at all phases of the gait. Prior to mid stance, the lowering phase of the lowering strategy would require extension of the swing leg length, simulating ankle plantar flexion. The simple kinematics of our models did not allow this. After mid stance, implementing a lowering phase was possible using the passive dynamics of the model: we allowed the swing foot to move downward only, driven by the continuing forward stance leg motion. During the lowering phase of the kneed model, a knee torque was required to ensure full knee extension prior to heelstrike (illustrated in Figure 5.3b).

We considered a recovery action to be successful when it returned the model to a stable state at the end of the recovery step. A state was classified as stable when the model converged back to its cyclic gait during the consecutive steps, relying only on its passive dynamics. The method to determine torque magnitudes that result in such a stable state was as follows. Let us consider the most simple case, which is stumble recovery of the simplest walking model. For this model, the state at the end of the recovery step is only a function of two independent variables: the stance leg

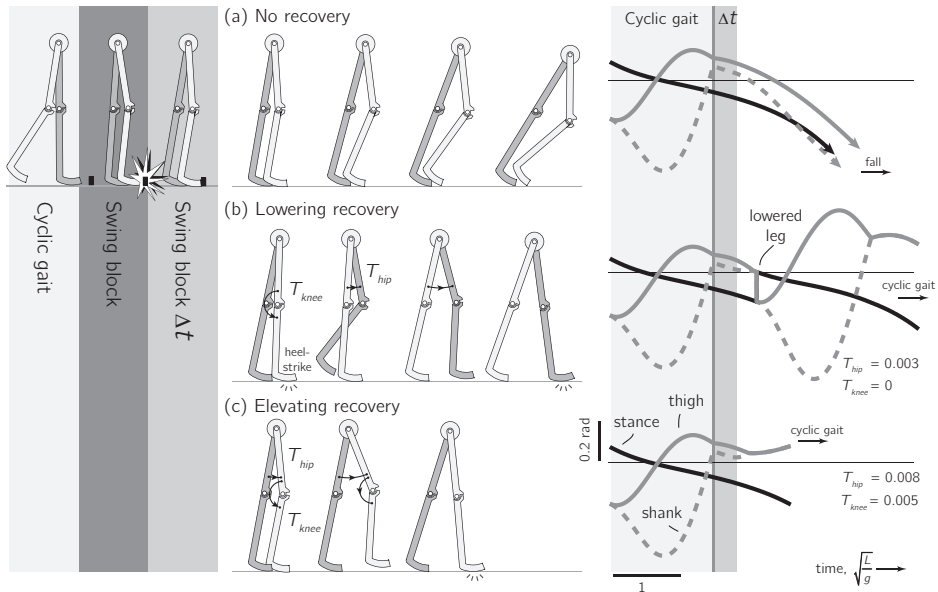


Figure 5.3: A graphical representation of three recovery scenarios for the kneed anthropomorphic model. Angles versus time are shown for each recovery scenario together with a cartoon showing an impression of the walker's behavior. The data is generated for an induced stumble occurring at 58% of the step time, at which time the forward swing foot motion is blocked for a duration of 15% of the total step time. (a) A recovery action is required to prevent a fall. (b) The lowering strategy consists of two phases: The first phase consists of rapid placement of the perturbed step. In this phase, the kneed walking model requires a knee torque to extend the leg prior to heelstrike (in this example $T_{knee} = 0.001$). The second phase consists of a powered recovery step of the new swing leg. Hip and knee actuation can be used to ensure proper foot placement with a fully extended leg. The illustrated example requires only hip actuation. (c) The elevating strategy consists of a powered elongated perturbed step. Hip and knee actuation are used for recovery.

angle θ and angular velocity $\dot{\theta}$ (Garcia et al., 1998). Figure 5.4 shows the Poincaré map of the system at this instant. A stable state lies on the fixed point (i.e., the limit cycle) or within the Basin of Attraction (Wisse et al., 2005b): a state inside this basin will asymptotically converge to the fixed point during the consecutive steps using only passive dynamics. A (perturbed) state that lies outside this basin will result in a fall. We determined the range of recovery torque magnitudes that prevent such a fall by returning the state back into the Basin of Attraction (as illustrated in Figure 5.4 for a single recovery).

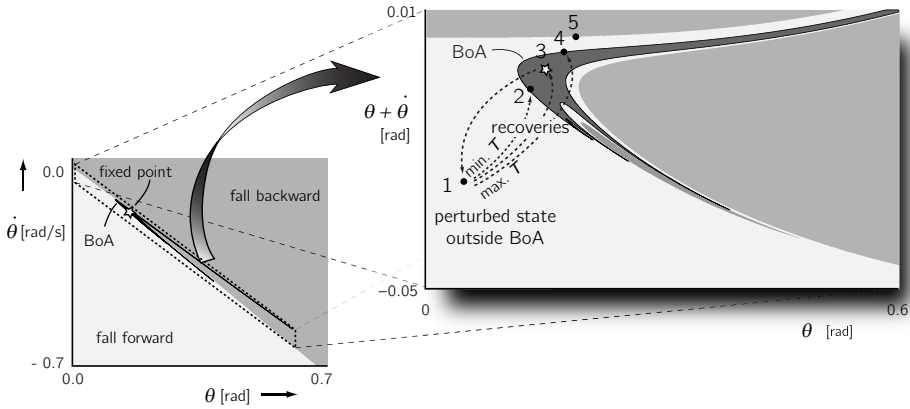


Figure 5.4: The Poincaré section for the simplest walking model just after heelstrike, for a slope angle of $\gamma = 0.004$ (Wisse et al., 2005b). The failure modes, Basin of Attraction (BoA) and fixed point (star at $(\theta, \dot{\theta}) = (0.1534, 0.1516)$) are shown. The right graph shows an enlarged and sheared section which better visualizes the Basin of Attraction. A single lowering recovery is shown. Swing foot blockage results in a shortened step: The state of the model can now be represented by point number 1. The walker would fall forward if no recovery step would be performed. Recovery torques can place the state back towards the fixed point (point 3) or just within the Basin of Attraction (point 2 and 4), or lead to a fall if the torque magnitude is too large (point 5) or small. After a successful recovery, the model will passively converge back to its cyclic gait in the subsequent steps.

5.2.4 The hypothesized cost of recovery

We considered a total of five hypothetical cost measures for the case of stumble recovery. The selected measures were typically used in evaluation of energetic cost of muscle activity in human walking (Alexander, 1989; Kuo, 2001).

We hypothesized that the recovery cost was proportional to the required peak recovery torque for swing leg placement (i.e., the magnitude of the required constant recovery torque). Another possibility was that recovery cost was proportional to the torque divided by the duration that the torque is applied, since it was found to correlate well with the energetic cost of forced swinging of the swing leg in human walking (Kuo, 2001), in human running (Kram and Taylor, 1990) or in isolated swinging of the leg in an experimental setup (Doke and Kuo, 2007). We also studied proportionality to the integral of the torque over the duration, i.e. the impulse (Kuo, 2001). Furthermore, we determined the required peak power for successful recovery, assuming that not only peak torque is important, but also peak angular velocity of the swing leg. Finally, we evaluated proportionality to a combination of positive and negative swing work.

5.3 Results

5.3.1 Recovery of the simple model

For the simplest walking model, Figure 5.5 shows all possible torque magnitudes that result in successful recovery from a stumble at different gait phases. The model recovered passively from stumbles occurring in early and late swing. A recovery action was required for stumbles induced between approximately 20% to 80% of the step time. At approximately 65% percent, the model required the largest recovery torque for the elevating strategy. At this gait phase, the swing foot velocity prior to the blockage was relatively high (85% of its maximum velocity) and the recovery time of the perturbed step was relatively small (6% of the nominal step time), which

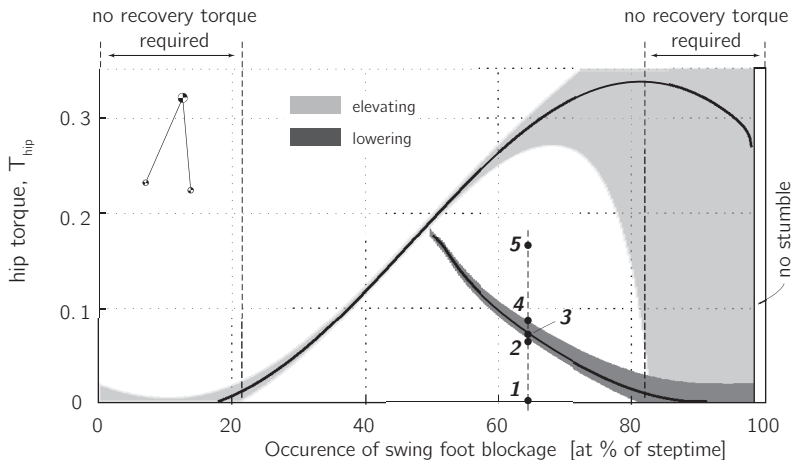


Figure 5.5: Stumble recovery for the simplest walking model (slope angle $\gamma = 0.004$) using an elevating and lowering strategy. Both contour maps indicate the torque magnitudes that result in successful recovery from an instantaneous swing foot blockage. The horizontal axis shows the timing of the induced stumble during a step, from swing initiation (at 0% of the step time) to heelstrike (at 100% of the step time). Just before heelstrike, no stumbles can be induced by swing foot blockage because the model shows a negative foot velocity. The vertical axis shows the applied constant hip recovery torque. The solid black lines represent recoveries where the recovered state ends up onto (or closest to) the fixed point. To explain the graph, five points are shown that represent different lowering recoveries in response to a stumble at approximately 65% of the step. All recoveries employ different recovery torque magnitudes. (Corresponding points can be found in Figure 5.4). If the applied recovery torque is too low (1) or too high (5), the model will fall within the current or the consecutive steps. Minimal (2) or maximal (4) successful recovery torque will just result in recovery. If the applied recovery torque is just right (3), the model state returns onto (or closest to) the fixed point.

resulted in a high required recovery torque.

In contrast to the elevating strategy, the minimum torque requirement for the lowering strategy decreased as the timing of the induced stumble moved from mid to late swing. This can be explained by the fact that the length of the aborted step increased when the timing of the induced stumble approached the nominal step time. This longer step resulted in a lower stance leg velocity during the next step, which in its turn decreased the required recovery torque.

5.3.2 Recovery of the complex models

Figure 5.6 shows the recovery torque requirements for both anthropomorphic models. The increased complexity of both models lead to less smooth contour maps compared to the simplest walking model. The straight-legged model appeared to be susceptible to failure due to foot-scuffing just after midswing (which was not ignored, contrary to foot-scuffing prior to midswing). For the kneed model, we clearly observed the influence of the knee joint and its latch.

The results showed quantitative differences in the required recovery torque between the simple and more complex walking models. The difference in model geometry and mass distribution lead to different torque requirements, as expected. However, substantial qualitative similarities were observed. Again, we observed that the most challenging stumbles were around midswing and that the predicted recovery cost for late swing stumbles was lower for the lowering strategy compared to the elevating strategies.

5.3.3 Increased obstruction duration

We obstructed the forward swing foot motion over longer periods of time. An example case is shown in Figure 5.7. The longer duration of the perturbation resulted in a decreased time to recover, which in turn resulted in an increased torque requirement. However, the relative difference in torque requirement between the two strategies remained.

The duration of an elevating recovery was typically shorter compared to a lowering strategy 5.3. Therefore, the influence of the longer perturbation duration on the total recovery time was more significant for the elevating strategy compared to the lowering strategy.

5.3.4 Multiple-step recovery

Figure 5.8 shows an example case of successful two-step recoveries for the simplest walking model. Both contour plots showed a division into two branches: one branch

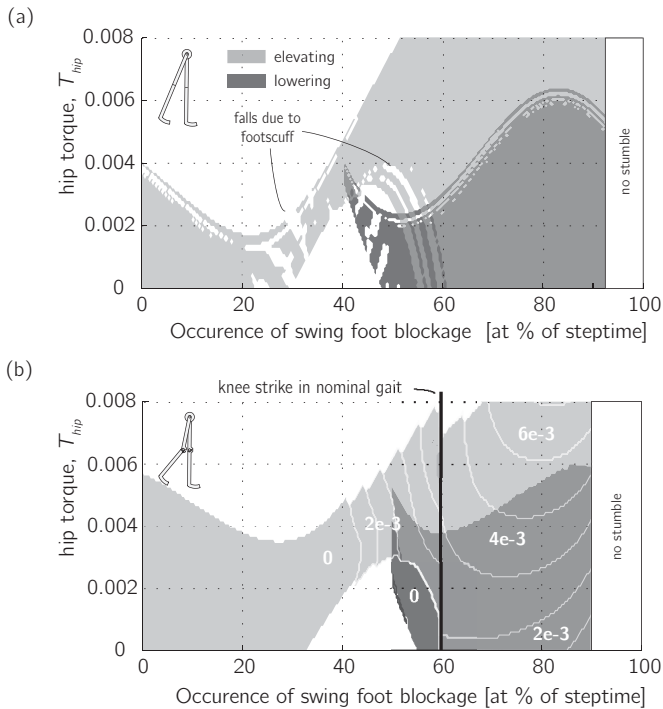


Figure 5.6: Stumble recovery for the anthropomorphic walking models (slope angle $\gamma = 0.0075$) using an elevating and lowering strategy. The two superimposed contour maps indicate the torque magnitudes that result in successful recovery from an instantaneous swing foot blockage, as in Figure 5.5. (a) For the straight-legged model, the minimal amount of ground clearance of the swing foot frequently lead to failed stumble recovery due to foot scuffing. (b) For the kneed model, knee actuation may be required to stretch the leg prior to heelstrike. Patches, bounded by white is curves, indicate the recoveries that have equal knee torque requirements. The numbers in white indicate the magnitude of the required knee torque. The lowering phase of the lowering strategy required a constant torque of ($T_{knee} = 0.001$).

was characterized by recovery torques that were mostly higher than the single step recovery torques and one branch was characterized by torques that were mostly lower. The higher recovery torques of the upper branch represented an overpowered recovery which resulted in a large maximum swing leg angle during the recovery step. The lower branches showed lower torques than the previous “single-step” strategies and are therefore most relevant in this study on cost minimization.

Multiple-step strategies over more than two steps resulted in almost identical results: increasing in the allowed number of steps to recover did not result in a further decrease in the required recovery torque. For both the multiple-step lowering strategy

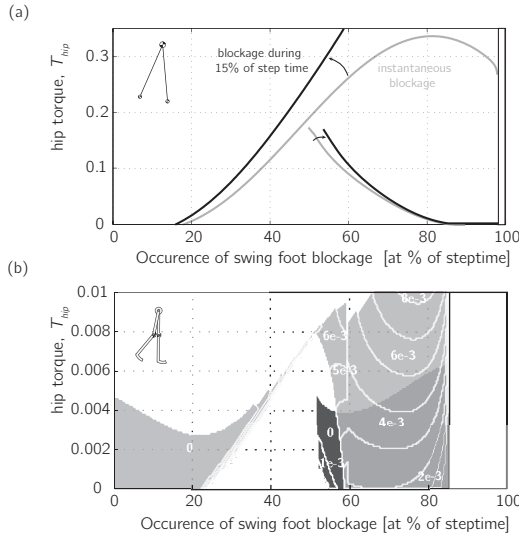


Figure 5.7: Example recovery scenario in response to a swing foot obstruction applied for a duration of 15% of the step time. (a) For the simplest walking model, as Figure 5.5. The effect of the delay between the perturbation onset and recovery initiation is shown in comparison to an instantaneous blockage. For clarity, we only present torque magnitudes for recovery towards the fixed point. (b) For the kneed anthropomorphic model, as Figure 5.6b.

and the multiple-step elevating strategy, the required recovery torque magnitude is close to the magnitude of the single-step lowering strategy.

5.3.5 Hypothesized cost measures

The results from the simplest walking model were used to calculate all hypothetical measures for cost of recovery. Figure 5.9a shows the peak torque profiles derived from Figure 5.5. As expected, the cost proportional to the torque divided by burst duration (Figure 5.9b) increased for short burst durations, as observed for the elevating recovery towards late swing. In contrast to that, the cost proportional to the integral of force over time logically decreased for short burst durations (Figure 5.9c). The hypothesized cost measure proportional to peak power (Figure 5.9d) showed a clear slope change for the elevating strategy in late swing. This was due to a transition from a positive to a negative directed velocity of the recovery leg, since the effect of the gravitational moment acting on the leg increased for larger leg angles towards late swing. This also caused an increase of negative swing work for the elevating strategy in late swing (Figure 5.9e). All but the cost proportional to swing work predicted a favorable lowering strategy from mid to late swing. Focusing

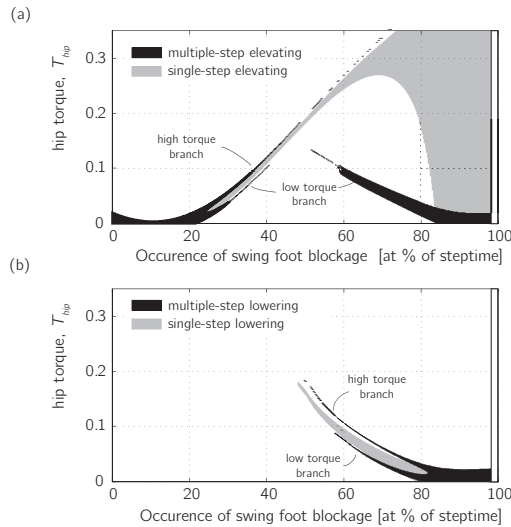


Figure 5.8: Multiple-step recovery for the simplest walking model. The contour map of a two-step strategy is superimposed on the contour map of the corresponding single-step strategy from Figure 5.5. The vertical axis shows the applied constant torque that is applied to the hip during the first recovery step and the following step. The contour map has different branches, for recoveries that have a higher or lower torque requirement compared to the single-step recoveries.

on the positive swing work (when assuming a higher energetic cost for positive work compared to negative work (Abbott et al., 1952)), the cost of recovery proportional to swing work did not show a clear favorable strategy.

5.4 Discussion

The purpose of our research was to study if the human strategy preference in stumble recovery is based on cost minimization, which we assumed to be proportional to the required peak torque for recovery.

All three walking models show that the required recovery torque to overcome stumbles induced in early swing is relatively low (or even zero). Stumbles occurring at mid-swing are most demanding (which is in agreement with human stumble experiments (Forner Cordero et al., 2003)). At this point, the elevating and lowering strategy are fairly similar in their torque requirement. For late swing stumbles, the torque of a lowering strategy is significantly lower compared to an elevating strategy.

A humanlike strategy preference emerges if the models would always execute the strategy that minimizes recovery cost based on the required recovery torque. This

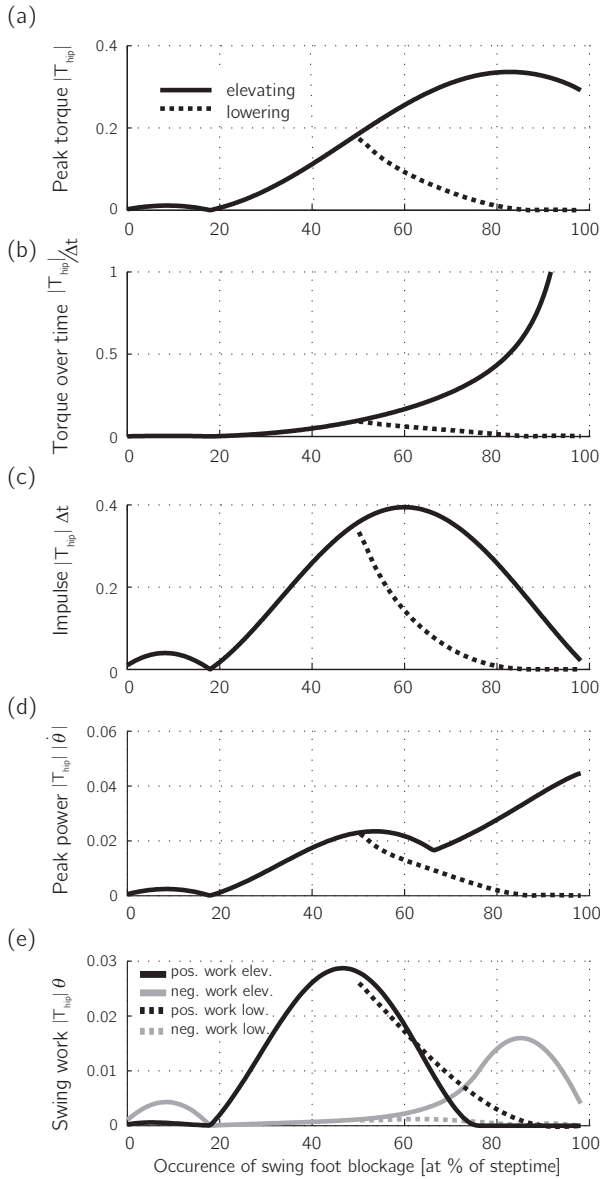


Figure 5.9: Hypothetical measures for cost of recovery for the simplest walking model, as a function of the strategy and timing of the stumble. Cost of recovery hypothetically proportional to: (a) The absolute magnitude of peak recovery torque. (b) The torque divided by the duration of the recovery Δt . (c) The torque integrated over the duration of the recovery step, i.e. the impulse. (d) The peak recovery torque multiplied by the maximum of the recovery leg angular velocity $\dot{\theta}$. (e) The amount of swing work performed: the torque multiplied by the angular displacement of the swing leg.

also holds for the cost models based on proportionality to impulse, torque divided by time and power. The hypothesized measure based on swing work does not show a clear advantageous strategy. However, we have reason to suspect that a measure based on swing work alone is not a good cost model for rapid swing leg placement. Doke and Kuo (Doke et al., 2005; Doke and Kuo, 2007) showed that the energetic cost of swing leg motions is a function of swing work, but mostly dominated by cost in proportion with torque, and in inverse proportion to duration of torque for increasing swing leg speed. Our cost measures based on torque and torque duration show a very significant difference between both strategies. We therefore reason that strategy selection in stumble recovery may indeed be based on cost minimization. Based on this simulation study, we cannot determine which measure describes the cost of active swing leg placement best.

The relative simple nature of our models defines their strengths and limitations. A limitation is that the kinematics and dynamics of our walking models are too simple to give a realistic approximation of the real magnitude of the hypothesized cost measures. We can only predict the relative cost difference for the different strategies and stumble occurrences. Furthermore, the measures are all based on a single variable, i.e. hip torque for hip flexion, to realize swing leg placement. In reality, this is a combined effect of multiple mechanisms that all contribute to adequate placement of the recovery leg. These for example include knee extension, ankle plantar flexion (Pijnappels et al., 2004), arm movements (Roos et al., 2008) and trunk movements (van der Burg et al., 2005). Inclusion of all these individual mechanisms will lead to improved estimates for the hypothesized recovery costs. It will also allow us to study their contribution to the success or failure of the recovery. Inclusion of the knee joint already revealed more humanlike failure modes compared to the straight-legged models, which could only fall due to wrong recovery leg placement. Inducing a severe stumble on the kneed model revealed that recovery could only be achieved using a recovery leg that had a flexed knee, since the hip height at the end of the recovery was significantly decreased. Allowing such a recovery introduces failure mechanisms such as recovery limb buckling and improper support limb loading, which are both important failure mechanisms in human stumble recovery (Pavol et al., 2001).

The strength of the models is that they allow us to consider limb kinematics and dynamics during the unperturbed and perturbed gait, which is different from other models used in balance recovery studies (van den Bogert et al., 2002; Hsiao and Robinovitch, 1999; Maki and McIlroy, 1999). This gives us the possibility to demonstrate the relation between the timing of the perturbation, the reaction time, the employed strategy, the required strength and the stability of the recovery. Therefore, we believe our simulation study provides a valuable first approach towards a

clear understanding of the fundamental principles of stumble recovery. The models may also be helpful in creating more insight in the mechanisms specifically related to age-related declines in recovery ability. For example, our simulation results showed a significant reduction in the required recovery torque for a multiple-step elevating strategy in late swing compared to the single-step elevating strategy. This suggests that execution of a multiple-step recovery is required when the actuator strength is insufficient to perform a single step elevating recovery. This may be related to the observed behavior of human subjects when regaining standing balance using an elevating type step in response to an induced forward fall: older human subjects rely more often than young subjects on a multiple-step recovery, which was conceptually associated with muscular aging effects (Luchies et al., 1994; McIlroy and Maki, 1996).

5.5 Conclusion

Simple walking models are used to study if the preferred strategy selection in human stumble recovery is based on an attempt to minimize recovery cost. The hypothetical cost measures for swing leg placement proportional to the required peak torque, power, impulse, and torque divided by time, predict a humanlike strategy preference as a function of the gait phase. The cost measure proportional to swing work does not predict a favorable strategy as a function of the gait phase. We reason that the cost of rapid swing leg placement is likely dominated by the cost of producing torque, rather than swing work. Therefore, the results suggest that the preferred strategy selection in human stumble recovery is based on an attempt to minimize recovery cost.

Appendix

All simulation models of this research can be downloaded on the website of the Delft Biorobotics Laboratory. When entering the website, click on 'walking robots' and subsequently on 'TUlip'.

Acknowledgements

We thank S.H. Collins and two anonymous reviewers for editorial and technical comments. The work presented in this article has been carried out with financial support from the Commission of the European Union, within Framework Programme 6, RTD programme IST, under contract no. FP6-2005-IST-61-045301-STP.

6

Robot prototype: design

Parts of this chapter have previously been published in:

D. Hobbelen, T. de Boer, M. Wisse,
“System overview of bipedal robots Flame and TULip: Tailor-made for
Limit Cycle Walking”, *Proceedings of the International Conference on
Intelligent Robots and Systems*, 2008

6.1 Introduction

Balance control is in essence about the ability to control the forces that act on the CoM by controlling the CoP with respect to the CoM. This ability obviously depends on both the controller of the system and the mechanics of the system. This chapter considers the mechanical aspect of walking and presents the newly developed humanoid robot prototype 'TUlip' (depicted in Figure 6.1).

Previous chapters have demonstrated how balance can be controlled by the stepping strategy, ankle strategy and hip strategy. These strategies requires accurate and fast positioning of the swing foot and the ability to control joint torques in order to control the center of pressure location underneath the stance foot. Logically, the realization of robotic hardware that can demonstrate these capabilities has been our priority in this research.

However, it has to be taken into account that the initial design of TUlip was not realized with these requirements in mind. The initial design was based on the experience and knowledge gained through research on Limit Cycle Walking (Hobbelen and Wisse, 2007b). Unfortunately, the initial ideas on the control and mechanics of walking that followed from this Limit Cycle Walking approach lead to unsatisfactory performance when applied to TUlip. This result lead to a radical change in our view on the control and mechanics of walking and therefore made us change the adopted research approach (of which the result can be read in this thesis). Consequently, the design of TUlip was also changed during the research period pertaining to this thesis.

This chapter starts with an overview of the evolution of the design of TUlip in Section 6.2, describing how the design has changed and why. In Sections 6.3 to 6.5, a detailed overview of the current robot hardware is given in terms of the mechanics, electronics and software. Special attention is paid to the actuator design and the leg design, since their design is most important for the ability to control balance in bipedal locomotion.

Since the overall performance of the system depends on the implemented controller, Chapter 7 presents the developed control algorithms. Chapter 8 presents several robot experiments and evaluates the overall robot performance.

6.2 Robot evolution

The development of TUlip started in 2006, at the onset of the research described in this thesis. Its initial design was tailor-made for Limit-Cycle Walking (Hobbelen and De Boer, 2008). This approach has been used for the analysis and control of walking

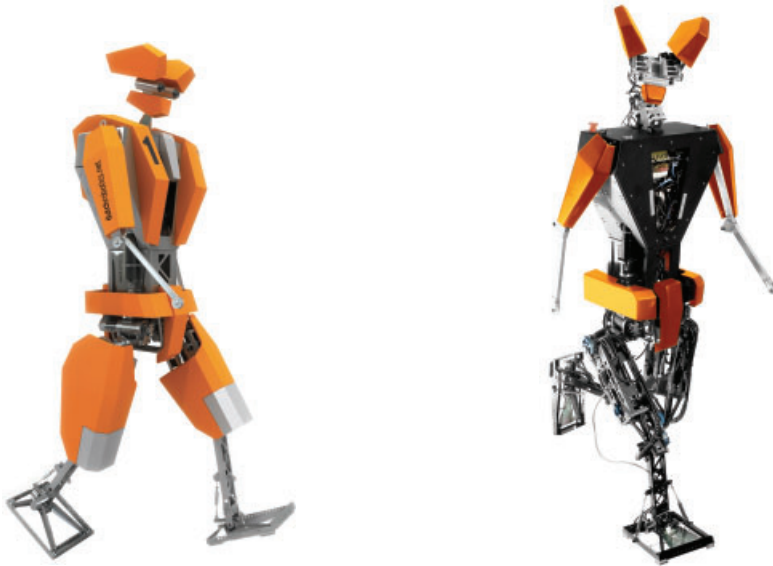


Figure 6.1: Design and realization of the humanoid robot TULip (year 2008).

in the Delft Biorobotics Laboratory since 1995. The experience gained through the development of about ten walking robot prototypes (Delft Biorobotics Laboratory, 2011) provided a basis for the design of TULip. The design of TULip shares many features with its predecessor Flame (Hobbelen and De Boer, 2008). The point of departure was to develop a robot that had a mechanical structure that was naturally conducive to walking and required only low levels of control for stabilization. This approach had already led to life-like and energy-efficient cyclic gait on previous prototypes (Hobbelen and Wisse, 2007b; Wisse et al., 2007). The idea was to synthesize controllers that would intervene in this energy-efficient gait in case disturbances occurred or when more versatile, aperiodic behavior would be desired. To stimulate the development in this direction, the goal was set to successfully compete in the robotic soccer competition of RoboCup (RoboCup, 2011). The concepts of the Limit Cycle Walking approach, as well as the participation in robot soccer matches, have influenced the design of TULip in the following way.

A key concept of the Limit Cycle Walking approach is to exploit the potentially stabilizing properties of the passive (i.e., unactuated) dynamics of the robotic system. Based on this concept, we chose to implement compliant force/torque controllable actuators in TULip which allow for a passive swing leg motion (through zero torque control). Also, the lateral stance phase motion was made passive through the appli-

cation of a passive-compliant ankle roll joint.

Competing in robotic soccer games requires robots to be truly advanced. As a first step towards this goal, effort was put into making TULip autonomous, and lightweight and mechanically robust to survive a fall. Also, we selected a human-like joint configuration and range of motion for the legs to enable versatile behaviors, such as steering, kicking a ball and standing up when fallen down. Robotic soccer also requires enough processing power to support a software architecture for control software, vision processing, world modeling, and decision making.

In 2008, TULip became operational. Two other copies were also realized and adopted as research platforms by the other technical universities in the Netherlands, Eindhoven University of Technology and the University of Twente. Despite combined effort, a stable walking gait was not achieved for any of the prototypes. After lots of tuning, the best walking performance was achieved in 2009 (Figure 6.2), which still required operator intervention to prevent a fall.

After these disappointing results, we decided to completely overhaul the robot, both in hardware and in software. In short, this involved the following. Regarding the mechanics, the repeatability, predictability and controllability of the behavior had to be increased, for example by reducing friction, play, and elasticity, and by thorough



Figure 6.2: Images showing TULip performing a stride of a cyclic walking gait (Takahashi, 2009). The closeness of the operator gives away that small interventions are required every few steps to prevent a fall. Images are stills from the video visible on the Delft Biorobotics Laboratory website (Delft Biorobotics Laboratory, 2011).

system identification, and calibration. Regarding the electronics, some of the amplifiers were replaced for more powerful ones and the allowed motor current was increased (above the rated current) to increase the motor performance. Regarding the software, effort was put into the development of software that could log, monitor and modify variables on the robot in real-time to allow for better debugging and tuning. Regarding the control software, a shift was made towards the control approach described in this thesis. Limit Cycle Walking control offered no feedback mechanisms to assess the gait robustness, which offered no systematic approach (but trial and error) to improve the gait. Also, the control consisted of multiple joint based controllers that were tuned without having a clear understanding of their effect on the rest of the system. Therefore, we chose to start the design of controllers based on relevant state indicators such as the CoP and the location and the velocity of the CoM, and to controllers that control multiple joints and take the kinematics of the whole system into account (see Chapter 7 for more information).

The above described evolutionary process of the robot demonstrates not only that working with real robot hardware can be time consuming. It also demonstrates the value of this prototype within this research and the impact it had on the project. Working with the robot has given us insight in the relevant issues in the mechanics and control of walking, both for robots and for humans.

All together, the current realization of TULip can be considered an innovative bipedal robot prototype. The large scale application of force controllable actuators in its legs sets it apart from many of the currently existing humanoids and offers implementation of novel force-based control approaches. Only few humanoid robots have similar abilities, such as the Sarcos robot (Raytheon Sarcos, 2011), the M2V2 robot (Pratt et al., 2009) and the recently developed DLR-biped (Ott et al., 2010) and the cCub (Tsagarakis et al., 2011).

6.3 Mechanics

For the mechanical design, the initial design requirements were as follows:

- application of actuators that offer force control and position control;
- low mass moment of inertia of the legs with respect to the upper-body, enabling fast swing leg; motions
- human-like leg morphology, enabling human-like versatile behavior;
- low overall weight, enabling low power motors that consume little energy.

The resulting mechanics are now described, with a strong focus on the design of the actuators and their implementation in the robotic structure.

6.3.1 Actuation

The main choice of actuation for the robot was series elastic actuation, introduced by Pratt and Williamson (1995) and successfully implemented in various other robotic applications (Pratt et al., 2001; Veneman et al., 2006). Series elastic actuation comprises a motor which connects to the joint through a compliant elastic element. This system is selected because of the following properties.

Firstly, measuring the elongation of the compliant element allows for determining and controlling the amount of force/torque that is exerted by the motor on the joint. The used control scheme is schematically shown in Figure 6.3. Initially, the ability to control joint torque was deemed necessary because the control of zero joint torque can leave the passive dynamics of the system intact and thus enable the proper application of Limit Cycle Walking control. Currently, we believe force control is most beneficial for balance control because it allows for control of the interaction forces with the environment to regulate the CoM motion.

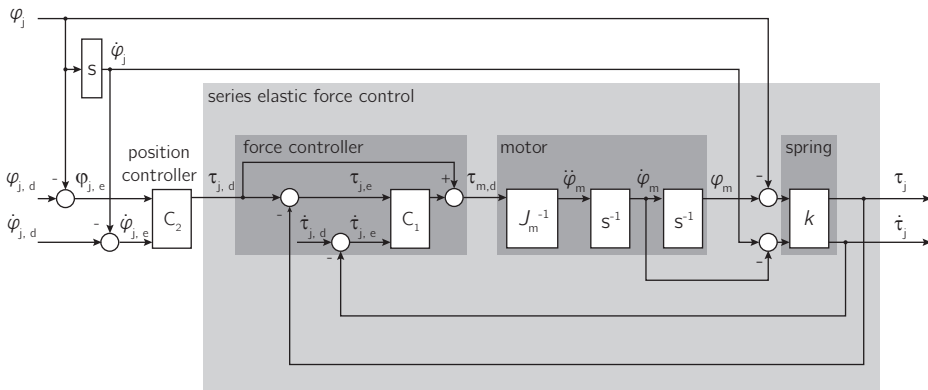


Figure 6.3: Control scheme of a series elastic actuator as used in TULip. A motor (with inertia J_m , angle ϕ_m and input torque τ_m) is connected to the joint (index 'j') through a compliant element with stiffness k . The spring torque τ_j acting on the load is proportional to the measured difference between the motor angle ϕ_m and joint angle ϕ_j . The torque applied to the joint is controlled by a feedback and feedforward controller that outputs the desired input motor torque (i.e., current) based on the error between the current joint torque and the desired torque $\tau_{j,d}, \dot{\tau}_{j,d}$. The desired joint torque can be set by a high level force controller (e.g., a Virtual Model Controller as described in Chapter 7) or a position controller. The latter is shown as an example: the inner force controller uses the output of an outer position controller as an input.

Secondly, series elastic actuation is shock tolerant. When impacts occur in the robot (and this does happen frequently in walking), the series elasticity protects the motor and especially the gears in the gearbox.

Finally, series elastic actuation enables measurement of the energy/torque on the joint level, excluding the energy consumed by the motor. This allows us to compare torque/energy of our gait with other robotic or human gait, irrespective of the motor type used.

The following paragraphs will describe how the concept of series elastic actuation is implemented in TULip and present its performance.

Realization of series elastic actuation

Motor selection

In the series elastic actuation system, we chose to use a common geared brushed DC motor. The use of electric DC motors generally allows high control bandwidth with relatively little system overhead. In comparison to pneumatic or hydraulic actuators this makes the electric motor an attractive candidate for an autonomous robot. The application of a geared motor instead of direct drive is chosen because in case of direct drive the magnitude of the desired torques in our robots would require a heavy motor with overly high power rating. Typically, a geared motor has a low force fidelity and high impedance, due to a high reflected inertia and damping/friction. However, through the integration of the motor in the series elastic actuation system, these undesirable property can be solved. The low output impedance of the series elastic actuator is illustrated in the frequency domain in Figure 6.4 and in the time domain in Figure 6.5. The dynamic effects of the motor inertia and gearhead friction are nearly invisible at the output of the series elastic actuator.

The power and gear head ratio of the geared DC motors is selected based on prior experience with the robot Flame (Hobbelen and De Boer, 2008). The motor specifications are shown in Table 6.1 on page 121.

Compliant element selection

We chose to connect the DC motor to the joint using linear springs in series with steel cables. This configuration showed satisfactory performance on other robotic prototypes (Hobbelen and De Boer, 2008; Hobbelen and Wisse, 2008b; Veneman et al., 2006), while being mechanically relatively simple. Furthermore, the use of cables offered the possibility to place the actuator at a different location and orientation than the joint axis, which facilitated the design of a leg with a relatively low moment of inertia, as will be described in the next section.

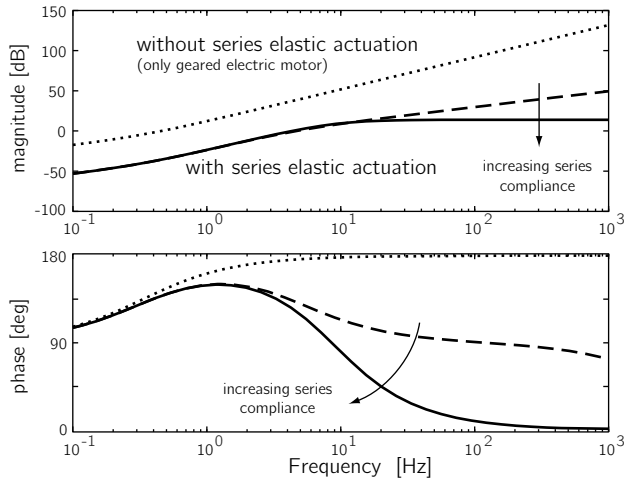


Figure 6.4: Modeled output impedance of an actuator with and without the presence of a series elastic element plus active torque controller controlling zero torque. The geared electric motor dynamics are modeled by the motor input torque/current together with a reflected inertia and damping. The torque controller is a proportional-derivative controller. System parameters and controller parameters are matched with the real robot parameters for the hip pitch joint. Input signal is a joint oscillation at 1 Hz with an amplitude of 1 rad. Impedance of the series elastic actuator (dashed and solid line) is approximately 35 dB (about a factor of 50) lower at all frequencies compared to the geared motor (dotted line). Using a more compliant element (instead of a typically stiff torque sensor) results in a lower actuator impedance at high (uncontrolled) frequencies and a decrease in phase lead (i.e., less ‘damping’ added by the actuator). Figure from Hobbelen and De Boer (2008).

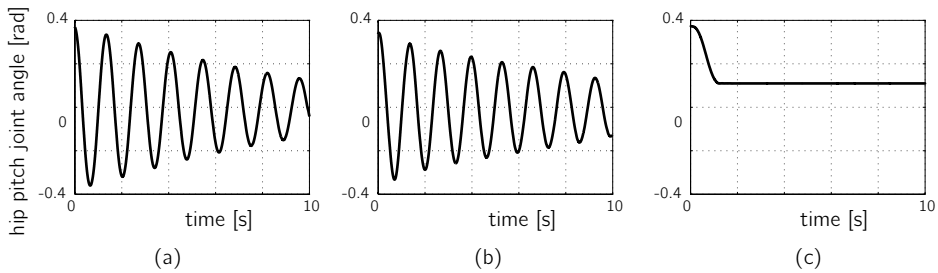


Figure 6.5: Measured hip pitch joint motion during swinging of a leg. (a) passive leg swing with actuators disconnected from joint (b) leg swing with a geared electric motor connected to the joint with series elastic element plus active controller that actively controls zero torque. (c) leg ‘swing’ with a geared motor directly connected to the joint; severely influencing the unactuated swing leg motion. Figure from Hobbelen and De Boer (2008).

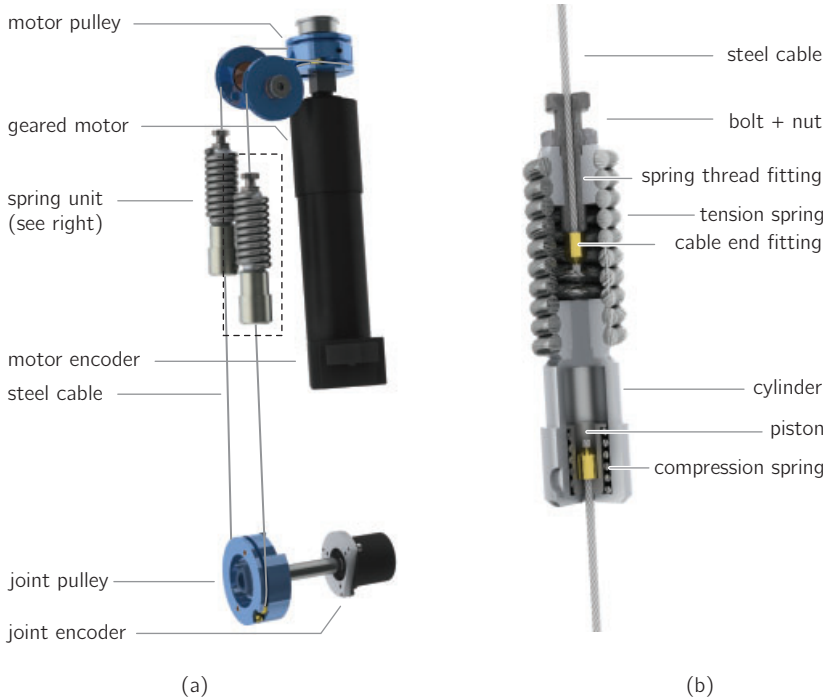


Figure 6.6: Implementation of series elastic actuation in the knee joint of TULip. (a) By measuring the difference in orientation of the encoder on the motor and joint side, the extension of the spring and steel cables can be determined, which is a measure for the torque exerted on the joint. (b) A cut-away view of the used 'spring unit', which consists of a high-stiffness tension spring in series with a low-stiffness compression spring with a mechanically constrained maximum compression (which serves as a tension spring with a constrained maximum extension but is easier to implement). Proper operation of the series elastic actuator requires the application of a preloading force (through the use of the bolt) that will just drive the piston down in the cylinder and fully compresses the compression spring (as explained in 6.7).

The compliant element that is used is a compliant unit, referred to as 'spring unit', that comprises a tension spring and a compression spring. Figure 6.6 shows an example implementation of a series elastic actuator in TULip. The reason why these spring units are used is as follows. In the initial design of TULip, normal tension springs were used as series elastic elements. To prevent cables from going slack and potentially come off the pulley when the motor applies torque on the system, a preload is required on the system. This preload is determined by the maximally allowed joint torque, as explained in Figure 6.7. This maximally allowed joint torque

has over the years been increased beyond the original design specifications to fulfill the need for higher joint torques during certain motions (e.g., from 8 to 15 Nm for the knee joint). The result was that the preloading force increased (from 450 to 900 N) and caused friction and wear of the cables. This hampers actuator performance, as is demonstrated by the frequency response measurement of the initial series elastic actuator Figure 6.8. The coherence measurement shows a generally low coherence which suggests non-linear system behavior due to the presence of friction. Application of the spring unit can solve this problem by reducing the required preload and thereby reduce the friction. The required preload of the spring unit was selected to be 75 N to overcome the internal prestress in the spring coil of the large tension spring. The frequency response measurement of Figure 6.9 shows a significant improvement of the coherence measurement.

Actuator performance

Figure 6.9 shows that the torque controllable bandwidth of the series elastic actuator in the knee joint lies between 10-15 Hz for torques up to 5 Nm and approximately 5-10 Hz for torques between 5-12 Nm. The bandwidth decreases for higher torque amplitudes since motor saturation occurs for high-speed and large spring deflections. This demonstrates a disadvantage of the series elasticity: the low stiffness of the element between the joint and the motor forms a limitation on the torque and position control bandwidth that can be obtained on the joints. However, this is not necessarily a problem since gait stability does not necessarily require high-speed torque or position control (which is also demonstrated by the walking robot Flame that has a comparable control bandwidth (Hobbelen and De Boer, 2008)).

The resolution of the force measurement depends on the resolution of the encoders, the transmission reduction and the stiffness of the deflecting elements. In the case of the knee joint, encoders are used on the motor and joint side that have 2000 and 30000 counts per revolution respectively. The reduction of the planetary gearhead is 111:1 and the reduction of the pulley system is 1.4:1. The stiffness of the large tension spring and small compression spring are 81 kN/m and 7.9 kN/m respectively. With the current selection of springs, the elasticity of the steel cables is not negligible. The stiffness of the steel cable is 64.5 kN/m per meter cable with about 0.3 m of cable used on each side of the pulley. This results in an equivalent rotational stiffness around the joint of about 25 Nm/rad. The resulting smallest resolvable torque is $5 \cdot 10^{-3}$ Nm with a maximum output torque of 15 Nm, which gives a dynamical range of maximally 3000:1.

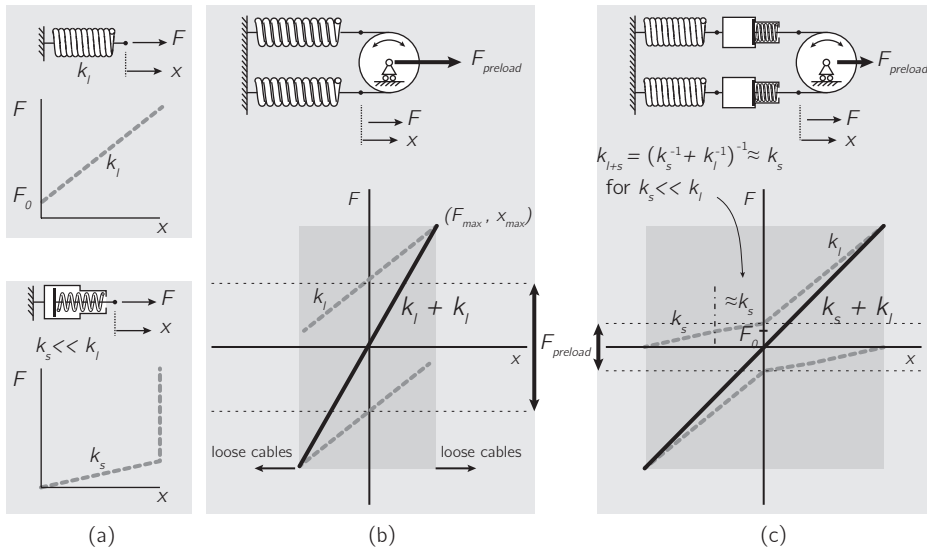


Figure 6.7: Series elastic actuation (SEA) with normal tension springs and with ‘spring units’ (as shown in Figure 6.6b). (a) Shows the relation between force F and displacement x for the large tension spring with stiffness k_I and small compression spring with stiffness k_s . The compression of the compression spring is limited mechanically. (b) Schematic representation of SEA configuration with tension springs. Application of force F extends one of the two springs. The configuration requires a preload to prevent the cable on the other side to go slack and come off the pulley. The required preloading force $F_{preload}$ is equal to the maximal force that the springs should cope with during operation. The resulting stiffness of the spring system is then $2k_I$. (c) Schematic representation of SEA configuration with the spring unit. A small compression springs is selected of which the stiffness is a factor of ten smaller than the tension spring and the force at maximum compression exceeds the internal prestress in the spring coil F_0 . The required force $F_{preload}$ is significantly reduced compared to the configuration in b. The preload compresses both compression springs. Applying a force F on the SEA system extends one tension spring and a compression spring on the opposite side. Though the stiffness profile of an individual spring unit is non-linear, the stiffness of the SEA system is linear and equal to $k_s + k_I$.

6.3.2 Actuator implementation in robot design

The legs are the most complex parts of the robot. To minimize their complexity and mass as much as possible, it is designed with a minimal number of degrees of freedom that would still allow for human-like versatile behavior. An overview of the degrees of freedom of the robot is given in Figure 6.10. Not all joints implement series elasticity. Again, this is out of concern for the robot’s mass and complexity. In comparison to regular a position-controlled DC motor, a joint with series elastic actuation requires substantially more building space since it requires at least one additional encoder

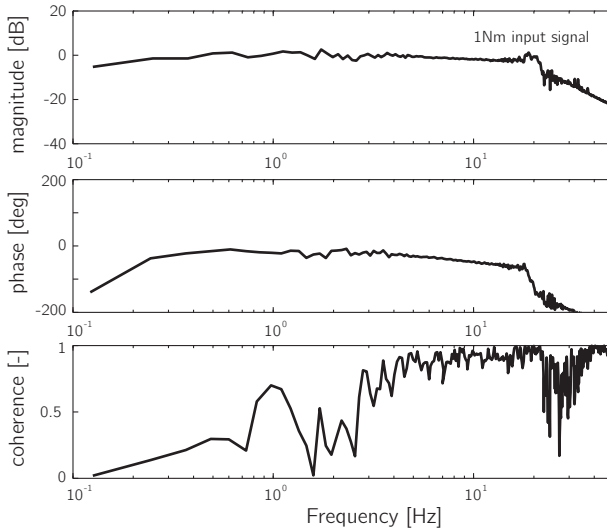


Figure 6.8: Frequency response measurement of the series elastic actuator of the knee with conventional linear springs as compliant element. The input is the requested joint torque, which is a 1 Nm sine sweep signal superimposed on a sine signal (0.5 Hz, 0.1 Nm) to overcome static friction. The output signal is the measured joint torque by the deflection of the spring.

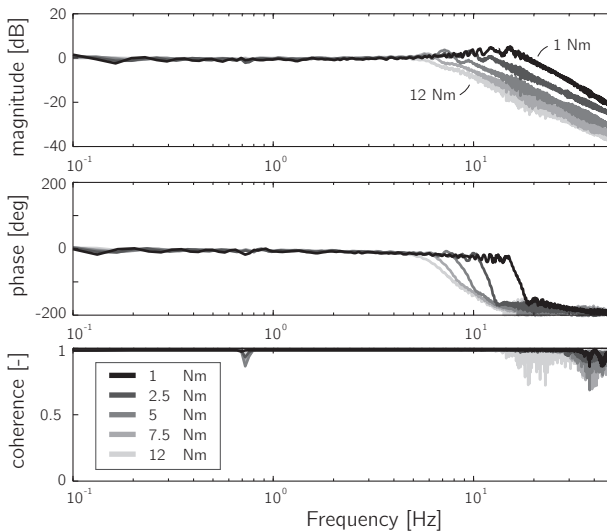


Figure 6.9: Frequency response measurements of the series elastic actuator of the knee with 'spring units' (see Figure 6.6b) as compliant element. Measurement protocol is equal to Figure 6.8. Five measurements are conducted for different sine sweep amplitudes between 1 and 12 Nm.

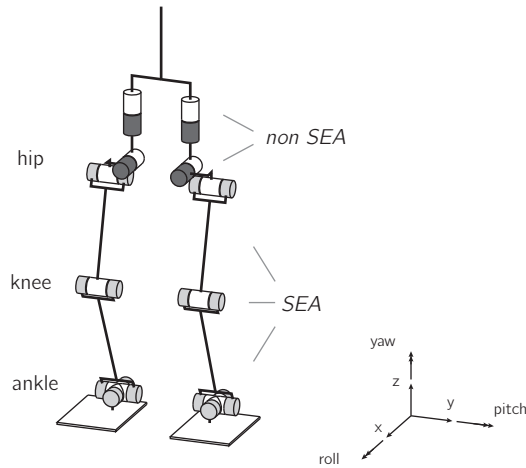


Figure 6.10: Degrees of freedom of TULip. TULip has a total of 12 degrees of freedom in the legs and two degrees of freedom (pitch and yaw) in the head which are not shown.

and additional springs. Therefore, series elastic actuation was only applied in those joints that were considered to have most benefit from its implementation: the joints that have a highly dynamic (fast) motion during walking and those that are in direct contact with the ground. This comes down to all but the hip yaw joint and hip roll joint. In the initial realization of TULip, an even more simple design approach was chosen: the ankle roll joint was left unactuated and two tension springs were used to create a mechanically adjustable ankle roll stiffness. Though this implementation was found practical in periodic straight-line walking (Hobbelen and De Boer, 2008; Wisse et al., 2007, 2006), it was quickly found that controllability of the ankle roll torque was desired to achieve more versatile behavior.

Figure 6.11 shows the current implementation of all the actuators in the robotic leg. For all series elastic actuated joints, we have placed the actuators high up in the leg and body to minimize the mass at the end of the legs. The actuator for the ankle pitch joint was placed in the upper body and is connected to the joint through a Bowden cable drive. This enables active plantarflexion of the ankle and passive dorsiflexion through a return spring. This positioning of the actuators effectively reduces the mass moment of inertia of the legs with respect to the hip joint, which enables faster swing-leg motion. A naturally fast swing-leg motion is desirable for rapid foot placement strategies to prevent a fall and can also decrease energy use (Kuo, 2002). The resulting mass moment of inertia of both the legs with respect to

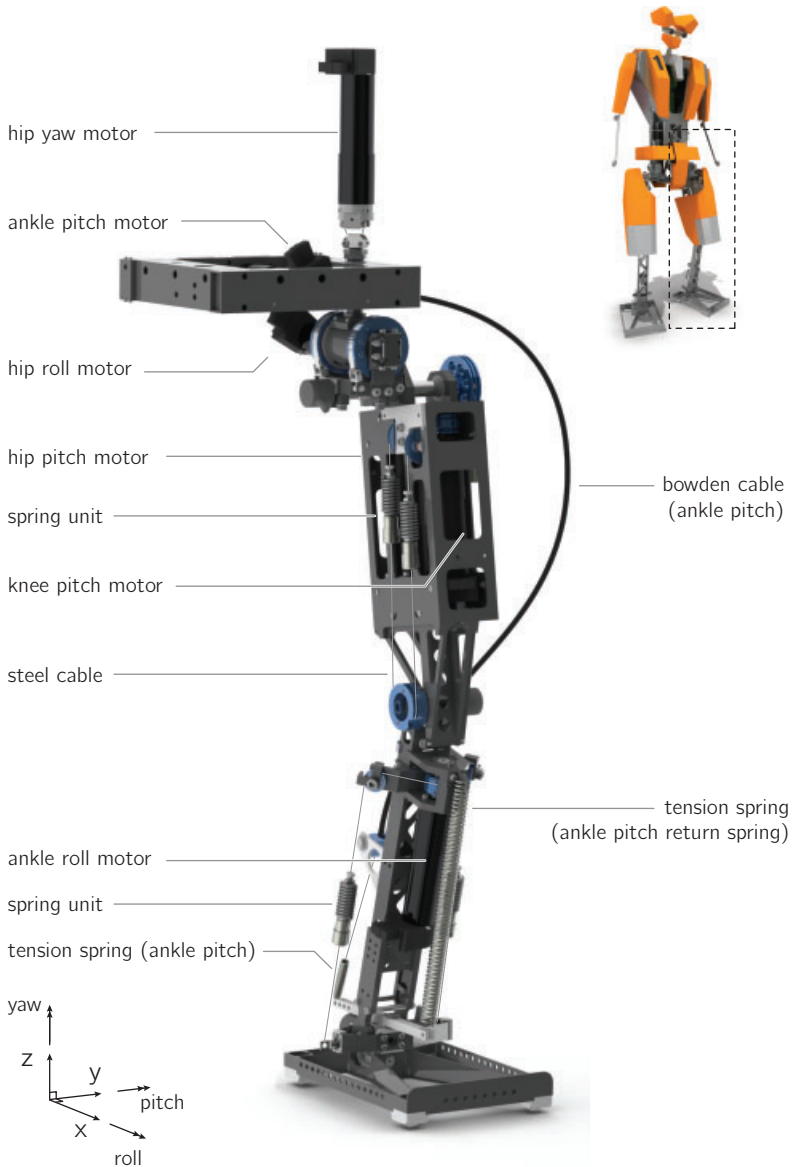


Figure 6.11: Design of the hip and left leg of TUlip (year 2011) without protective covers.

the hip is about 60% of the mass moment of inertia of the upper body with respect to the hip. This can be considered low when comparing this to the human case, for which this is 75% (Winter, 1990).

Kinematic data of the leg is shown in Figure 6.12. The leg was placed slightly to the side of the pelvis/trunk of the robot to allow for a large range of motion of the hip pitch and hip roll joints¹. This large range of motion was required to be able to demonstrate the necessary skills for the RoboCup competition: standing up from a lying position, kicking a ball, and performing a split for goalkeeping.

The design of the upper body is rather straightforward as can be seen in Figure 6.1. The structure of the trunk is stiff, lightweight and simple and houses all the electronics. The arms (which currently only have esthetic value) and head are actuated through simple Dynamixel servomotors. The total robot's mass and joint layout are shown in Table 6.1 on page 121.

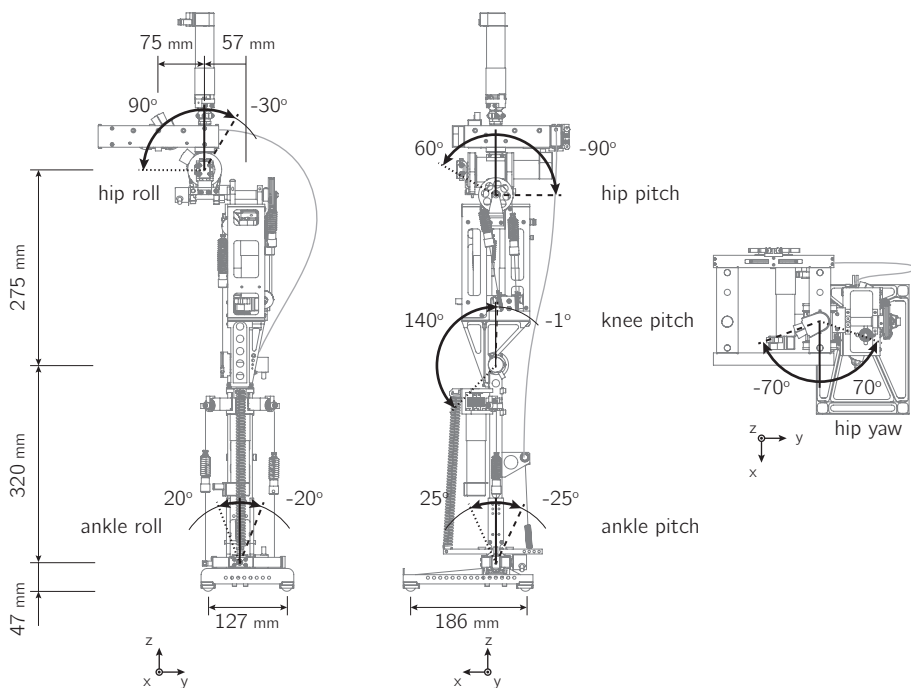


Figure 6.12: Joint range of motion and main dimensions of the hip and left leg of TULip (year 2011). The total length of the robot is approximately 130 cm.

¹The range of motion of the hip pitch joint has been reduced in 2011 from $[-140, 90]$ to $[-90, 60]$ (as shown in Figure 6.12) to accommodate for longer springs in the series elastic actuator.

6.4 Electronics

For the electronic architecture, the initial design requirements were as follows:

- preferably the use of proven, off-the-shelf technology to prevent inevitable bugs in custom made components;
- preferably the use of components with low mass and small volume;
- separate power circuitry for low power electronics and high power motors to prevent interaction.

An overview of the resulting design is given below.

6.4.1 Computing

All the sensors and actuators are connected to a single on-board computer that does all the computing. Distributed control may have been favorable from a weight point of view since locally placed controller boards can reduce the number of power cables and sensor cables significantly. However, at the time available distributed controller boards were not yet small and powerful enough for these robotic applications. The selected central computer is a Diamond Systems 1GHz, 512MB SDRAM Poseidon single board computer (Diamond Systems, 2011) with EPIC form factor. To expand the I/O capabilities of the Poseidon board, two Mesa 4165 Anything-I/O cards (MESA, 2011) are stacked onto its PC104-Plus bus. The FPGA's on these cards (Xilinx Spartan-II 200k gate) are programmed to count encoder pulses and communicate with the Elmo Whistle digital servo amplifiers (Elmo Motion Control, 2011).

6.4.2 Sensors

Rotary encoders are mounted on each motor shaft. These Maxon encoders of type HEDS 5540 have 500 pulses per revolution. All actuators with series elasticity have a second rotary encoder placed on the joint axis, a Scancon 2RMHF (Scancon, 2011) incremental encoder with 7500 pulses per revolution corresponding to a resolution of $2 \cdot 10^{-4}$ rad (approximately 0.01 deg). Though incremental encoders require initialization, these encoders are preferred over absolute encoders because of their high resolution. Lower resolution encoders were found to output non-smooth velocity signals at low angular joint velocities, which significantly limits the amount of damping that can be added to the system through velocity based control.

The upper body of the robot is equipped with an Xsens Mti (Xsens, 2011) inertial measurement unit. It comprises three accelerometers, three gyroscopes and three magnetometers and outputs an estimate on the orientation and rotational velocity of the unit in three dimensions.

Each foot is equipped with four Tekscan Flexiforce (Tekscan, 2011) sensors to detect contact with the floor. These pressure sensors are connected to a custom designed ARM7 board, which is used to process the signals and interface with the main computer through USB.

Two wide angle cameras are mounted on the head, which are part of a Surveyor Stereo Vision System (Surveyor, 2011). Image processing is done on the on-board Blackfin processors. The stereo vision module is connected with the Poseidon computer through LAN.

6.4.3 Power supply

The sensors and computer are powered separately from the motors to prevent interaction. The sensors and computer are powered by a Kokam (Kokam, 2011) battery of three 3.7V 3.3Ah LiPo cells, the motors by a Kokam battery of eight 3.7V 2.0Ah LiPo cells. The motors and amplifiers are protected from overvoltage (induced by back-EMF) through the application of a shunt circuitry. When the shunt circuitry detects a voltage above 50V, a low resistance ($12\ \Omega$) shunt resistor is added to the load of the bus which dissipates all excess energy (with a dissipative power of 100W).

6.5 Software

For the software architecture, the main design requirements were as follows:

- real-time I/O loop, preferably at 1 kHz;
- preferably the use of available, open source middleware;
- development of a robot simulation model that shares the same controller as the real robot.

The resulting design is as follows. A Xenomai Linux build on Debian Lenny was selected as a real-time operating system. The architecture of the control software was based on the RoboFrame architecture which was specifically designed for software integration and communication for complex robots in complex environments such as RoboCup (Friedmann et al., 2006; Petters et al., 2007). This C++ based framework

provides a number of services to robot control applications, such as module management, timers, intra and inter-process communication and monitoring and debugging facilities. Various modules were programmed within the RoboFrame application (RoboApp, see Figure 6.13), such as modules related to motion control, vision, world modeling, communication and soccer-strategy.

Within RoboFrame's RoboGUI, a graphical user interface was created that can be used to monitor and log robot data or modify control parameters. Multiple instances of this user interface can run on other Windows or Linux platforms and communicate with the RoboApp. Since RoboFrame has no native support for real-time platforms, all real-time control was done in a separate process outside RoboFrame which interfaces with an adapter module inside RoboFrame. This RoboFrame module accepts commands from other modules (e.g., soccer-strategy) and relays these to the real-time process. Data from the real-time process can be sent back through the adapter to be used by any of the other RoboFrame modules.

The real-time control algorithm consists of different layers. The bottom layer consists of hardware drivers that interface with the I/O ports on the Poseidon and Mesa cards. The TUlip Hardware Abstraction Layer (HAL) provides abstracted access to the robot hardware. All platform dependencies, such as device registers, memory locations or bus accesses were stored in the TUlipHAL layer. The software layer above TUlipHAL, TUlip Motion Control (MC), communicates with the hardware objects only through abstract hardware objects and is, consequently, more immune to hardware changes. As a result, controlling the real robot or controlling the simulated robot is defined by the interface with either real objects or simulated objects through the abstract objects of TUlipHAL. This is similar to the approach adopted by Kanehiro et al. (2004) and Pratt et al. (2009).

TUlipMC implements all the control algorithms. It consists of a hierarchical statemachine that switches between behavioral states (e.g., walking, kicking a ball, balancing) and their behavioral substates (for example for walking: "single stance, left is swing leg"). The statemachine is responsible for the timely update of robot inputs and outputs, and handles event based transits between the different behavioral (sub)states. A detailed description of the various states that were implemented in the statemachine is given in Chapter 7. The statemachine is set to run at 1 kHz, for which the load on the computer's processor is about 35%.

The software architecture is designed in such a way that the control algorithms within TUlipMC can also be directly used to control a model of the robot. When working in simulation, TUlipMC gets input data from simulated sensors and outputs to simulated motors. This architecture is chosen to speed up the development and evaluation of

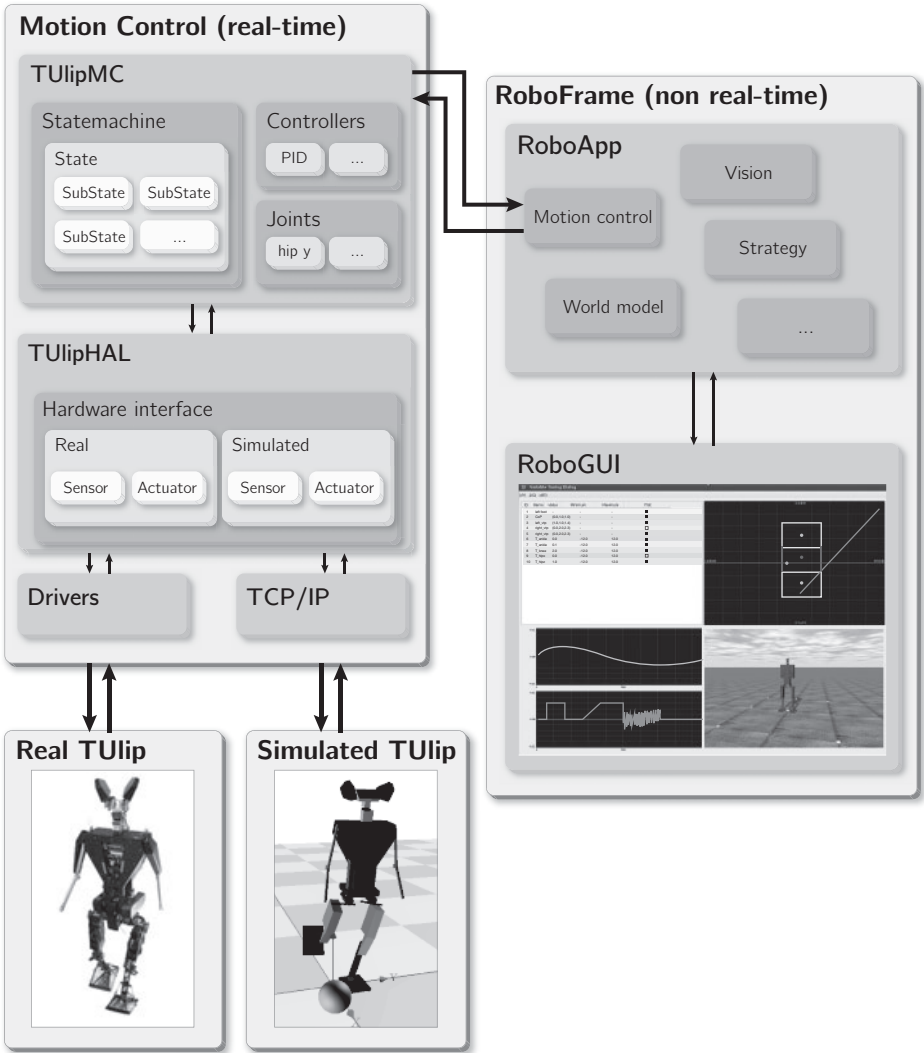


Figure 6.13: Schematic overview of the software architecture used to operate TULIP.

new control concepts and behaviors without having to deal with hardware issues. The model is created using the 20sim software package (Controllab Products B.V., 2011; Kemp, 2010). It consists of a full multi-body model of the robot with the same mechanical properties as the real robot. Physical contact with the environment can be simulated for the feet, the 'hands', and the 'elbows' through the application of either Kelvin Voight, or Hunt-Crossley models. Motor dynamics and friction are currently not included in the simulation to limit the model's complexity.

6.6 Conclusion

We could say that the design of TULip is satisfactory as most of the design requirements were met. The desirable low mass moment of inertia of the legs was realized, the electronics were found to be reliable, and the software was found to function properly. Furthermore, the overall weight of the robot was indeed relatively low, less than 20 kg for a 1.3 m tall robot (see Figure 6.14).

However, many of the design requirements were based on estimates and experience rather than trustworthy research, as the purpose of the robot was to validate these estimates. The resulting performance of the robot will be evaluated in Chapter 8.

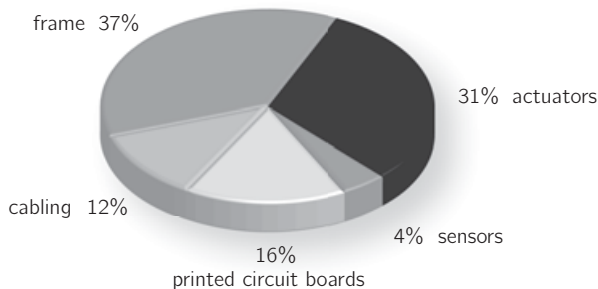


Figure 6.14: Weight allocation of the robot TULip, with a total mass of 19.8 kg.

Acknowledgements

The development of the TULip prototypes could not have been done without the help of the many students and staff that have contributed through the DutchRobotics foundation (DutchRobotics, 2011).

Table 6.1: TULip's mechanical system properties (year 2011). Each row represents a joint and its associated rigid body. In the coordinate system, x is forward, y is to the left, and z is up. The right leg is a mirror image of the left leg (L). All motors are 24V 60W RE30 Maxon motors; the transmission reduction (gearhead plus pulley ratio) is given for each joint.

Joint	Rigid body	Parent	Offset from Parent [m]			Reduction	Mass [kg]	CoM offset [m]			
			x	y	z			x	y	z	
root	trunk	N/A	-	-	-	-	10.8	0.06	0.0	0.0	-0.020
L hip yaw	L hip X motor	root	0.085	0.075	-0.392	1:86	0.61	-0.009	0.0	0.0	0.007
L hip roll	L hip Y axle	L hip yaw	0.0	0.0	-0.071	1:190	0.08	0.0	0.0	0.0	0.0
L hip pitch	L thigh	L hip roll	0.0	0.038	-0.04	1:222	2.14	0.005	0.030	0.030	-0.097
L knee	L shank	L hip pitch	0.0	0.0	-0.275	1:155	1.02	0.038	0.019	0.019	-0.150
L ankle pitch	L ankle	L knee	0.0	0.0	-0.32	1:325	0.08	0.0	0.0	0.0	0.0
L ankle roll	L foot	L ankle pitch	0.0	0.0	0.0	1:286	0.37	0.035	0.010	0.010	0.010

7

Robot prototype: capturability based control

Integral copy of:

J. Pratt, T. Koolen, T. de Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, P. Neuhaus

Submitted as "Capturability-Based Analysis and Control of Legged Locomotion: Part 2, Application to M2V2, a Lower Body Humanoid" to *International Journal of Robotic research*, 2011

Note:

This chapter is realized in strong collaboration with the Institute for Human and Machine Cognition in Florida, USA. It describes a robotic control algorithm based on the concept of capturability. Its application is presented for IHMC's force-controllable robot M2V2 but also applies to TULip. There are some notable differences. The swing leg controller in TULip is based on inverse kinematics and does not use inverse dynamics. Also, walking with the presented algorithm has not (yet) been successfully demonstrated on TULip. Chapter 8 presents the results obtained with TULip.

Abstract

This two-part paper (i.e., Chapter 3 and 7) discusses the analysis and control of legged locomotion in terms of N -step capturability: the ability of a legged system to come to a stop without falling by taking N or fewer steps. We consider this ability to be crucial to legged locomotion and a useful, yet not overly restrictive criterion for stability.

Part 1 (i.e., Chapter 3) introduced the N -step capturability framework and showed how to obtain capture regions and control sequences for simplified gait models. In Part 2, we describe an algorithm that uses these results as approximations to control a humanoid robot. The main contributions of this part are 1) step location adjustment using the 1-step capture region, 2) novel instantaneous capture point control strategies, and 3) an experimental evaluation of the 1-step capturability margin. The presented algorithm was tested using M2V2, a 3D force-controlled humanoid robot with 12 actuated degrees of freedom in the legs, both in simulation and in physical experiments. The physical robot was able to recover from forward and sideways pushes of up to 21 Ns while balancing on one leg and stepping to regain balance. The simulated robot was able to recover from sideways pushes of up to 15 Ns while walking, and walked across randomly placed stepping stones.

7.1 Introduction

Making humanoid robots useful in complex environments requires attaining good disturbance rejection properties while performing other tasks, such as walking. Current robots have not sufficiently demonstrated this ability. In Part 1 (i.e., Chapter 3), we proposed to approach this problem using the concept of N -step capturability, informally defined as the ability to come to a stop in N steps or fewer. We provided a capturability analysis of three simplified gait models. This chapter presents capturability-based control algorithms for balancing and walking that are robust to pushes and unexpected ground variations. In the presented control algorithms, we make use of the concepts derived in the previous part of this two-part paper. These control algorithms were implemented on M2V2, a force controlled 3D lower body humanoid with two 6 DoF legs.

The three main areas of contribution of this part are:

1. step location adjustment using an approximated 1-step capture region, based on the 3D Linear Inverted Pendulum Model (3D-LIPM) with finite-sized foot;

2. novel instantaneous capture point control strategies using placement of the center of pressure (CoP);
3. an experimental evaluation of the control algorithms using the *N-step capturability margin* introduced in Part 1 (i.e., Chapter 3).

Furthermore, the current part extends some of the control strategies from previous work (Pratt et al., 2001) to 3D.

To date, we have achieved push recovery during one-legged balancing on the physical robot. In simulation, we have also achieved push recovery while walking on level ground and while walking over stepping stones. These results demonstrate that capturability-based control using simplified dynamic models may be useful in developing bipedal walking control algorithms that are robust to disturbances. While performing one-legged balancing, the physical M2V2 was able to recover from forward and sideways pushes of up to 21 Ns. The simulated version of the robot was able to recover from sideways pushes of up to 15 Ns while walking on level ground.

The remainder of this paper is structured as follows. Section 7.2 presents related literature. In Section 7.3 we describe the M2V2 robot. Section 7.4 describes the simulation environment for the robot. In Section 7.5 we describe the control tasks that we are interested in achieving with the robot. Section 7.6 describes some of the control concepts that we employed in developing control algorithms. Section 7.7 presents implementation details of our control algorithms. Section 7.8 presents results. In Section 7.9, we discuss capturability-based analysis and control and suggest future work. Finally, we conclude the paper in Section 7.10.

7.2 Background

The literature on control algorithms for humanoid robots is extensive. Here we provide a brief survey of some widely used control techniques and focus on their disturbance rejection properties.

ZMP-based trajectory tracking control

The Zero Moment Point (ZMP) is the point about which the resultant ground reaction torque has no horizontal component (Vukobratovic and Borovac, 2004). ZMP-based trajectory tracking control usually encompasses choosing a desired ZMP trajectory based on available footholds and desired gait properties, and calculating the center of mass (CoM) motion that results in that ZMP trajectory (Kagami et al., 2002; Kajita et al., 2003). It is widely used in legged robot control since maintaining

the ZMP strictly inside the support polygon at all times guarantees that it is physically possible to track the reference joint trajectories using conventional control tools. The distance from the ZMP to the edge of the support polygon can be used as a measure of robustness. Modifying the reference ZMP trajectory online has also been explored, including both small local ZMP changes and larger step placement changes (Nishiwaki and Kagami, 2010). Another ZMP control approach treats the ZMP as a control input, which is manipulated to produce a desired motion of the CoM. For example, central pattern generators have been used to calculate the reference ZMP trajectory to produce walking (Sugihara, 2010).

Typical ZMP-based gait generation techniques cannot be used to generate gaits for which the stance foot rolls from heel to toe, as observed in fast human walking. This limitation is due to the ZMP needing to be inside the support polygon to meet the ZMP stability criterion (Vukobratovic and Borovac, 2004). However, when the robot rotates about an edge of its support foot, the area of the support polygon decreases to too small of a value to provide robustness against small modeling errors. Also, the reference joint trajectories themselves might lead the robot to a fall by design, even if the ZMP is kept inside the support polygon at all times. Hence, the ZMP criterion is not a necessary condition to avoid falling (Pratt and Tedrake, 2006). In addition, the ZMP criterion is not applicable to non-flat terrain (Wieber, 2002). Most importantly for the present work, ZMP analysis does not answer the crucial question of where to step to recover from large disturbances.

Passive Dynamics Based Control

Another approach to walking control explicitly relies upon the passive behavior of the robot's mechanical components. Straight-line periodic walking has been demonstrated for purely passive devices walking down a slope (Collins et al., 2001; McGeer, 1990a). Adding limited actuation to machines designed for passive walking can yield a controlled, efficient gait (Collins et al., 2005). Rejection of small disturbances has been shown for planar walkers under limited control (Kuo, 1999; Tan et al., 2010), as well as locally stable gaits with purely reflexive control (Geng et al., 2006). A major focus of our current work is endowing robots with the ability to recover from disturbances large enough to require significant actuation, so we cannot rely on passive dynamics alone to avoid falling.

Hybrid Zero dynamics

Another approach to locomotion control identifies relationships between the degrees of freedom of a robot that lead to a steady gait (Westervelt et al., 2003). These relationships are then enforced by a feedback controller, yielding a locally stable,

periodic gait. This method has been shown to be able to cope with moderately rough terrain (Sreenath et al., 2010) (Westervelt et al., 2004). More recently, this method has been used to generate three dimensional walking (Chevallereau et al., 2009). However, it requires off-line computation of a repetitive gait, and therefore it currently has no mechanism for explicitly handling rough terrain with impassable areas. Also, it is still unclear how a robot using this method will handle large pushes that significantly disturb the state of the machine from the preplanned gait.

Compliant Strategies for Force Controllable Robots

Force controllable robots have led to the development of compliant control strategies. Virtual model control and other intuitive control strategies were used on the 2D walking robot Spring Flamingo (Pratt et al., 2001). Coros et al. (2010) combine Jacobian transpose control, joint space PD control, and gravity compensation with step planning based on an inverted pendulum model to obtain a walking controller that works for a range of simulated characters, while performing secondary tasks. Stephens and Atkeson (Stephens and Atkeson, 2010) introduced an algorithm that combines joint PD control, virtual model control and Dynamic Balance Force Control. Dynamic Balance Force Control is an inverse dynamics approach based on the contact forces obtained from a CoM planner. Hyon and Cheng (2007) achieved disturbance rejection using a passivity-based controller, later complemented by CoM control using a Dynamic Balancer (Hyon et al., 2009).

Compliant strategies can enhance the robustness of a walking algorithm since they focus on interaction forces with the environment to achieve higher level goals, instead of relying on high gain position control and extremely accurate ground models to achieve perfect kinematic trajectories. The presented work also uses a compliant control strategy.

7.3 Description of M2V2 Robot

M2V2 (see Figure 7.1) is a lower body humanoid with two 6 DoF legs (Pratt et al., 2009). There are three degrees of freedom at each hip, one at each knee, and two at each ankle. See the Appendix for approximate inertia parameters and joint offsets, which we use in the simulation model. It is a second generation redesign of M2, a robot developed at the MIT Leg Laboratory (Paluska, 2000; Pratt, 2000a).

Each degree of freedom is driven by an identical force controllable Series Elastic Actuator (SEA) (Pratt and Williamson, 1995; Pratt and Pratt, 1998). These actuators use a spring in series between the drive train and the load. By measuring the spring deflection, the force exerted by the actuator can be measured. Using a feedback



Figure 7.1: M2V2, a 3D force-controllable humanoid robot with twelve actuated degrees of freedom.

controller, the actuator force can be controlled accurately. For M2V2, each actuator can produce a force of up to 1.3 kN, with a smallest resolvable force of approximately 4.4 N, giving it a 300 : 1 dynamic range. The force controllable bandwidth of each actuator is approximately 40 Hz. M2V2 has two U.S. Digital EM1-0-500 linear encoders and LIN-500 encoder strips at each Series Elastic Actuator, one to measure position and one to measure spring deflection.

Onboard computation is provided by a PC104 with a dual core Pentium-M processor. Sensor reading is done by several AccesIO 104-Quad8 encoder input boards. Desired current is output as a PWM signal through two Real Time Devices 6816 PWM boards. Body orientation and angular rate are measured using a MicroStrain 3DM-GX3-25 inertial measurement unit. Current control is provided through twelve Copley Controls Accelnet module ACM-180-20 amplifiers. The PC104, I/O cards, and current amplifiers are all located in the body of the robot.

A custom designed push stick equipped with a digital Loadstar ILoad Pro load cell was constructed for measuring pushing forces applied to the robot. This load cell is connected to the robot to eliminate data synchronization issues, but the control algorithm does not have access to its output.

7.4 Simulation environment

We have developed a simulation model of M2V2 using the Yobotics Simulation Construction Set (Yobotics, Inc, 2011). This software package allows for rigid body

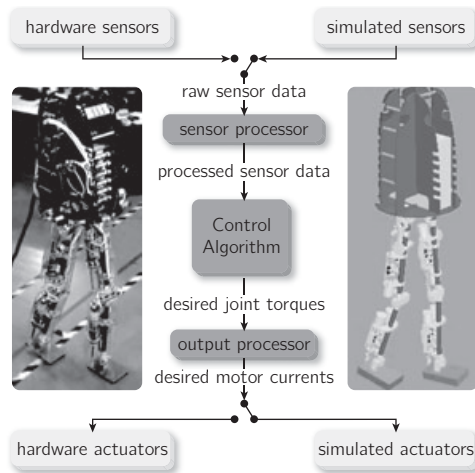


Figure 7.2: Overview of the software architecture. The majority of the software runs both in simulation and on the real robot, eliminating the need to maintain separate versions, and allowing for significant development and testing in simulation.

dynamics modeling and simulation using the Articulated Body Algorithm (Featherstone, 1987; Mirtich, 1996) and a fourth order Runge-Kutta integrator. The integration step size we use is 0.1 ms. Ground contact forces are determined using spring-damper ground models, sometimes referred to as penalty-based methods.

Pushing disturbances are modeled as high intensity forces of constant magnitude that are applied for a short duration. The force is applied to the midpoint between the hip joints. In future work we intend to investigate recovery from pushes at any location on the robot. Stepping stones are modeled as polygons.

Our software architecture (see Figure 7.2) is designed to have a common control algorithm that is used both in simulation and on the physical robot. The only differences between the simulation and the physical robot are the source of the raw sensor data and the destination of the desired motor currents. The simulated sensors include noise and discretization error. The actuators are modeled as low pass filters to simulate the bandwidth of the SEAs and have maximum output force limits. The control algorithm threads run at the same rate in simulation and on the embedded computer. By having a shared code base, we eliminate the effort and bugs that are typical when porting from a simulation environment to real hardware.

7.5 Control tasks

The controller was designed for two separate control tasks: 1) balancing on one leg and 2) walking. These tasks require slightly different implementations of some of the modules in the controller, although the main control algorithm is the same.

7.5.1 Balancing

Balancing on one leg (subsequently called 'balancing') entails going from a double support configuration to single support, and remaining in single support for as long as possible. If the robot is significantly perturbed while in single support, it will need to take a step to prevent a fall. Possible disturbances include pushes of significant magnitude and, to a lesser extent, sensor and actuator imperfections. Pushes of a magnitude requiring the robot to take a step will be assumed to have a direction that does not require a cross-over step. For example, if the robot is balancing on its left leg, then significant pushes could be directed forward, backward or to the right, but not to the left. Balancing was chosen as a precursor to walking because it is challenging and requires good foot placement, but it is not as difficult as walking since sustained forward progression is not required. To date we have achieved balancing on the physical robot, recovering from forward and sideways pushes up to 21 Ns.

7.5.2 Walking

The walking task requires the robot to move forward and be able to change walking direction. In addition, the controller needs to be able to handle walking over stepping stones. To date, walking on flat ground without pushes has been achieved on the physical robot. Walking in the presence of pushes up to 15 Ns and walking over stepping stones have been achieved in simulation.

7.6 Control concepts

This section describes three key concepts we use in our control strategies, regardless of the control task. These concepts are 1) capturability-based control using an approximate model, 2) force control and 3) virtual model control.

7.6.1 Capturability-based control using an approximate model

We consider bipedal locomotion to be an inherently robust control problem, which does not require great accuracy in controller design. This motivates our use of an approximate model. Out of the three simplified gait models described in Part 1 (i.e., Chapter 3), we have chosen to use the 3D-LIPM with finite-sized foot for approximations. See Section 7.9.1 for the motivation for this choice. We base our

control strategy on the instantaneous capture point and the approximated 1-step capture region. Although no guarantees on capturability can be made for the robot using this approximation, we have found that it works well in practice.

Considering the 3D-LIPM with finite-sized foot, the instantaneous capture point is the point on the ground where the CoP should be placed instantaneously and maintained to come to a rest with the CoM directly above the CoP. We do not specify a desired CoM trajectory; instead, desired instantaneous capture point paths are used as a basis for control. The linear dynamics of the instantaneous capture point allows us to find a desired CoP location within the base of support that ‘pushes’ the instantaneous capture point along the desired path.

For the balancing task, we determine whether taking a step is necessary based on whether the instantaneous capture point has left the base of support (see Corollary 2 in Chapter 3 and (Pratt et al., 2006)). For both control tasks described in this part, the controller will attempt to step to a desired step location in the 1-step capture region. If the robot is disturbed significantly in mid-swing, the desired step location will be adjusted so that it always lies in the 1-step capture region. Section 7.9.2 provides a discussion on why we chose to base the controller on the 1-step capture region, as opposed to using an N -step capture region with $N > 1$.

7.6.2 Force control

Force controllable actuators allow a biped to walk smoothly and naturally. These actuators allow for compliant control methods that are forgiving to external forces and unknown terrain. Traditionally, many humanoid robots use high-gain position control to track prescribed joint trajectories using non-backdrivable actuators. This approach typically requires near perfect knowledge of the terrain, a near perfect dynamic model of the robot, and no external forces. When pushed or encountering unexpected terrain, these robots may no longer be able to follow the prescribed joint trajectories, and either a new trajectory must be computed on the fly, or the robot falls.¹

Our control approach avoids the use of joint trajectory tracking for the stance leg. Instead, we use low-impedance feedback controllers that control the fundamental aspects of walking (foot placement and body height, orientation and speed), rather than attempting to rigidly control each degree of freedom. Force control provides some robustness to rough terrain since the exact foot/ground contact configuration

¹Trajectory tracking bipedal robots often use compliant foot pads, force sensors, and real time modification of ankle trajectories to control the CoP location. This technique essentially converts the ankle actuators to lower impedance force controllable actuators.

is less important than the interaction forces between the feet and the ground. In addition, force controllable actuators simplify control of the CoP location, and allow compliant control techniques, such as virtual model control.

7.6.3 Virtual model control

Virtual model control is a tool that allows a designer to control a robot by choosing virtual components, such as springs and dampers, to intuitively achieve task goals (Chen et al., 2001; Hu et al., 1998, 1999; Pratt et al., 2001). Once these components are chosen, the kinematic model of the system and additional user defined constraints allow direct calculation of the joint actuation required to simulate the effect of the desired components.

For example, in previous work on Spring Turkey, a planar walking robot with 4 actuated degrees of freedom, we decomposed the requirements of walking and designed simple virtual components to achieve each one (Pratt et al., 2001). CoM height was maintained by a virtual vertical spring-damper “granny walker” and forward travel was achieved with a virtual “track bunny” with constant forward velocity connected to the robot’s body by a virtual horizontal damper. The joint actuation was calculated using the transpose of a Jacobian that spans the joints between two virtually controlled points. While kinematic singularities remove degrees of freedom from the possible actuation, the joint torques for the remaining degrees of freedom can still be calculated. Similar techniques were used in the control of Spring Flamingo, a planar walker with 6 actuated degrees of freedom. Virtual model control is used in the control of M2V2 to maintain CoM height and body orientation, and to achieve approximate CoP control.

7.7 Controller implementation

We now present a detailed description of the balancing and walking controller. The controller’s input is comprised of joint angles and angular velocities, and the orientation and orientation rates of the upper body. The controller’s output is comprised of desired torques at each joint. The controller consists of five main parts:

1. the state machine, which keeps track of the gait phase for each leg and acts as a supervisory system that calls the appropriate lower level routines (see Section 7.7.1),
2. the capture region calculator, which determines the instantaneous capture point and the 1-step capture region (see Section 7.7.2),

3. the desired footstep calculator, which determines where to step to next (see Section 7.7.3),
4. the swing sub-controller, which computes the torques for the swing leg joints, if any (see Section 7.7.4), and
5. the stance sub-controller, which computes the torques for the stance leg joints (see Section 7.7.5).

See Figure 7.3 for an overview of the control architecture. The same state machine structure is used for both control tasks (balancing and walking), but with different control actions and transition conditions for each task. While the swing and stance sub-controllers are different for each task, they share many of the underlying control modules. The desired footstep calculator is also task-specific.

The capture region calculator is task independent and is the module most linked with capturability-based analysis and control. This module may also be useful in other legged robot control architectures and with other walking control techniques.

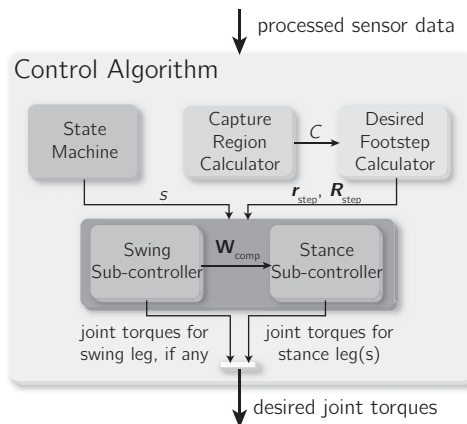


Figure 7.3: Overview of the control architecture. Arrows represent data flow. Arrows that point to the edge of a block signify that the corresponding information is available to all subblocks. The state machine produces the controller state s . The capture region calculator approximates the 1-step capture region C . The desired footstep calculator uses the capture region to compute a step location r_{step} and orientation R_{step} . The swing sub-controller computes torques for the swing leg if the robot is in a single support state. It also produces a wrench W_{comp} that the stance sub-controller uses to compensate the swing motion. The stance sub-controller computes the torques for both legs in double support, or just the stance leg in single support.

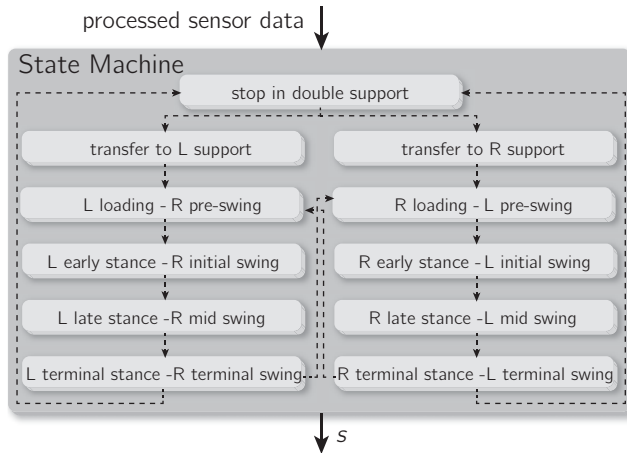


Figure 7.4: The state machine, which produces the controller state s . Blocks represent states and dashed lines represent available state transitions. When the control algorithm is started, the robot is in the 'stop in double support' state.

7.7.1 State machine

The state machine structure, shown in Figure 7.4, is based on the gait phases that a *single leg* goes through during human walking, as described in the biomechanics literature (Perry, 1992). See Figure 7.5 for a graphical depiction of these gait phases. The gait phases can be grouped into stance phases and swing phases. Below we provide a short description of each gait phase. See (Perry, 1992) for more detailed descriptions.

Stance phases

During *loading response*, the shock of initial ground contact is absorbed by bending the knee and using the heel as a rocker. The leg is loaded, while the trunk is kept upright. Once the opposite leg is lifted, the robot transitions into *mid stance*, in which the CoM moves forward over the stance foot as the leg is straightened. The robot transfers into *terminal stance* when the CoM is in front of the support foot. During this state, the heel rises and the knee is first straightened further and then begins to flex slightly.

Swing phases

When the opposite leg makes contact with the ground, the robot transitions into *pre-swing*. The leg is unloaded and bent more in preparation of the swing phase. *Initial swing* begins as the foot lifts off the floor. Foot clearance is achieved and

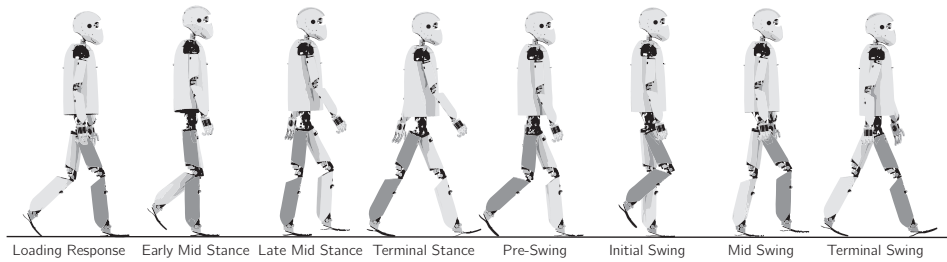


Figure 7.5: The gait phases upon which the state machine is based. The labels correspond to the phase of the highlighted right leg. Adapted from (Perry, 1992), which describes the phases of gait for human walking.

the leg is swung forward. When the swinging limb is opposite the stance limb, the robot enters *mid swing*. The hip is flexed further and the knee is allowed to extend in response to gravity. Finally, the robot transitions into *terminal swing*, in which the knee is extended as limb advancement is completed.

Transitions between gait phases for the right leg are directly coupled to those for the left leg. The states shown in Figure 7.4 were hence created by combining one gait phase for the left leg and one gait phase for the right leg, e.g. ‘left early stance - right initial swing’. In addition to the eight walking states, there is also a state in which the robot is stopped in double support and states in which weight is transferred to one leg, allowing the robot to start from a stop.

While this state machine is based on walking, it is easily adapted to one-legged balance, through appropriate selection of control actions in each state and transition conditions between states.

7.7.2 Capture region calculator

The capture region calculator approximates the 1-step capture region C , the set of locations where the *contact reference point*, a chosen point on each foot, may be placed at the next step so that the robot is able to reach a captured state. The algorithm is based on the capturability analysis of the 3D-LIPM with finite-sized foot, as presented in Part 1 (i.e. Chapter 3), and we refer to that part for the underlying theory; here we only present the implementation details of the capture region calculator. See Algorithm 1 for pseudocode. Figure 7.6 describes the algorithm graphically.

The algorithm can handle limitations on where the CoP may be placed by specifying *allowable CoP regions*, represented as a set of polygons, denoted $A = \{A_0, A_1, \dots\}$.

The allowable CoP regions may be used to model stepping stones, as well as to avoid cross-over steps and overlapping footsteps.

Algorithm 1 1-step capture region calculator

Input: $S, r, \dot{r}, A, t_{s,\min}, t, r_{\max}, l_{\max}$

- 1: $r_{ic} \leftarrow Pr + \sqrt{\frac{z_0}{g}} \dot{r}$
- 2: $C_P \leftarrow A$
- 3: **if** $\neg \text{pointInPolygon}(S, r_{ic})$ **then**
- 4: $Q = (q_0, \dots, q_n) \leftarrow \text{visibleVertices}(S, r_{ic})$
- 5: **for** $i = 0$ to n **do**
- 6: $p_i \leftarrow [r_{ic} - q_i]e^{t_{s,\min}-t} + q_i$
- 7: **end for**
- 8: $C_P \leftarrow \text{bound}(C_P, p_0, r_{ic})$
- 9: $C_P \leftarrow \text{bound}(C_P, r_{ic}, p_n)$
- 10: **for** $i = 1$ to n **do**
- 11: $C_P \leftarrow \text{bound}(C_P, p_{i-1}, p_i)$
- 12: **end for**
- 13: **end if**
- 14: **return** $(C_P, r_{\max}, l_{\max})$

Algorithm 1 assumes that the orientation of the swing foot may be chosen without limitation, which means that the maximum distance r_{\max} from the contact reference point to the edge of the foot polygon (see Figure 7.7) fully specifies which contact reference point locations allow a given CoP location to be attained. We further specify a maximum step length l_{\max} , defined as the maximum distance between subsequent contact reference point locations.

The output of the algorithm is a tuple $(C_P, r_{\max}, l_{\max})$, consisting of a set of polygons C_P , the maximum offset distance to these polygons r_{\max} , and the maximum distance to the contact reference point l_{\max} . The basic idea of the algorithm is to determine the set of polygons C_P consisting of the parts of A where the instantaneous capture point may be located when a step is taken, so that the CoP may be placed there after the step, letting the robot reach a captured state. The capture region C is then found by determining allowable contact reference point locations that enable the CoP to be placed inside C_P as shown in Figure 7.7.

In Algorithm 1, S is the current support polygon represented as a clockwise ordered sequence of vertices, r is the CoM position, $t_{s,\min}$ is the estimated earliest time at which a step can be taken, t is the current time, r_{\max} is the maximum distance

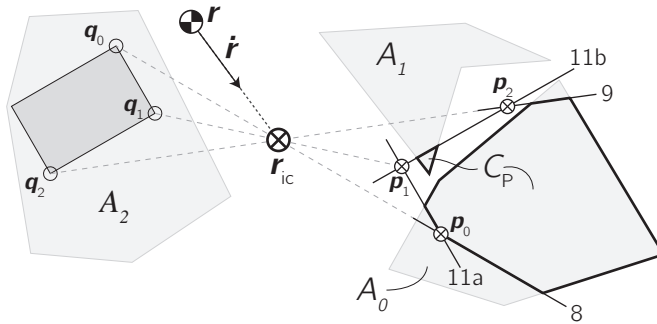


Figure 7.6: Graphical depiction of Algorithm 1. The instantaneous capture point r_{ic} is first calculated (line 1). The ordered set of support polygon vertices that are visible from the instantaneous capture point, (q_0, q_1, q_2) is then determined (line 4), as well as the corresponding possible locations of the instantaneous capture point at $t_{s,min}$, (p_0, p_1, p_2) (lines 5–7). The set of polygons C_P is finally determined by bounding the set of allowable CoP regions, A , using the lines denoted 8, 9, 11a and 11b in the figure. The numbers correspond to line numbers in Algorithm 1.

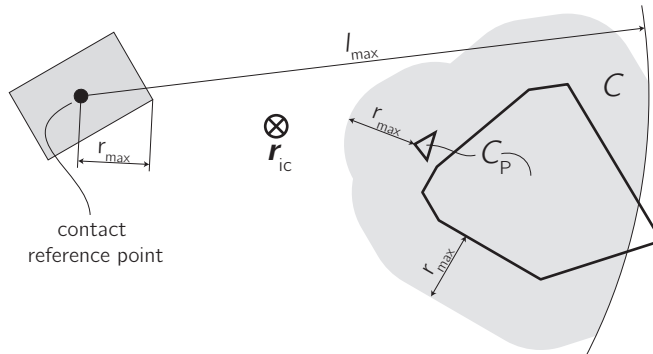


Figure 7.7: The 1-step capture region C , as found by 1) offsetting the set of polygons C_P by r_{max} , and 2) intersecting the result with a disk of radius l_{max} and the contact reference point as its center to take the maximum step length constraint into account. Note that for this figure, the centroid of the foot was chosen as the contact reference point.

from the contact reference point to the edge of the support polygon, and l_{\max} is the maximum step length, as measured from the contact reference point.

Line 1 in the algorithm is the instantaneous capture point definition, (8) in Part 1. Line 6 describes the instantaneous capture point motion for a constant CoP and originates from (22) in Part 1.

The function $\text{pointInPolygon}(X, \mathbf{p})$ determines whether point \mathbf{p} is contained within the polygon defined by the sequence of vertices X .

Function $\text{visibleVertices}(X, \mathbf{p})$ computes a clockwise ordered sequence of vertices $Q = (\mathbf{q}_0, \dots, \mathbf{q}_n)$ taken from X that 'can be seen' from \mathbf{p} , in the sense that a line segment starting from \mathbf{p} and pointing to \mathbf{q}_i does not intersect the polygon spanned by X . The first element of the returned sequence is the rightmost visible vertex as seen from \mathbf{p} , and the last element is the leftmost.

The function $\text{bound}(P, \mathbf{r}_0, \mathbf{r}_1)$ takes a set of polygons P and returns a modified version of P , where each individual polygon is intersected with a half-plane containing all points to the right of a boundary line through \mathbf{r}_0 and \mathbf{r}_1 . Note that the order of the arguments \mathbf{r}_0 and \mathbf{r}_1 determines the 'direction' of the boundary line and hence the direction of the half-plane. This function is used to cut off parts of the allowable CoP regions to which the instantaneous capture point may not be directed or which are not reachable given the minimum step time.

Note that if the current support foot is only partially placed inside the allowable CoP regions A , the support polygon S is the intersection of the foot polygon and A , and Algorithm 1 can be used without modification. Furthermore, note that if there are limits on allowable foot orientations, the step of offsetting the set of polygons C_P by r_{\max} should be replaced by convolving the allowable configurations of the swing foot polygon about C_P .

7.7.3 Desired footstep calculator

The desired footstep calculator (see Figure 7.8) determines the desired position of the swing ankle \mathbf{r}_{step} and orientation of the swing foot \mathbf{R}_{step} at the end of the upcoming step. The desired position and orientation are expressed in a frame fixed to the support foot. Because footstep planning depends greatly on the control task, we have created two separate implementations of this module: one for the balancing task and another for the walking task. The general pattern used for both implementations is to choose a desired initial footstep $(\mathbf{r}_{\text{step,init}}, \mathbf{R}_{\text{step,init}})$ at the start of the swing phase and adjust it during the swing phase if necessary, for example if the robot is significantly perturbed, to obtain the final output $(\mathbf{r}_{\text{step}}, \mathbf{R}_{\text{step}})$.

Balancing

The desired footstep calculator for the balancing task is a minimal implementation. The initial desired step location is computed using a fixed step length l_{step} in the direction of the instantaneous capture point:

$$\mathbf{r}_{\text{step,init}} = \mathbf{r}_{\text{ankle}} + l_{\text{step}} \frac{\mathbf{r}_{\text{ic}} - \mathbf{r}_{\text{ankle}}}{\|\mathbf{r}_{\text{ic}} - \mathbf{r}_{\text{ankle}}\|} \quad (7.1)$$

where $\mathbf{r}_{\text{ankle}}$ is the position of the stance foot ankle. The initial foot orientation is chosen to be the same as the stance foot orientation and is never adjusted. The direction in which the robot steps is adjusted by recomputing (7.1) during the first 0.1 s after the instantaneous capture point has left the foot polygon. After that it remains fixed for the remainder of swing.

Walking

The desired footstep for the walking task is chosen in such a way that forward progression is made, while the robot remains 1-step capturable. The initial step length is determined by multiplying the desired walking velocity by a proportional gain, while the initial step width is set to a constant value. On flat ground, the step height is set to zero but may be changed to any feasible desired value when necessary, for instance to climb a slope. Footstep yaw is set equal to the desired walking direction. Footstep pitch and roll are set to zero on flat ground but may be used to walk on rough terrain to reduce the need for compliance in the ankle joint.

If the initial desired step location ceases to be within the capture region by a desired margin at any instant during the swing phase, an adjusted desired step location is computed by projecting the initial location inside the 1-step capture region by the desired margin. The orientation is not adjusted.

7.7.4 Swing sub-controller

The task of the swing sub-controller (see Figure 7.9) is to compute the desired torques for the swing leg joints. It accommodates changes in the desired step location, obtained from the desired step location calculator, on the fly. The controller contains a trajectory generator for the position and orientation of the swing foot, which is used to compute desired joint angles, velocities and accelerations through inverse kinematics. The trajectory is tracked using inverse dynamics augmented by PD position control in joint space.

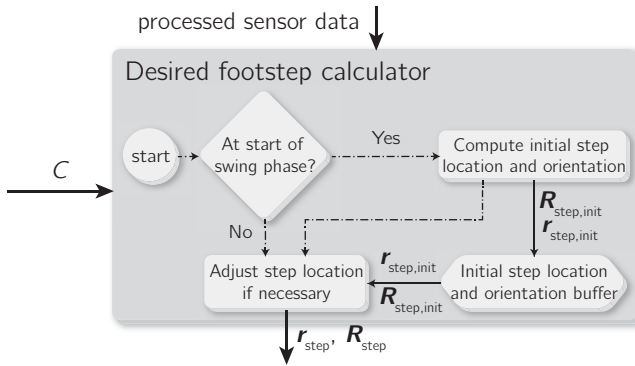


Figure 7.8: Basic structure of the desired footstep calculator. The desired footstep calculator determines the desired position r_{step} and orientation R_{step} of the foot for the upcoming step. Solid lines represent data flow and dash-dotted lines represent flow of control.

Trajectory generation

The swing foot trajectory is updated, even in mid swing, to allow for changes in step location in reaction to external disturbances. We chose a second-order system as a reference model to determine the desired linear position $x_d \in \mathbb{R}^3$ and velocity \dot{x}_d of the ankle of the swing foot at each control time step by integrating an appropriately chosen desired acceleration \ddot{x}_d . Initial values for x_d and \dot{x}_d are set equal to the actual position and velocity of the swing foot ankle.

The swing foot reference model goes through three phases: take-off, cruise, and landing. Limits are placed on the magnitudes of acceleration and velocity, and we specify a clearance height and a take-off and landing slope. During the take-off phase, the desired acceleration \ddot{x}_d is chosen to have maximal magnitude until the velocity limit is reached, and be directed at the current desired step location r_{step} , while moving upwards using the specified take-off slope. Once the minimum clearance height is reached, the dynamic system transitions into the cruise phase, in which \ddot{x}_d is chosen to maintain constant height while still adjusting for changes in the desired step location. The transition into the landing phase takes place based on the landing slope, and the desired position decelerates maximally while the height is gradually reduced according to the landing slope.

For the orientation trajectory of the swing foot, we use a simple interpolation between the measured orientation at the start of the swing phase and the desired final foot orientation, R_{step} . The interpolation parameter is obtained from a quintic spline starting and ending with zero velocity and acceleration, and is not updated during

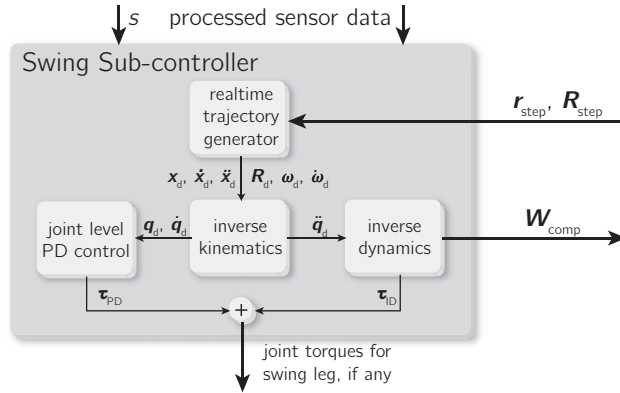


Figure 7.9: Swing sub-controller. A real-time trajectory generator computes the desired position, linear velocity and acceleration ($x_d, \dot{x}_d, \ddot{x}_d$ respectively), as well as the desired orientation, angular velocity and angular acceleration ($R_d, \omega_d, \dot{\omega}_d$ respectively), expressed in world frame. Inverse kinematics is used to translate this trajectory from Euclidean space to desired joint positions, velocities and accelerations (q_d, \dot{q}_d and \ddot{q}_d respectively). This information is used by an inverse dynamics algorithm, which computes pin joint torques τ_{ID} and an upper body compensation wrench W_{comp} , used by the stance sub-controller. The output τ_{PD} of a joint-space PD controller is added to τ_{ID} to obtain the swing leg joint torques.

the step.

Position control

The desired position and orientation of the swing foot at every control time step are used to compute the corresponding joint angles of the swing leg using inverse kinematics. The desired linear and angular velocity and acceleration of the swing foot specify corresponding desired joint velocities and accelerations through the inverse of the swing leg Jacobian. Problems due to the singularity that occurs when the knee is stretched are circumvented by gradually scaling the desired joint velocities and accelerations back to zero (that is, using pure damping) based on the value of the Jacobian determinant.

An inverse dynamics algorithm (Featherstone, 2008), augmented by PD position control in joint space is used to compute the desired torques for the swing leg joints. We exclude the stance leg joints in computing the inverse dynamics. The desired spatial acceleration of the upper body is set to zero. In addition to the torques across the swing leg pin joints, the inverse dynamics algorithm also returns a wrench that should be exerted across the 'floating joint' that connects the upper body to the world to achieve the desired zero spatial acceleration. This wrench will be used in the

stance sub-controller as a feed-forward term to compensate the swing leg torques and reduce upper body oscillations.

Previously, we have also implemented a virtual model control-based swing sub-controller, but the current implementation outperforms the virtual model control implementation in terms of accuracy and swing speed, which we consider to be key ingredients for dynamic walking and push recovery.

7.7.5 Stance sub-controller

The stance sub-controller controls balance by computing desired torques for the stance leg(s). The goals of the stance sub-controller are to control 1) instantaneous capture point location, 2) upper body orientation, and 3) upper body height. The stance sub-controller is an implementation of virtual model control. For each stance leg, it computes a desired wrench on the upper body, to be exerted by the leg, that satisfies these control goals. Jacobian transpose control is then used to find corresponding desired leg torques. The stance sub-controller consists of multiple control modules, as shown in Figure 7.10. The following sections will describe these modules in more detail.

Upper body height control module

The upper body height control module regulates the upper body height by determining the net vertical force, f_z , that the legs should exert on the upper body. In double support, f_z is set to a constant value that slightly overcompensates the estimated gravitational force acting on the entire robot, simulating a virtual constant force spring. In single support, f_z is set to the weight of the stance leg and the body plus the z-component of the force from the swing leg compensation wrench \mathbf{W}_{comp} , cancelling out some of the dynamic effects due to the swing leg motion.

Upper body orientation control module

The upper body orientation control module determines a desired torque $\boldsymbol{\tau}$ that the legs should exert on the upper body. It is used to control the orientation of the upper body with respect to the world (as perceived by the inertial measurement unit). The desired pitch of the upper body is constant and set to zero. Both the desired yaw and the desired roll depend on the gait phase and state of the support leg. Yawing and rolling are used to obtain a longer reach for the swing leg and to make the gait look more humanlike. In addition, the desired yaw also depends on the desired walking direction.

The desired upper body torque is computed using PD control on the roll, pitch and yaw corresponding to the rotation matrix that describes the orientation of the actual

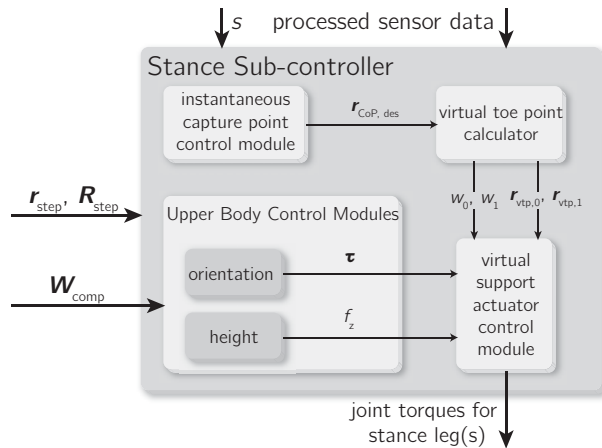


Figure 7.10: Stance sub-controller. The instantaneous capture point control module computes the desired CoP ($r_{CoP,des}$) within the base of support. This desired CoP is used by the virtual toe point calculator to compute leg support fractions (w_0, w_1) and virtual toe points ($r_{vtp,0}, r_{vtp,1}$), which determine each leg's contribution to supporting the upper body. The orientation and height upper body control modules determine the torque τ and the vertical force f_z to be exerted on the upper body. In single support, these modules use the swing leg compensation wrench W_{comp} to compensate swing leg motion. Finally, the virtual support actuator control module computes joint torques for the stance legs which result in the desired virtual toe points, leg strengths, upper body torque and force.

upper body with respect to the desired orientation. In single support, the torque part of W_{comp} is added to the result to compensate the swing leg motion.

A lunging strategy based on the results of the 3D-LIPM with finite-sized foot and reaction mass has not yet been implemented and is part of future work.

Instantaneous capture point control module

The goal of the instantaneous capture point control module is to regulate the location of the instantaneous capture point. Its output is the desired location of the CoP, $r_{CoP,des}$. This control module switches between two modes of operation, depending on whether the instantaneous capture point is inside the support polygon.

Instantaneous capture point inside support polygon

See Figure 7.11. When the instantaneous capture point is inside the support polygon, a *desired instantaneous capture point*, $r_{ic,des}$, is determined. Given the current and desired location of the instantaneous capture point, the following simple control law

is used to obtain the tentative location of the desired CoP, $\bar{\mathbf{r}}_{\text{CoP,des}}$:

$$\bar{\mathbf{r}}_{\text{CoP,des}} = \mathbf{r}_{\text{ic}} + k_{\text{ic}}(\mathbf{r}_{\text{ic}} - \mathbf{r}_{\text{ic,des}}) \quad (7.2)$$

where k_{ic} is the proportional gain. This proportional control law is motivated by the linear instantaneous capture point dynamics for the 3D-LIPM with finite-sized foot described in Part 1. If $\bar{\mathbf{r}}_{\text{CoP,des}}$ lies inside the support polygon, then the final output of this control module is $\mathbf{r}_{\text{CoP,des}} = \bar{\mathbf{r}}_{\text{CoP,des}}$. Otherwise, $\mathbf{r}_{\text{CoP,des}}$ is obtained by projecting $\bar{\mathbf{r}}_{\text{CoP,des}}$ to the edge of the support polygon along a line through \mathbf{r}_{ic} and $\mathbf{r}_{\text{ic,des}}$, as shown in Figure 7.11b.² The idea behind this control law is that the instantaneous capture point is always pushed away from the CoP, and hence towards the desired instantaneous capture point.

The desired instantaneous capture point is determined as a function of the state and the control task. For the balancing task, the desired instantaneous capture point coincides with the centroid of the support polygon during the double support state. When the robot is commanded to start balancing on one leg, the desired instantaneous capture point is moved to the centroid of the upcoming support foot, where it remains as long as the robot is able to maintain its balance without taking a step. This location maximizes robustness against external disturbances from unknown directions.

For the walking task, the desired instantaneous capture point is located near the toes of the leading foot during the double support states, promoting forward motion. At the start of the swing phase, the desired capture point is moved outside the stance foot in the direction of the upcoming step location.

Instantaneous capture point outside support polygon

See Figure 7.12. The instantaneous capture point will move outside the support polygon if the robot is pushed significantly, or because it has been driven outside the stance foot polygon during the walking task. When \mathbf{r}_{ic} is outside the support polygon, it is not possible to track a desired instantaneous capture point location, since the instantaneous capture point will always exponentially diverge away from the stance foot. We therefore only control the direction in which it diverges away from the foot. This is done by specifying a *guide line*, L_g , and choosing the desired CoP such that the instantaneous capture point is kept along this guide line.

²In practice, we use a slightly smaller support polygon when projecting the tentative desired CoP, to prevent the feet from tipping at times when this is not desired. The use of a smaller support polygon is necessary because of unmodeled dynamics, inability to perfectly track the desired CoP, and model uncertainty.

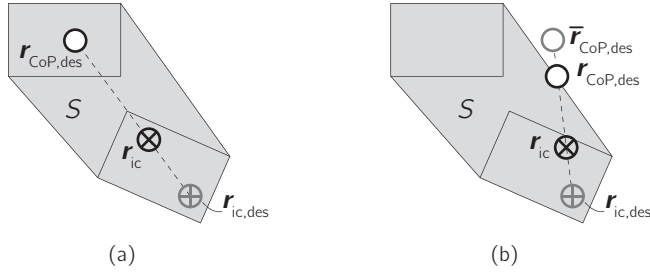


Figure 7.11: Action of the instantaneous capture point control module when r_{ic} is inside the support polygon S . (a) the tentative desired CoP $\bar{r}_{CoP,des}$, as determined using (7.2), is inside the support polygon, so it coincides with the final output $r_{CoP,des}$. (b) $\bar{r}_{CoP,des}$ is outside the support polygon and is projected onto its edge to obtain $r_{CoP,des}$.

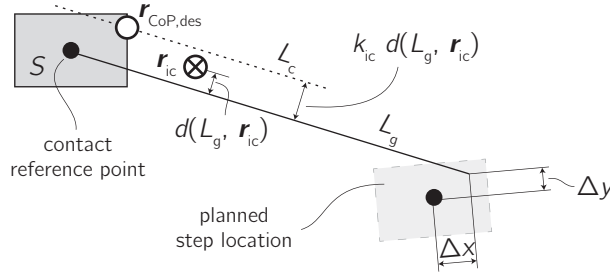


Figure 7.12: Action of the instantaneous capture point control module when r_{ic} is outside the support polygon S . The guide line L_g passes through the contact reference point on the support foot and a point Δx in front and Δy inside of the contact reference point for the planned footstep location, see (7.3). The control line L_c is parallel to the guide line; the distance between the two is determined by (7.4). The desired CoP is chosen as the intersection of L_c and S .

We define the guide line by two points. The first is the contact reference point on the support foot. The second is a point Δx in front and Δy to the inside of the contact reference point for the planned footstep, with

$$\Delta x = k_{xx} v_{des,x} \quad (7.3a)$$

$$\Delta y = k_{xy} |v_{des,x}| \quad (7.3b)$$

where k_{xx} and k_{xy} are positive gains and $v_{des,x}$ is the desired average velocity of the robot in the forward direction, in a frame oriented to match the planned step location. The effect of this simple control law is that the instantaneous capture point is pushed forward and to the inside of the upcoming support foot as desired forward

velocity is increased.

Given the guide line and the instantaneous capture point location, a second line, called the control line, is defined. The control line is parallel to the guide line. The distance $d(L_g, L_c)$ between the guide line L_g and the control line L_c is set to be proportional to the distance between the guide line and the instantaneous capture point, $d(L_g, r_{ic})$:

$$d(L_g, L_c) = k_{ic}d(L_g, r_{ic}) \quad (7.4)$$

where k_{ic} is a positive gain.

Finally, the desired CoP is computed by finding the intersection of the stance foot polygon and the control line that is closest to r_{ic} . If there are no intersections, the support polygon vertex closest to the control line is used. The control law (7.4) causes the instantaneous capture point to be pushed back onto the guide line after a deviation.

Virtual toe point calculator

The virtual toe point calculator uses the desired CoP to compute a *virtual toe point* (VTP) and a *leg support fraction* for each leg (Pratt and Pratt, 1998). Controlling VTP locations and leg support fractions results in approximate control of the overall CoP of the robot.

The virtual toe point of a foot is the point about which no torque is commanded in the horizontal plane. VTPs are similar to the centers of pressure for each foot, except that a VTP is a commanded quantity, not a measured one, and is only based on a static analysis. Details on how the VTPs are used are given in Section 7.7.5.

In single support, the VTP for the stance leg is placed at the location of the desired CoP. In double support, we use a heuristic based on geometric relations, depicted and explained in Figure 7.13, to determine VTPs $r_{vtp,0}$ and $r_{vtp,1}$ in such a way that the desired CoP and both VTPs lie on one line.

The leg support fractions are two scalars, denoted w_0 and w_1 , such that $w_0 + w_1 = 1$ and $w_0, w_1 \in [0, 1]$. A leg support fraction w_i describes the fraction of the desired torque τ and vertical force f_z to be exerted by stance leg $i \in \{0, 1\}$. The distances between the VTPs and the overall desired CoP determine the leg support fractions:

$$w_i = \frac{\|r_{CoP,des} - r_{vtp,1-i}\|}{\|r_{vtp,0} - r_{vtp,1}\|}, \quad i \in \{0, 1\} \quad (7.5)$$

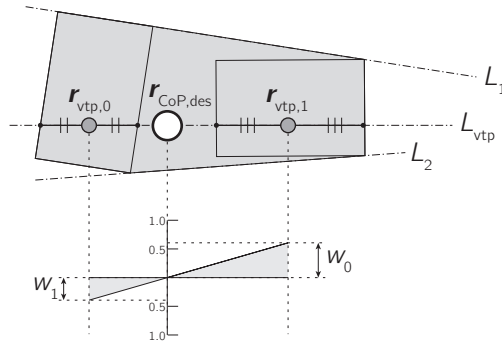


Figure 7.13: Implementation of the virtual toe point calculator during double support. Line L_{vtp} is first constructed. It passes through the desired CoP and the intersection of lines L_1 and L_2 , the edges of the support polygon that connect the foot polygons. The intersections of line L_{vtp} with the foot polygons define two line segments. The VTPs $r_{vtp,0}$ and $r_{vtp,1}$ are found as the center points of these line segments. The leg support fractions are then found based on the distances between the VTPs and the desired CoP using (7.5), which also has a clear geometric interpretation as shown in the figure. If the desired CoP lies on the outside of a foot, so that it lies farther to the edge of the support polygon than the center of the line segment for that foot, then the VTP for that foot is chosen to be equal to the desired CoP and the leg is assigned a leg support fraction of 1 (this case is not shown in the figure).

This equation stems from a moment balance around the desired CoP: if a VTP is far removed from the desired CoP, a force exerted at this VTP has a large moment arm, and hence the associated leg support fraction should be small. Support is gradually transferred from one leg to the other in double support due to continuously changing leg support fractions, associated with a continuous desired CoP trajectory.

Virtual support actuator control module

The virtual support actuator control module distributes the torque $\boldsymbol{\tau}$ and the vertical force f_z over the support leg(s) using the leg support fractions w_i as weighting factors:

$$\begin{aligned} f_{z,i} &= w_i f_z & i \in \{0, 1\} \\ \boldsymbol{\tau}_i &= w_i \boldsymbol{\tau} \end{aligned} \quad (7.6)$$

where $f_{z,i}$ and $\boldsymbol{\tau}_i$ are the z-component of the force and the torque to be exerted by leg i , respectively. We use these partial wrenches to compute a complete desired wrench \mathbf{W}_i for each leg, where

$$\mathbf{W}_i = \begin{pmatrix} \mathbf{f}_i \\ \boldsymbol{\tau}_i \end{pmatrix} \quad \text{with} \quad \begin{aligned} \mathbf{f}_i &= (f_{x,i}, f_{y,i}, f_{z,i})^T \\ \boldsymbol{\tau}_i &= (\tau_{x,i}, \tau_{y,i}, \tau_{z,i})^T \end{aligned} \quad (7.7)$$

The remaining x - and y -components of the force \mathbf{f}_i for each leg are computed using the virtual toe points. The virtual toe point constraint, which requires that no torque should be applied about either horizontal axis at the virtual toe point, can be enforced as follows. We consider the virtual toe point for a foot to be the intersection of the axes of two virtual pin joints, located on the sole of the foot. Their orthogonal axes of rotation lie in the plane of the foot. The virtual pin joints do not exist on the physical robot, but provide a simple way of computing $f_{x,i}$ and $f_{y,i}$ since the torques across these joints should be zero. The virtual pin joints come after the real joints of the robot in the kinematic chain from upper body to foot. Their rotation angles are set to zero, but their location on the foot changes in time, depending on the location of the virtual toe point. We use $\boldsymbol{\tau}_{\text{vtp},i} \in \mathbb{R}^2$ to denote the vector of virtual joint torques exerted at the virtual pin joints for stance leg i . A static analysis results in

$$\boldsymbol{\tau}_{\text{vtp},i} = \mathbf{J}_{\text{vtp},i}^T \mathbf{W}_i. \quad (7.8)$$

In this equation, $\mathbf{J}_{\text{vtp},i} \in \mathbb{R}^{6 \times 2}$ is the Jacobian that maps the joint velocities of the virtual pin joints to the twist of the foot with respect to a virtual body attached 'after' the virtual pin joints in the kinematic chain, expressed in an upper body-fixed frame.

Splitting the Jacobian $\mathbf{J}_{\text{vtp},i}$ into a 2×2 block $\mathbf{J}_{\text{vtp},i,2 \times 2}$ and a 4×2 block $\mathbf{J}_{\text{vtp},i,4 \times 2}$ and using (7.7), we can rewrite (7.8) as

$$\boldsymbol{\tau}_{\text{vtp},i} = \mathbf{J}_{\text{vtp},i,2 \times 2}^T \begin{pmatrix} f_{x,i} \\ f_{y,i} \end{pmatrix} + \mathbf{J}_{\text{vtp},i,4 \times 2}^T \begin{pmatrix} f_{z,i} \\ \boldsymbol{\tau}_i \end{pmatrix}. \quad (7.9)$$

We require that the torques at a leg's virtual toe point be zero. We can hence solve (7.9) to find the values of $f_{x,i}$ and $f_{y,i}$:

$$\begin{pmatrix} f_{x,i} \\ f_{y,i} \end{pmatrix} = -\mathbf{J}_{\text{vtp},i,2 \times 2}^{-T} \mathbf{J}_{\text{vtp},i,2 \times 4}^T \begin{pmatrix} f_{z,i} \\ \boldsymbol{\tau}_i \end{pmatrix} \quad (7.10)$$

The matrix $\mathbf{J}_{\text{vtp},i,2 \times 2}^T$ is invertible as long as the following two conditions hold: 1) the z -coordinate of the origin of the upper body, expressed in the foot-fixed frame, is not zero, and 2) the z -axis of the upper body-fixed frame does not lie in the horizontal plane of the foot-fixed frame. These conditions are always satisfied during normal operation.

Now that $f_{x,i}$ and $f_{y,i}$ are also known, we know the complete wrench \mathbf{W}_i to be exerted on the upper body by stance leg i , and we can use a different Jacobian, $\mathbf{J}_{\text{leg},i} \in \mathbb{R}^{6 \times 6}$

to find the joint torques $\tau_{\text{leg},i}$:

$$\tau_{\text{leg},i} = J_{\text{leg},i}^T W_i, \quad (7.11)$$

where $J_{\text{leg},i}$ maps the joint velocities of the *real* joints of leg i to the twist of the upper body with respect to the foot, expressed in the upper body-fixed frame.

Computing $f_{x,i}$ and $f_{y,i}$ based on virtual toe points instead of specifying these forces directly has as an advantage that virtual toe points are closely related to the CoP, which plays a major role in the instantaneous capture point dynamics described in Chapter 3. This relation to the CoP also means that limits due to the finite-sized support polygon are straight-forward to accommodate. We simply make sure that each foot's virtual toe point lies inside its convex polygon.

7.8 Results

This section presents results obtained for both the balancing task and the walking task. Balancing and walking without pushes was achieved on the real M2V2 robot. Walking while recovering from pushes and walking over stepping stones was achieved on the simulated M2V2 robot. Note that figures are labeled either [REAL] if the data is from the physical robot or [SIM] if from the simulated robot. See Extension 1 for a video of results from both real-world experiments and simulations.

In this section, we evaluate the N -step capturability margin proposed in Part 1, that is, the area of the N -step capture region. Here we use $N = 1$ since the presented algorithm is based on remaining 1-step capturable.

7.8.1 Balancing task

On the real M2V2 robot we achieved balancing on one leg and recovering from sideways and forward pushes. Figure 7.14 shows time-elapsed images of M2V2 recovering from a push. Figure 7.15 shows the norm of the disturbance force, as recorded from the push stick, and the 1-step capturability margin. We see that the robot was able to recover from pushes of approximately 21 Ns.

7.8.2 Walking task

On the real M2V2 robot we achieved flat ground walking without disturbances. On the simulated M2V2 robot we achieved walking while recovering from pushes and walking over stepping stones.

Figure 7.16 shows time elapsed images of M2V2 walking on flat ground. For this walk, the robot uses a constant step length and width. During the walk, the capture

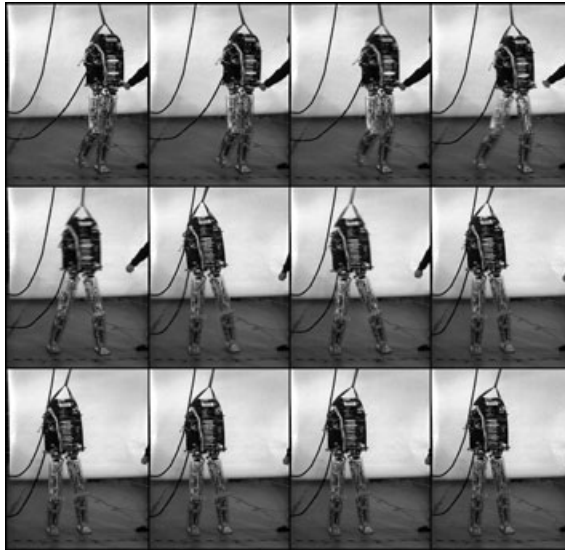


Figure 7.14: [REAL] M2V2 recovering from a push while standing on one leg. Images are from left to right starting at the top left.

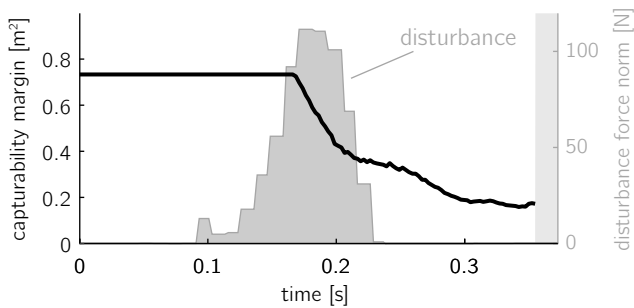


Figure 7.15: [REAL] Norm of disturbance force and 1-step capturability margin of M2V2 recovering from a push while standing on one leg. The capturability margin is not shown after transition to the 'stop in double support' state (gray area) to avoid cluttering.

region is computed, but since there are no pushes, the robot does not have to change where it steps.

Figures 7.17 through 7.20 show plots of a single data set obtained from simulation for walking on flat ground while recovering from pushes. Three different pushes to

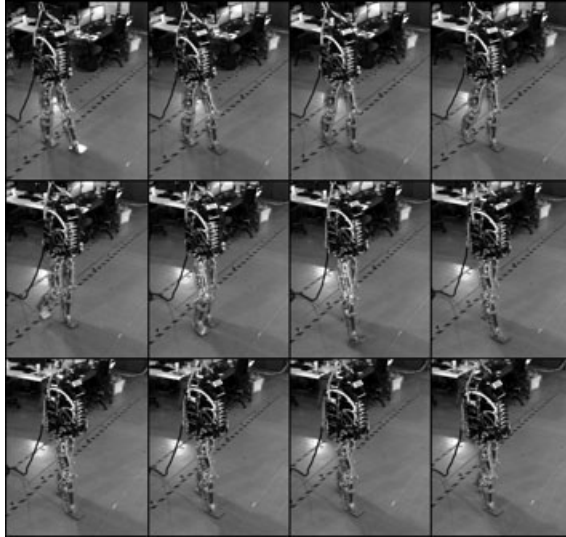


Figure 7.16: [REAL] M2V2 robot walking on flat ground. Images are from left to right starting at the top left. In this walk, the robot uses a constant step length of 0.35 m and a constant step width of 0.20 m. The average walking velocity is 0.21m/s.

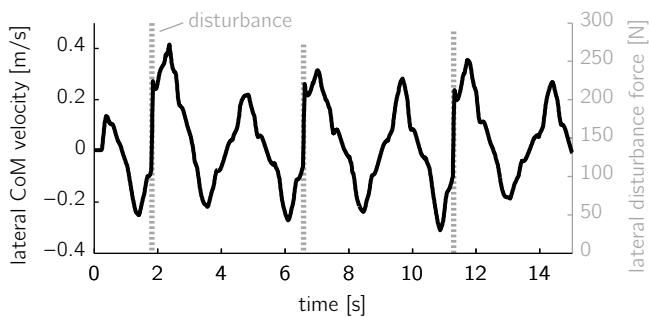


Figure 7.17: [SIM] Lateral velocity and disturbances of the M2V2 simulation while recovering from pushes during walking. After each push, the lateral CoM velocity increases by up to 0.44 m/s.

the left occur at approximately 4, 8.5, and 13 seconds. These pushes are modeled as forces applied to the midpoint between the hip joints and are 300 N in magnitude for a duration of 0.05 seconds. This corresponds to an impulse of 15 Ns. Note that pushes were to the left while the left foot was swinging. Pushes to the opposite side would require either a cross-over step or two quick steps, both of which are more

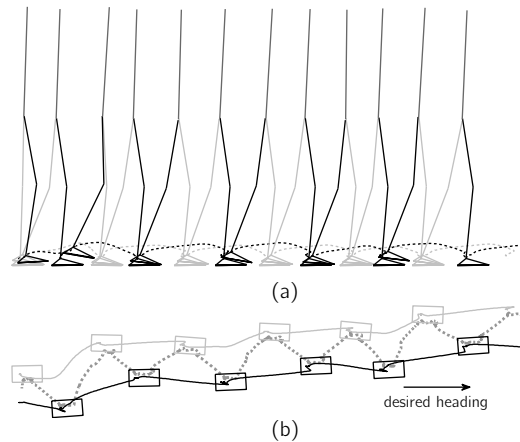


Figure 7.18: [SIM] Robot as it walks while being pushed laterally to the left every second step. (a) Side view. Trajectories of the ankles are indicated with dashed lines. (b) Overhead view. Actual ankle trajectories connect the sequence of footprints for each foot. The instantaneous capture point trajectory is indicated with a dashed line.

difficult to achieve and are an area of future work.

Figure 7.17 shows the sideways pushes applied to the M2V2 simulation, and the resulting change in velocity while recovering to pushes during walking. Since the pushes were mostly to the side, the change in lateral velocity is more prominent than forward velocity. After each push we see that the robot recovers with one step.

Figure 7.18 shows side and overhead views of the robot. Each time a push occurs, the robot steps to the left to recover from the push. Also plotted in the overhead view are the instantaneous capture point trajectory and the trajectory of each ankle.

Figure 7.19 shows the 1-step capturability margin during walking of the M2V2 simulation while recovering from pushes. We see that the capturability margin significantly decreases after each of the 3 pushes, corresponding to the decrease in area of the capture region as seen in Figure 7.20.

Figure 7.20 shows a time-lapsed overhead view of the robot. In the first two frames the capture region is relatively large during the beginning of swing. The robot is then pushed between the first and second frame, decreasing the size of the capture region and requiring the robot to choose a different place to step. In frames 3-6 the robot steps further to the left than originally intended, landing in the 1-step capture region, and successfully recovering from the push.

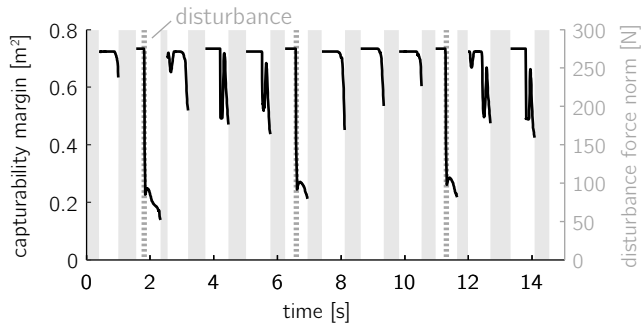


Figure 7.19: [SIM] 1-step capturability margin during walking while recovering from pushes. After each push, the 1-step capturability margin significantly decreases showing that the robot is in danger of falling. After each recovery step, the capturability margin recovers. Note that during double support and periods during single support when the instantaneous capture point is inside the support polygon the capturability margin is not plotted (gray areas).

Walking over various stepping stones was achieved on the simulated M2V2 robot. Figure 7.21 shows an example of walking over stepping stones that are clustered in groups of three. The robot was given exact knowledge of the stepping stones. To ensure that the entire foot rested on each stone, the allowable CoP region for each stepping stone was computed by shrinking the stepping stone based on the size of the foot. However, due to inaccuracies in foot placement, the foot would sometimes slightly overhang the edge of a stone.

7.9 Discussion and future work

7.9.1 Using simple models for complex robots

In this part we used the 3D-LIPM with finite-sized foot, described in Part 1, to estimate the 1-step capture region, which was then used to help control balancing and walking in M2V2, a 3D lower body humanoid. This simple model was sufficient for controlling balancing on one foot, walking, and recovering from pushes while walking. The simplified model was sufficient for three reasons:

- The model accounts for the dynamics of the CoM with respect to the CoP, which are the key dynamics of walking.
- The model allows for the use of feet, and the modulation of the CoP in real time. This adds robustness, as opposed to an algorithm that predetermines a CoP trajectory.

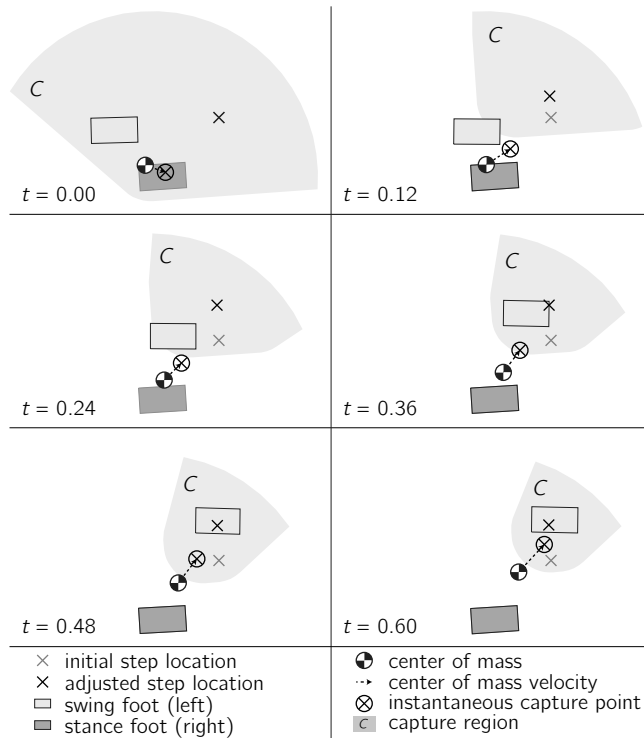


Figure 7.20: [SIM] Time-lapse overhead view of the robot during the walking task, showing one step. The robot is perturbed laterally at the start of the step ($t = 0$).

- The 1-step capture region is relatively large for moderate speed walking. Therefore, there is a large degree of robustness to modeling errors.

The presented control algorithms did not exploit angular momentum of the upper body as a means of control. As we start to address more challenging tasks, like walking over rough terrain and over narrow beams, we will likely need to use more complex models for computing capture regions and developing control strategies. In Part 1, we analyzed the 3D-LIPM with finite-sized foot and reaction mass. This model should be useful in control when upper body angular momentum is used to prevent falling. In future work we will use this model and will investigate other strategies for using multi-joint upper body angular momentum in walking and disturbance recovery.

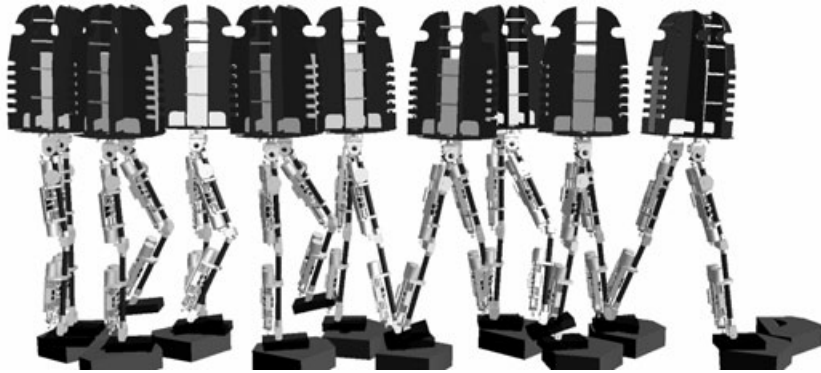


Figure 7.21: [SIM] M2V2 walking over stepping stones that are clustered in groups of three. Snapshots are taken at 1.5 second intervals.

7.9.2 1-step versus N -step capture regions

In this part we developed controllers that always step into the 1-step capture region. This restriction is overly cautious in general, and it is likely the case that fast walking may require periods when the legged system is only 2-step, or perhaps only 3-step capturable. However, M2V2 currently has a long swing time, making the 2-step capture region not much larger than the 1-step capture region.

For M2V2, the falling time constant is $1/\omega_0 = \sqrt{z_0/g} \approx 0.32$ seconds, where $g = 9.81m/s^2$ is the gravitational acceleration and $z_0 \approx 1.0m$ is the CoM height. The minimum step time Δt_s is currently approximately 0.6 seconds for M2V2. The geometric ratio governing the diminishing returns for the N -step capture region as N increases is $\exp(-\sqrt{z_0/g}\Delta t_s) \approx 0.15$ (see (16) in Part 1). This means that the radius of the 2-step capture region is only about 15% larger than the radius of the 1-step capture region. Therefore, there is not much to be gained in considering 2-step capturability over 1-step capturability with M2V2, until we can get the robot to swing its legs more quickly.

With human walking, on the other hand, the minimum swing time is approximately 0.3 seconds. This results in a geometric ratio of approximately 0.4. Hence, the 2-step capture region for human walking should be relatively large, and even 3-step capturability should be considered.

7.9.3 Capturability margin

We presented an experimental evaluation of the N -step capturability margin, where we used $N = 1$. Figure 7.15 and Figure 7.19 showed a significant decrease in

this capturability margin when the robot was perturbed, as expected. For both the balancing task and the walking task, the capturability margin recovered completely after taking one step.

According to the theoretical considerations presented in Part 1, the capturability margin should never increase unless a step is taken or an external force is exerted on the robot. Any other increases in capturability margin are due to modeling errors, sensor noise and lack of exact knowledge about when the swing foot hits the ground.

7.9.4 Estimation of center of mass position and velocity

The algorithm presented, like most feedback control algorithms for bipedal walking, relies on a good estimation of the CoM location and velocity, particularly in the horizontal plane. Obtaining such an estimate on a physical robot is difficult for several reasons.

Knowing the CoM projection on the ground requires knowing which way is down. A small error in the perceived orientation of the body can result in a significant error in the CoM projection. For example, if the CoM is at a height of 1 meter and there is a 0.01 radian error in the body orientation, that will result in approximately a 10 mm error of the CoM on the ground. For 3D robots, orientation is typically determined using an inertial measurement unit (IMU), and therefore having a good IMU and related sensor processing is important.

Using leg kinematics to estimate the CoM velocity has the problem that one must assume that the foot is not moving. However, if the foot is slipping, this assumption will make the CoM seem to be moving in the opposite direction. On the other hand, integrating an accelerometer to estimate velocity has the problem of error accumulation. For M2V2 we have used a combination of leg kinematics and accelerometer measurements. We have not yet determined how accurately our CoM velocity estimate is, but we believe we can do much better and therefore improve performance on the physical robot.

7.9.5 Uneven ground

Both the models presented in Part 1 and the control algorithm presented in Part 2 assumed flat ground. We have performed some preliminary simulations with moderate slopes and relatively small steps, and so far it appears that the same models are applicable to these types of terrain. However, for large slopes and steps, and very rough ground, the models will likely need to be expanded and further control strategies will need to be developed. In addition, we still need to develop capturability-based models and control strategies for situations where hands can push against walls or

hold handrails, or where feet can be on different slopes. In these cases concepts like force closure will need to be taken into account, rather than only using the CoP.

7.9.6 Controlling velocity versus coming to a stop

Capturability measures the ability of a legged system to come to a stop in N steps or fewer. However, we are not usually interested in coming to a stop, but rather maintaining an average speed. Even though capturability is based on the ability to come to a stop, using tools based on capturability does not *require* the legged system to come to a stop. Instead, capturability only specifies bounds on what is permissible. Within these bounds, there is ample room for such things as controlling velocity, for example by CoP or foot placement.

7.9.7 Cross-over steps

In this study, we only considered pushes during single support which did not require a cross-over step. Cross-over steps are challenging for a number of reasons. The swinging leg needs to make sure to not contact the support leg. To do that, the path of the leg may be longer, requiring longer swing time. Also, the length of the step will be smaller than it can be when the leg swings to the outside. An alternative is to quickly step straight down with the currently swinging leg and then quickly swing the other leg to prevent a fall. This two-step recovery strategy requires extra time to execute and for significant pushes will likely only be successful for robots with a relatively short swing time, on the order of how fast humans can swing their legs.

7.9.8 Virtual toe points and center of pressure

The presented control algorithm relies on the use of virtual toe points to track a desired CoP. Virtual toe points can be interpreted as the attachment points of virtual actuators on the feet, and each virtual toe point is used to gain approximate control of each individual foot's CoP. The virtual toe point and a foot's CoP will be theoretically identical if the robot is in single support, the vertical force of the virtual actuator equals the weight of the robot, and the vertical acceleration is zero. In simulation, the two points always remain close (within a few cm) during single support. During double support, there can be a large error between the desired and actual CoP, particularly when one of the legs is stretched. For example, if the hind leg is completely stretched, and the desired CoP is on the heel of the hind leg, then the virtual actuator on the hind leg will be assigned a large leg support fraction. However, since the leg is straight, the actual joint torques that the virtual actuator produces will be small, and the CoP will be located more forward than desired.

One way to obtain a better match between the desired and actual CoP is to keep the knees of the robot bent to avoid losing kinematic range. However, we wish to avoid that solution since human walking does not rely on bent knees and because it requires unnecessarily high torques at the knees. Another solution, which we will investigate in future work, is to use toe off on the rear leg to better control the CoP during double support. Currently, some toe off occurs at the end of the stride, but it is simply the result of the dynamics of the walk, rather than the result of an intentional control action.

7.9.9 Foot placement speed and accuracy

In this paper, we showed how foot placement can be used to regain balance after a push. Doing so requires a fast swing that is accurate enough to make the foot land in the capture region. However, due to the use of SEAs with very compliant springs, we have had difficulty to quickly and accurately swing the leg. Although SEAs enable compliant control, they can make joint position trajectory tracking challenging. We believe that we can achieve the same favorable compliant control characteristics and better tracking by increasing the stiffness of the series springs. On the other hand, we also believe that swing can be performed in a more compliant manner, determined mostly by the passive pendulum dynamics of the leg, as opposed to using traditional high-gain trajectory tracking. Determining swing strategies that allow for fast and accurate steps while exploiting the natural dynamics of the leg is an area of future work.

7.9.10 Application to other robots

We believe that capturability concepts can be applied to the analysis and control of other legged systems. Estimating capture regions and determining capturability-based robustness metrics should be possible for all legged systems. While we advocate compliant force control for legged robots, most of the techniques described in this paper should also apply to high-gain position trajectory tracking robots. Stepping strategies that take the capture region into account should be applicable to any robot that can change where it steps on-the-fly. Control of the instantaneous capture point should be applicable to any robot that can control its CoP location on the ground. We are currently expanding the algorithms presented in this paper and working toward their application on several different humanoid robot platforms.

7.10 Conclusion

This paper showed an application of the N -step capturability framework to a 12-DoF bipedal robot. The main contributions of this part were step location adjustment

using the 1-step capture region and instantaneous capture point control by CoP placement, which are key ingredients to the presented control algorithm. In addition, the 1-step capturability margin was experimentally evaluated.

Acknowledgements

The authors would like to thank A. Goswami and E. Westervelt for their helpful comments.

M2V2 is a second generation version of the M2 robot, developed by G. Pratt, D. Paluska, J. Pratt, and D. Robinson at the MIT Leg Laboratory. M2V2 was designed by B. Krupp, V. Ragusila, I. Olaru, T. Craig, and J. Pratt. M2V2 electronics and interface software was designed by G. Watkins, S. Emami, T. Hutcheson, J. Pratt, J. Smith, and S. Nayak. M2V2 was assembled and maintained by T. Craig and J. Taylor. The force sensing push stick used in push recovery experiments on M2V2 was designed and constructed by J. Joyner.

A team at Bucknell University, led by S. Shooter and K. Buffinton, have designed improved feet and a vision head for the robot. Students involved include M. Kandler, D. Snyder, L. Markison, J. Ricci and C. Hubicki.

Various low-level control modules for M2V2 were developed by C. Shake, N. van Nieuwenhuizen, F. Cañas, D. Garg, S. Tamadoni, R. van Doesburgh, T. Koolen, J. Pratt, M. Johnson, and P. Neuhaus. A team from NASA JSC, consisting of J. Braman, D. Gooding, S. Tamblyn, M. Goza and A. Hulse, has also contributed in this area.

M2V2 filming and video editing was performed by W. Howell. Part acquisition and lab management was performed by B. Layton.

Simulation Software and user interfaces were improved by J. Carff, M. Fortenberry, D. Reyes Duran, G. Barr, B. Waxler, and P. DeMonaco.

Appendix

Robot parameters

Table 7.1 shows the joint layout and inertia parameters of M2V2. Each row represents a joint and its associated rigid body. We use a coordinate system in which x is forward, y is to the left, and z is up. Note that the bottom 6 rows represent the left leg joints and masses (marked with the letter 'L'). The right leg is a mirror image of the left leg, and thus is identical to the left leg except for the y values, which are

all the additive inverse. For Pin-type joints, the letter following “Pin” refers to the rotational axis that the joint is aligned with.

Table 7.1: Robot joint and mass layout.

Joint	Parent	Offset from Parent [m]			Joint Type	Mass [kg]	COM offset [m]			Moment of Inertia [kg m ²]		
		x	y	z			x	y	z	I_{xx}	I_{yy}	I_{zz}
Root	N/A	0.0	0.0	0.0	Floating	24.0	0.0	0.0	0.0	2.756	2.756	0.380
L hip yaw	root	0.0	0.092	-0.381	Pin-Z	0.2	0.0	0.0	0.0	5.3E-4	5.3E-4	5.3E-4
L hip roll	L hip yaw	0.0	0.0	0.0	Pin-X	0.0	0.0	0.0	0.0	0.0	0.0	0.0
L hip pitch	L hip roll	0.0	0.0	-0.029	Pin-Y	4.6	-0.0318	0.006	-0.229	0.044	0.044	0.004
L knee	L hip pitch	0.0	0.0	-0.450	Pin-Y	4.3	-0.025	0.0	-0.229	0.054	0.054	0.003
L ankle pitch	L knee	0.0	0.0	-0.467	Pin-Y	0.11	0.0	0.0	0.0	2.6E-4	2.6E-4	2.6E-4
L ankle roll	L ankle pitch	0.0	0.0	0.0	Pin-X	0.727	0.0127	0.0	-0.051	3.63E-4	0.002	0.002

8

Robot prototype: performance

8.1 Introduction

This chapter presents and evaluates the performance of the robot prototype TULip introduced in Chapter 6 with the implemented control algorithms described in Chapter 7. We tested the performance of the robot on various aspects that we consider important to achieve a robust gait. These aspects have been discussed separately in earlier chapters; a short overview is given below.

- center of mass control through control of the center of pressure location;
- control of the center of pressure location through force controllable joints;
- compliant control of the force controllable joints to achieve natural robustness to unknown terrain or external perturbations;
- joint control through high-level task-based control of essential tasks in bipedal locomotion (e.g., body height, orientation and speed, foot placement control, center of pressure control);
- adaptive, fast and accurate foot placement.

We tested the robot on these five aspects by conducting several experiments, which are two-legged balancing, single-legged balancing, foot placement and single-legged balancing with foot placement to recover from a push.

Results of walking experiments are not included in this chapter since we found that the foot placement performance is currently not sufficiently accurate and fast to achieve a robust walking gait. Possible solutions are discussed in Section 8.6 which evaluates the robot hardware and software.

8.2 Two-legged balancing

First, we tested the ability to control the center of pressure location and thereby the center of mass. The robot was commanded to remain balanced on two feet with slightly bent knees while being pushed from the side and back. The active controller was the stance-sub controller described in Section 7.7.5 which controlled the upper body height, orientation, and instantaneous capture point location through control of the center of pressure location (by controlling the virtual toe points). Results of this experiment are shown in Figure 8.1. The figure shows how the virtual toe points and support fractions are adjusted to achieve the desired center of pressure location and drive the instantaneous capture point back towards the desired instantaneous

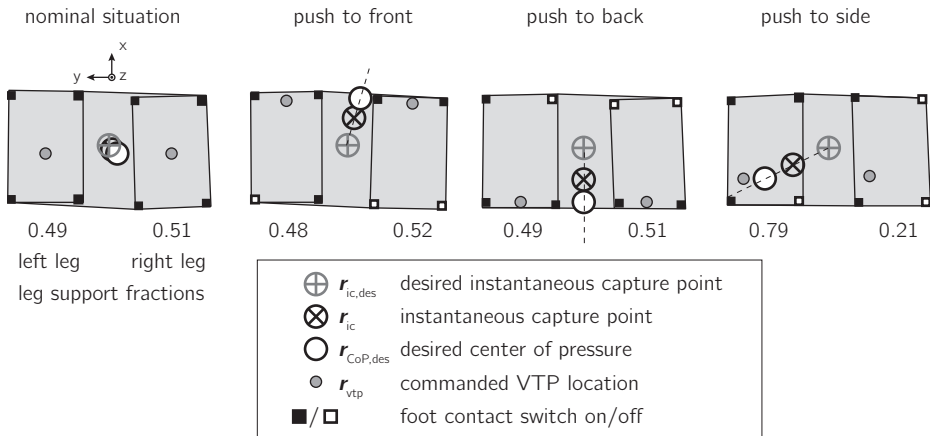


Figure 8.1: Ground projection of the robot's support polygon and state indicators during a balancing experiment. The robot's state just after a push is shown for three pushes in different directions.

capture point location after a push. Also, the figure nicely demonstrates that as the virtual toe points come close to an edge of the support polygon, the contact switches at the opposite side of the support polygon start losing contact due to small foot rotations. Though the exact accuracy of the center of pressure manipulation has not (yet) been determined, based on these results we estimate that the accuracy is about 10 mm.

Second, the compliant behavior and robustness of the stance-sub controller was tested. Again, the robot was commanded to keep an upright posture on two feet with slightly bent knees. External pushes were applied and one of the feet was placed on a moving platform. Images from the experiments are shown in Figure 8.2. The accompanying video is visible on the Delft Biorobotics Website (Delft Biorobotics Laboratory, 2011). The figure and video show that the control algorithms are insensitive to changes in the position or orientation of the feet: only the control of the contact forces is relevant.

8.3 Single-legged balancing

The robot was commanded to remain balanced on one leg. This allowed us to further test the capabilities of the stance sub-controller and test the gravity compensation that was applied to the swing leg. The gravity compensation controller applies a feed-forward torque to the swing-leg joints to compensate for gravitational forces. The compensating torques are determined based on an internal kinematic model

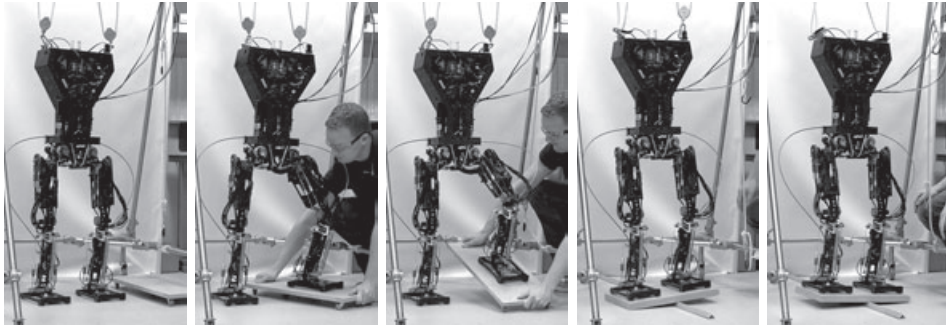


Figure 8.2: Images showing two-legged balance for different system configurations. The active controller (Chapter 7) is equal in all situations. The controller controls the location of the instantaneous capture point by controlling the center of pressure location through application of adequate torques at each joint. The images demonstrate that the algorithm is by design not sensitive to changes in foot location or orientation. Images are stills from the video visible on the Delft Biorobotics Laboratory website (Delft Biorobotics Laboratory, 2011). The video demonstrates the controller's robustness against external disturbances such as pushes or forced modifications to the base of support.

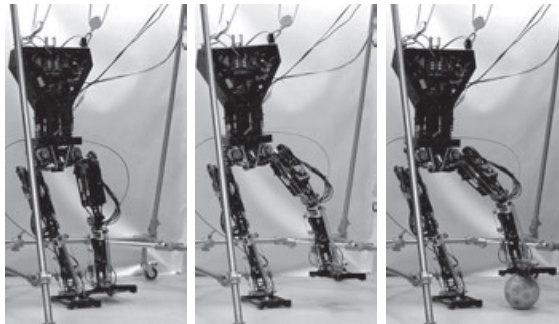


Figure 8.3: Images showing single legged balance for different system configurations. The stance sub-controller of Section 7.7.5 only controls the right leg instead of both legs in case of two-legged balancing (Figure 8.2). All weight of the robot is supported by the right leg, the joint torques in the left leg are set opposite to the gravitational torques. Images are stills from the video visible on the Delft Biorobotics Laboratory website (Delft Biorobotics Laboratory, 2011). The video demonstrates the controller's robustness against external disturbances such as pushes or forced modifications to the robot's state.

and the measured direction of the gravitational acceleration vector. The results are shown in Figure 8.3 and the accompanying video (see Delft Biorobotics Laboratory, 2011). The figure and video demonstrate how the robot remains balanced on one leg when the posture of the robot is changed. The gravity compensation worked sufficiently which validated the accuracy of the robot's internal kinematic model.

8.4 Foot placement

The attainable speed and accuracy of foot placement were determined.

First, the robot was commanded to swing the leg forward over a distance of 650 mm within 1 second. The swing leg trajectory was generated using a trajectory generator based on inverse kinematics. Inputs for the generator were three desired foot configurations: the foot position and orientation at the start of the step, at the end of the step and at one point in between. The foot position and orientation in between was selected in such a way that the toe would not scuff the floor in mid-swing. Outputs of the generator were the reference joint positions and joint velocities. These outputs were obtained by creating a smooth path in joint-space that would satisfy the three reference input configurations. The reference positions and velocities were tracked by digital PID position control in the joints (see Figure 6.3).

Kinematic data from a single step is shown in Figure 8.4 that illustrates the foot placement performance of TUlip in the forward direction. The tracking performance of the system was found to be poor: the error between the real foot position and desired foot position easily grew to about 150 mm along the trajectory. The final positional error was typically about 50 mm. The results demonstrate that the series elastic actuators proved incapable of accurately tracking the given reference trajectory. Section 8.6 discusses the causes of this problem and describes the research we have initiated to solve this problem.

Second, the robot was commanded to swing the leg to the side over a distance of 350 mm within 1 second. The resulting accuracy was about 15 mm (partly because the hip roll joint is not actuated through a series elastic actuator, contrary to the hip pitch joint). Gearbox play in the hip roll actuator did initially introduce an inaccuracy of about 40 mm in the lateral foot position, but this could be compensated for in software since the stepping motion was unidirectional.

The accuracy and repeatability of foot placement have to be improved, especially in the forward direction, before the robot can perform a robust walking gait. The lateral foot placement performance did prove to be sufficient to test the robot's

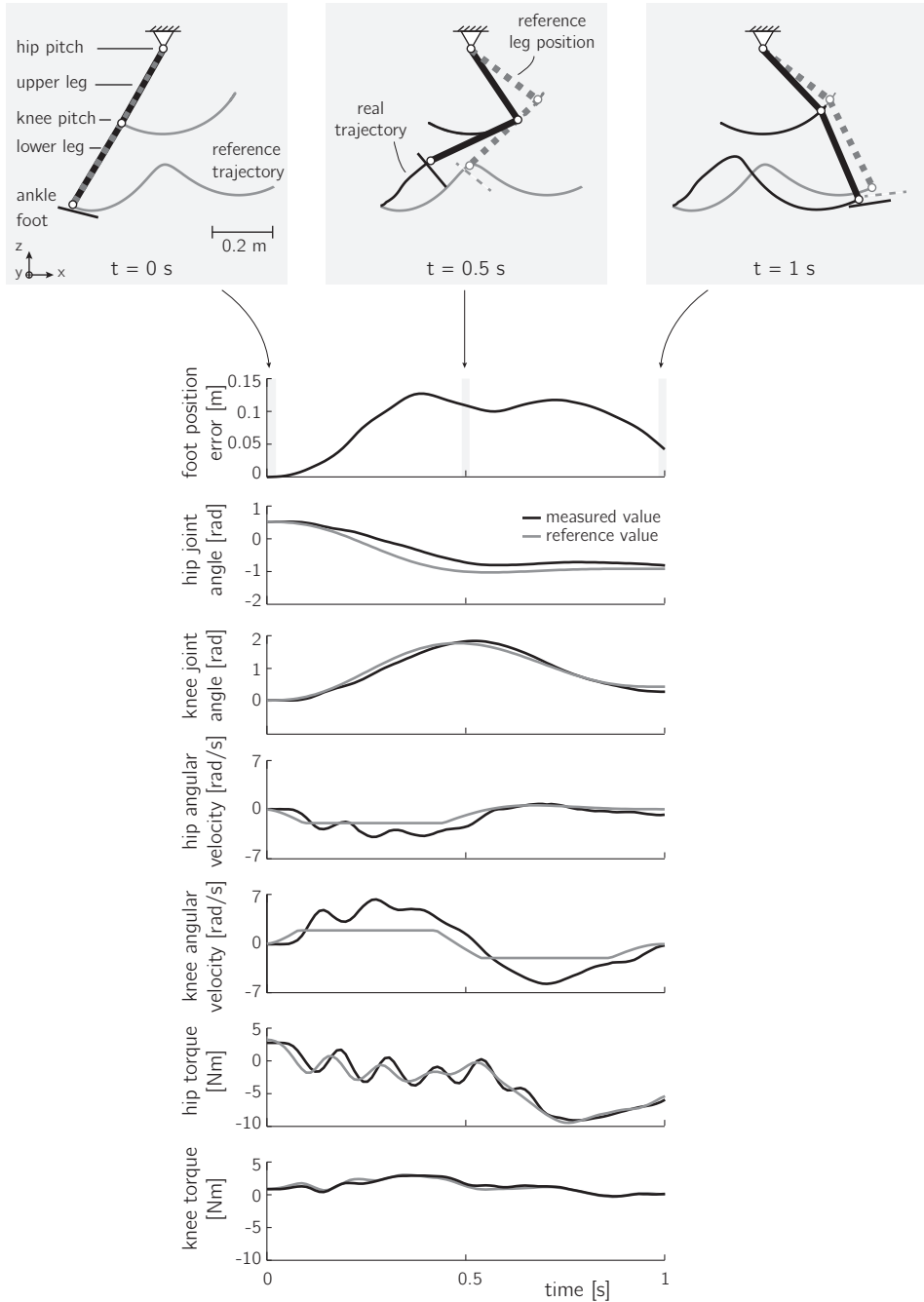


Figure 8.4: Kinematic data from a single leg swing in the forward direction.

performance in case of balance recovery through side-stepping, as described in the next section.

8.5 Push recovery by stepping

Figure 8.5 shows images from a video (see Delft Biorobotics Laboratory, 2011) that demonstrates the robot going from two-legged to single-legged balancing and performing a step to regain balance after a push. The push is applied in the lateral direction since foot placement in the fore-aft direction proved less accurate.

For different time instances, stills from the video are shown together with robot data. An overhead view of the support polygon is shown together with ground projections of points that give a representation of the robot's state (instantaneous capture point r_{ic} , center of mass r) and the control actions (desired instantaneous capture point $r_{ic,des}$, center of pressure r_{CoP} and the virtual toe points r_{vtp}). At t_0 the robot is pushed to the side. At the instant that the instantaneous capture point leaves the support polygon, a step is triggered to regain balance. The step time is set constant and the location of the step is continuously updated and a function of the instantaneous capture point. Seen from the stance foot, the desired foot placement location for the swing foot is offset about a foot's width beyond the estimated location of the instantaneous capture point at the predicted time of foot placement. The step was performed beyond the instantaneous capture point to make sure that both legs remained in contact with the ground and could therefore both contribute to regaining balance. As expected, stepping exactly onto the instantaneous capture point resulted in the full robot weight being transferred to the recovery leg. Regaining balance using only one leg proved too challenging because of lacking motor power in the hip roll joint.

8.6 Discussion

This section gives an overview of the lessons learned with respect to the Tulip research platform. In addition to scientifically interesting aspects, we also present practical information for future robotic projects. Furthermore, it gives an insight in what the daily reality is of trying to do research on a real robot prototype: doing lots of engineering to fix problems in the mechanical, electrical and software domain.

We will start by reviewing the robot's ability to control forces and perform accurate and fast foot placement, since we contend that these abilities are crucial to achieve human-like walking performance. This is followed by a more general evaluation of the robot's mechanics, electronics and software.

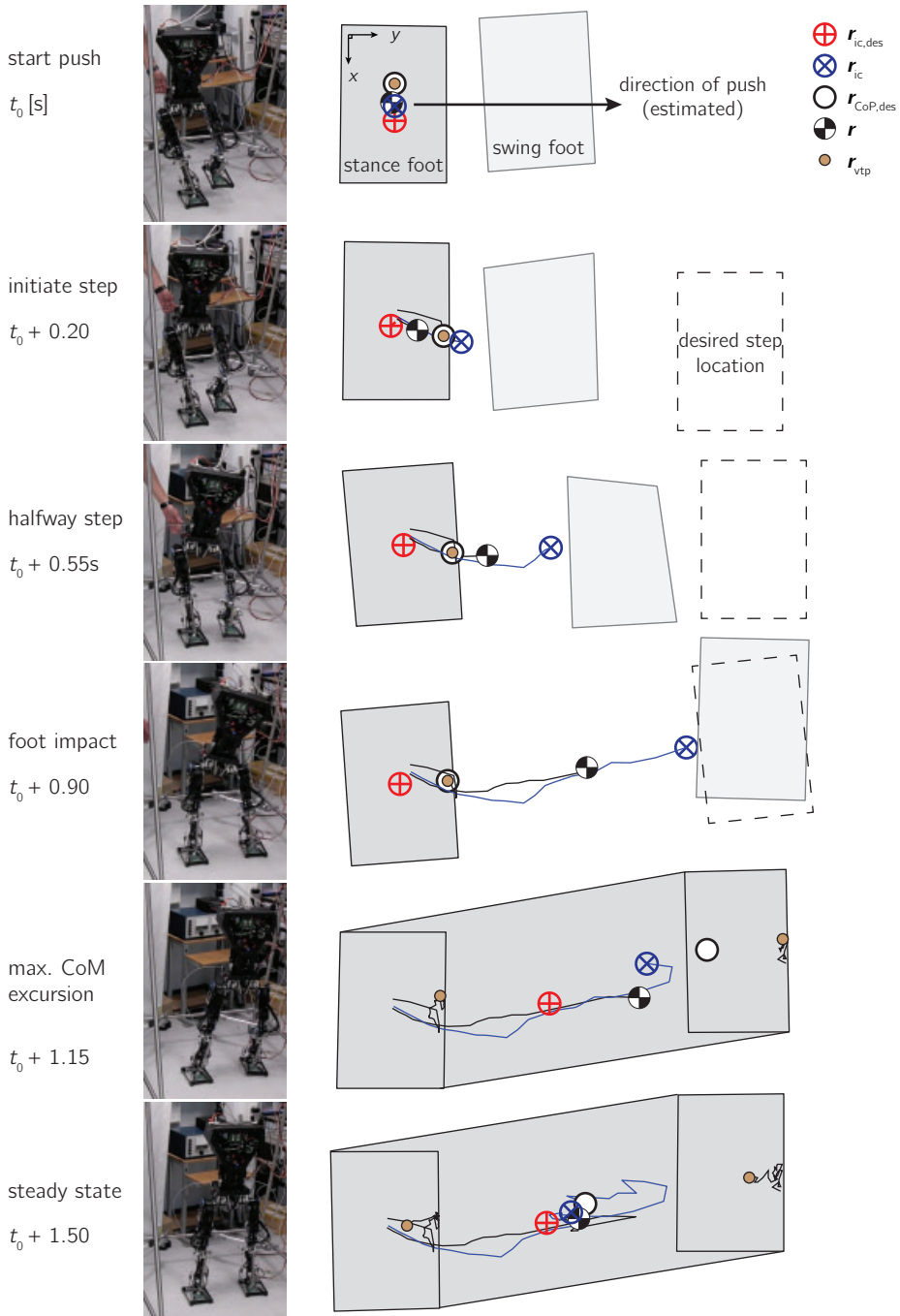


Figure 8.5: Example result of a step to regain balance after a push. The step is directed towards the predicted instantaneous capture point at the predicted time of foot impact.

8.6.1 Force controllability

The benefit of force based control with respect to robust balance control was demonstrated by the balancing experiments presented in this chapter. Force control offered the ability to control the contact forces between the foot and the ground, which made the system insensitive for the exact position and orientation of the foot. Also, the low impedance of the series elastic actuator made the system forgiving to external perturbations: the robot's posture could be easily changed by the operator without leading to a fall of the robot.

However, the intrinsically low impedance of the employed series elastic actuator did hamper fast and accurate trajectory tracking of joint positions. The resulting low force controllable bandwidth of the actuators resulted in poor tracking performance when using conventional high-gain feedback control. Increasing the tracking performance would require a well designed feedforward signal. For non repetitive motions, designing such a feedforward signal would require an accurate dynamic model (e.g., see Randazzo et al., 2011). In order not to depend too much on such potentially complex models, we believe that future robots are best served with series elastic actuators with a higher series stiffness. To compensate for the resulting loss in shock tolerance, a low-stiffness element could be placed in the foot. This configuration would result in an intrinsically high shock tolerance at the foot and moderate shock tolerance for impacts occurring elsewhere on the body. In the future, new concepts of variable impedance actuators might also be developed that have an acceptable power to weight ratio and thereby have added value over constant impedance actuators (van Ham et al., 2007; Hurst et al., 2004; Schiavi et al., 2008; Wolf and Hirzinger, 2008).

Furthermore, for future robots we see benefit in using a concept of series elastic actuation that requires no pre-loading force. The pre-loading force in the actuators of TULip affected the whole chain of components between the actuator and the joint. This resulted in friction and wear of the components. Though a concept can be used where the pre-load affects only the springs (e.g., see Pratt and Williamson, 1995), it is also possible to use an elastic element that requires no pre-load. For example, a compliant element with an angular stiffness such as a double spiral spring (Lagoda et al., 2010), torsion bar (Pratt and Williamson, 1995) or the motor gearhead itself (Neuhaus et al., 2011).

8.6.2 Foot placement

The robot's limited ability to perform fast and accurate foot placement should be solved in mechanics and in software.

Regarding the mechanics, besides the above-mentioned issues with the series elastic actuators, we found that gearbox play (about 1 degree for TULip's planetary gearheads) had a significant effect on the accuracy of foot placement. Different transmission concepts should be used on future robots that have less play, such as spindle drives or harmonic drives.

Regarding the software, a swing leg controller that generates swing leg trajectories based on inverse kinematics is simple to implement, but the generated joint trajectory can be far from optimal. We strive to use control methods that can generate a swing leg trajectory while taking system limitations into account (e.g., minimal foot clearance, maximum torque change). We aim not to optimize these control trajectories offline and simply play them back on the robot, but to optimize them online. We are investigating the use of model predictive control online, which also uses feedback of the real swing leg to account for modeling errors or disturbances.

8.6.3 Mechanics

The improvements made to the robot mechanics were mainly to improve the robustness, repeatability and controllability of the system.

Robustness

The robustness of the mechanical structure was improved. The strength of some of the components sometimes proved insufficient, as can be expected for a robot design where the focus has been on keeping the mechanical structure as light as possible. Especially in the case of unexpected and abrupt robot behaviors, components regularly broke or were damaged. A seemingly easy solution is of course to prevent these actions from occurring in the first place. Testing control algorithms in simulation or limiting output torques or angular velocities of course helps. However, robot behavior can remain hard to comprehend and non-intuitive as it involves many dimensions, especially in the case of control algorithms that control multiple joints. Moreover, issues with the robot's simulation model (see Section 8.6.5) made testing of new control algorithms hard. Therefore, most of the time, a slightly unsatisfactory and potentially deficient solution was chosen by replacing the broken part for a re-designed part and preventing the same destructive robot behavior from occurring again. Furthermore, more mechanical fuses and stops were added to the system to prevent damage. None of the newly designed parts lead to functional changes of the robot nor did it lead to new scientific insight. It does however illustrate how time consuming research on a real robot can be.

To improve the ease of servicing the robot and keeping it operational, we see great benefit from using more modular structural units in the robot. For example, modular

actuator units. TULip has many series elastic actuators which are all conceptually the same but mechanically different. This requires different control tools for each actuator and hampers quick repairs to the robot because of the many different components which are typically too costly to all keep on stock. A modular unit can solve these problems. Modular actuator modules at each joint can come at the cost of being less tailored to the specific needs of that joint and might result in a suboptimal robot performance. However, the exact needs for a specific joint are not always well defined, especially for a completely new robot prototype. Having thoroughly tested and well characterized actuator modules that are potentially slightly suboptimal can still be preferable. The experience is that the robot can only serve as a true research platform for walking experiments if the hardware is robust and easy to maintain. Otherwise, it will serve as a platform that can only be used to demonstrate a proof of concept, which is unfortunately the case for TULip and maybe even for most of the humanoid robot platforms that currently exist.

Repeatability

The repeatability of the robot behavior was initially insufficient and had to be improved: executing the same foot placement strategy one day or the next day could lead to a significant difference in the resulting robot behavior. The main cause of this was poor force-fidelity of the series elastic actuators, caused by friction in the drivetrain and the inadequate initialization of the actuators at the startup of the robot. These issues are discussed below.

Friction in a series elastic actuator is not necessarily harmful for its ability to accurately control joint torque: frictional losses between the geared motor and the elastic element are compensated because the true elements' deflection is fed back to the torque controller. Consequently, the force-fidelity of the geared motor can be low. However, due to the initially high pre-loading force in the series elastic system of TULip, viscous and coulomb friction was found between the elastic element and the joint. These frictional losses are not measured and therefore do affect the performance of the actuator. This issue was solved by lowering the pre-loading force on the system by replacing the tension springs with 'spring units' (Figure 6.7) and by replacing the sleeve bearings in the idler pulleys (Figure 6.6) with ball bearings.

The initialization of the robot comprises the calibration of the incremental joint and motor encoders. Especially the series elastic actuators rely heavily on accurate calibration of these encoders, since the difference between the measured joint angle and measured motor angle directly determines the measured joint torque. At the robot startup, there typically is a nonzero joint torque since the elastic elements typically have a nonzero deflection due to friction in the actuator (resulting in a poor

backdrivability of the actuator). Proper initialization should determine the joint angle and motor angle that correspond to a zero joint torque.

Initially, the robot was initialized using a dynamic process. The robot was hung upright and the series elastic actuated joints were swung to the front and to the back by applying an equal but opposite motor current in both directions. The joint encoders were calibrated when the index pulse was detected, which corresponded to a known joint angle. The motor encoders were calibrated based on the difference in the maximal measured joint angle in both directions. This difference was assumed to be caused by play in the gearbox and an initially present joint torque. Given the measured difference, an assumed zero position of the motor encoder was estimated. However, the repeatability of this calibration proved to be poor. Therefore, a mechanical calibration rig was constructed to solve this issue. The calibration rig fixes the robot joints at known angles. At the same time, we applied a sinusoidal signal to the motor of the series elastic actuators. Any initially present joint torque could be accurately detected since it resulted in a difference in the measured maximal motor angle in both directions.

Controllability

The controllability of the system was improved by adding ankle actuation in the ankle roll joint. Though the effect of lateral ankle actuation remains limited due to the small foot width that allows for relatively low maximum torques, creating versatile motions with a mechanically fixed ankle stiffness and equilibrium position proved impractical.

8.6.4 Electronics

Hardly any modifications have been made to the electronics. Since almost all the electronic components were high-end off-the-shelf components, only small problems arose with the electronics of TULip.

The only concern of the current electronic architecture is its weight. The electronic components contribute to about 32% of the total weight of TULip (Figure 6.14), which is very significant. About 40% of the weight of the electronics consists of cables running from the central computer to all the sensors and actuators. With the recent advancements in the development of small and powerful motor control boards, distributed control for future humanoid robots seems preferable. Their ability to do sensor processing locally reduces the amount of cables significantly. This reduction is even larger for controller boards that can operate in a daisy chain. Also, their ability to do motor control alleviates the central computer from this control tasks.

8.6.5 Software

Besides the large effort that was put into the development of new control software, a lot of effort was also put into the development of tools to visualize, log and tune robot parameters. These tools enable the analysis of experimental data offline or modify robot behavior on the fly, which proved necessary to design advanced motion controllers. Though the added value of these tools is obvious, development of these tools proved less obvious since it required communication from the real-time software through the non-real-time software to an external computer.

The overall software architecture functioned adequately, mainly because the whole control software architecture was custom-made and could therefore be made to fit our needs. On the one hand, we were satisfied with our custom-made software since it offered great flexibility and therefore was a good match for a research project. On the other hand, the lack of skilled people to develop all the custom-made software significantly slowed down the research progress. This practical issue should be taken into account in future robotic projects, especially for robotic research on a university, where the throughput of people is typically high and the experience typically low. Using control software such as The Mathworks MATLAB or National Instruments LabVIEW might be better suited in that case. Conversely, the community of people that create and share custom made robotic applications has grown rapidly over the past years with a good example being the Robot Operating System ROS (ROS, 2011).

The lack of manpower also resulted in simulation software that did not function as intended. Around 2008, a first simulation model of the robot was realized that could be controlled through TULipMC. However, the computation time was relatively long: in the order of 20 minutes for a few steps. This limited the practical use of the simulation model as a tool to quickly test and debug control software. Unfortunately, we lacked manpower to find the causes and solutions of this long simulation time. Consequently, we mostly used the real robot as a test platform.

8.7 Acknowledgements

The author would like to thank the many students and staff working within the DutchRobotics foundation (DutchRobotics, 2011) for their contribution to the development of TULip and thank Jerry Pratt from IHMC for helpful tips and code sharing with respect to the implementation of Virtual Model Control (Pratt et al., 2001) on TULip.

9

Discussion, conclusions and future directions

9.1 Introduction

The goal of this thesis was stated as follows.

The goal of this thesis is to improve the performance of robotic bipedal locomotion in terms of robustness, versatility and energy-efficiency, by increasing the understanding of the mechanics and control of foot placement. Since versatility and energy-efficiency are only relevant when the robot can prevent a fall, a strong focus lies on the robustness of gait.

To achieve this goal, three specific research questions were formulated concerning foot placement and robustness, versatility and energy-efficiency respectively:

1. *Regarding robustness: how can we determine where and when the foot should be placed to prevent a fall?*
2. *Regarding versatility: how can we determine where and when the foot should be placed to enable the system to evolve from its initial state to any desired future state?*
3. *Regarding energy-efficiency: how do actuator limitations influence foot placement strategies?*

This chapter starts with a section that summarizes the main findings on foot placement with respect to robustness, versatility and energy-efficiency. This is followed by a more general discussion on the role of foot placement in bipedal locomotion. Finally, we state the main conclusions of this research and give directions for future research.

9.2 Recapitulation

9.2.1 Foot placement and robustness

The most generic and unrestrictive definition of stability for a legged system is 'to avoid falling', described by the concept of viability (Aubin, 1991; Wieber, 2002). However, establishing which states are viable is generally computationally expensive, which causes this concept to be of limited practical value for controller synthesis for bipedal locomotion. Therefore, this thesis introduced the theoretical framework of capturability (Chapter 3). Capturability was shown to effectively approximate viability, to be computable for simple models and to be valuable for the analysis and control of bipedal locomotion. Characteristic to the capturability framework is that it analyzes fall avoidance by considering N -step capturability: the system's ability

to eventually come to a stop without falling by taking N or fewer steps, given its dynamics and actuation limits. The number of steps required to come to a stop is used as a measure of how close the legged system is to falling.

Computing N -step capturability for a simple model of a complex bipedal system provided valuable insight for the analysis and the control of the complex system. We used the 3D Linear Inverted Pendulum Model, which models the most essential dynamics of a bipedal system: the dynamics of the CoM with respect to the CoP. Extending the model with a foot and reaction mass allowed us to derive how ankle torques and upper body motions affect the CoM dynamics and contribute to N -step capturability. N -step capture regions were constructed on the ground that provided practical information on where and when to place the foot to prevent a fall. The size of these regions gives a measure of capturability of the system's state and is a useful robustness metric, termed the N -step capturability margin. The introduced d_∞ capturability level provides a state-independent measure of the capturability of the system.

The usefulness of the capturability framework was demonstrated by its successful integration in the control algorithm of two complex bipedal robots (Chapter 7 - 8).

9.2.2 Foot placement and versatility

The previous chapters presented how balance strategies can influence the CoM motion. These balance strategies can not only be used to prevent a fall, but also for achieving a wide variety of gaits. The capturability framework allowed the construction of N -step capture regions that indicated feasible stepping regions. The effect of a candidate CoP location within these regions could be directly derived by looking at its location relative to the instantaneous capture point location: the CoM would asymptotically converge to the line spanned by the CoP location and the instantaneous capture point location. The distance between the CoP location and the instantaneous capture point determined at what rate the instantaneous capture point moved away from the CoP location. Chapter 4 presented a foot placement controller that explicitly derived foot placement strategies given a desired future state. These findings provided simple and useful control heuristics for the control of robotic bipedal locomotion. Chapter 8 presented robotic single-legged balancing, two-legged balancing, and push recovery. It was also demonstrated how the derived foot placement strategies can lead to more versatile behavior such as walking while changing walking speed, changing walking direction and navigating over stepping stones.

9.2.3 Foot placement and energy-efficiency

Previous chapters studied how system limitations in general can affect foot placement and thereby balance. The capturability analysis of Chapter 3 demonstrated how system constraints such as maximal step length and minimal step time affect the capturability of a system. Since balance control is dictated by foot placement, such constraints determine to a large extent the system's capturability. Chapter 5 analyzed in more detail how actuator limitations can influence the execution and planning of foot placement. We studied the observed relation in human stumble recovery between the employed stepping strategy and the timing of the stumble in the gait phase. We found that simple gait models demonstrate human-like foot placement strategies in response to a stumble when optimizing for either one of the following cost measures for foot placement: peak torque, power, impulse, and torque divided by time.

9.3 General discussion

This general discussion addresses how foot placement control in robotic bipedal locomotion relates to: (1) the objective of an overall increase of robustness, versatility and energy-efficiency. (2) the control of the other balancing strategies, i.e., ankle torques and upper-body motions. Both issues are discussed by the evaluation of four fundamental properties of foot placement that determine the actual role of foot placement in bipedal locomotion: its effectiveness, availability, energetic cost and accuracy.

9.3.1 On the effectiveness of foot placement

With the application of the capturability framework to the simple gait models presented in Chapter 3, a quantification of the effectiveness of the stepping strategy, ankle strategy and hip strategy is possible. The effectiveness of a strategy can be analyzed based on the amount of control that it has over the CoM motion, represented by the instantaneous capture point. This amount can be quantified in terms of its contribution to the d_∞ capturability level of a system or the N -step capturability of a given system state. In Section 3.8 it was approximated that foot placement determines for at least 65% the d_∞ capturability level of a human (versus 30% for the ankle strategy and 5% for the hip strategy). The analysis can be applied to a wide range of legged systems and legged locomotion scenarios. For example, consider the case of one-legged balancing given in the introduction of this thesis (Figure 1.4, page 6). To be capturable using only the ankle strategy, the instantaneous capture point is maximally at the edges of the base of support, where the CoP can just be placed. For a human, the CoP location can be adjusted over approximately 45 mm in the frontal

plane (Hof et al., 2007; Hoogvliet et al., 1997; King and Zatsiorsky, 2002) and over 200 mm in the fore-aft direction (Winter, 1990). Using only the hip strategy, the gait model with reaction mass predicts that violent upper-body motions can only ensure capturability if the instantaneous capture point lies not more than approximately 40 mm outside the base of support. This corresponds well with observations from human balance studies (Hof, 2007; Horak and Nashner, 1986; Otten, 1999). Using only the stepping strategy, the instantaneous capture point is maximally a step size away from the stance ankle to become captured in one step, which is typically about 700 mm for a human. As expected, these results confirm that foot placement is the most powerful balancing strategy for control of the CoM motion.

9.3.2 On the availability of foot placement

Since foot placement control has the largest control over the CoM motion, balance control in the presence of large disturbances clearly relies mostly on foot placement. Though foot placement is powerful as a control mechanism, it is conversely only available at the occurrence of the next step. Thus, as the walking speed goes to zero and step time to infinity, balance control relies solely on the ankle and hip strategies. What does this mean for the control of walking? How should the three strategies interact during the control of unperturbed walking at a comfortable walking speed?

When relying only on foot placement for balance control, the stance phase becomes a purely ballistic phase. Any inaccuracy in foot placement results in a deviation from the desired CoM position and velocity at the end of the stance phase. The 3D-LIPM plus foot placement controller derived in Chapter 4 allows for predictions on how deviations from the desired CoM motion should be corrected for by the next step. For a typical human walking gait (step time 0.55 s, step length 700 mm, step width 120 mm and the reciprocal of the characteristic frequency $\omega_0 = 3 \text{ rad s}^{-1}$ (Donelan et al., 2002)), a lateral inward offset from the nominal step location requires the magnitude of the next step to be a factor of 3 larger in the outward direction to arrive back at the desired nominal CoM motion. This demonstrates how small errors at the beginning of the stance phase are magnified at the end of the stance phase.

However, this does not necessarily lead to instability, as long as foot placement is adjusted based on the deviated CoM position and velocity. The CoM may diverge quickly from the nominal motion, but a single well placed step can instantaneously reduce or eliminate the CoM deviation. In this way, each step corrects the deviation of the CoM motion that was introduced by the previous step. The capturability analysis can be used to get an indication of an acceptable step inaccuracy. For the above mentioned human walking gait (assuming a constant step time and no ankle

or hip torques), the 3D-LIPM model predicts that the nominal motion is 1-step capturable and that a step location inaccuracy of more than 35 mm can cause the system to become 2-step capturable. The foot placement controller of Chapter 4 can still cope with such step inaccuracies by adjusting the next step location (with about 100 mm). The resulting gait then does resemble a 'drunk' walk with large variations in the average forward and lateral velocity between steps. Instead of relying solely on foot placement to correct CoM motion deviations, the ankle strategy can be used. The continuous availability of the ankle strategy allows for continuous corrections to the CoM motion, as was shown graphically in Figure 3.8. Foot placement can provide gross control and ankle torques a fine tuning, as was found for human walking (Hof et al., 2007, 2010). Due to the smaller foot width than length, the amount of control that the ankle strategy has over the CoM is less in the lateral direction than the fore-aft direction. This implies that for lateral balance step accuracy is more important than for sagittal balance. This may explain that the absolute lateral step variability in human walking is larger than the fore-aft step variability (28 ± 3 mm versus 16 ± 3 mm, Bauby and Kuo, 2000).

The relatively low amount of control over the CoM of the hip strategy makes this strategy a less attractive control strategy in normal walking. Humans tend to use this strategy only in case of balancing tasks with limited footholds or in case of severe perturbations such as a stumble (Otten, 1999; Pijnappels et al., 2010; Popovic et al., 2005).

These results suggest that walking control only through foot placement is possible, but will result in a larger (than necessary) step variability to compensate for step inaccuracies. Foot placement control is preferably accompanied by the continuously available ankle strategy to fine-tune the CoM motion. The hip strategy can be used for the same purpose.

9.3.3 On the energetic cost of foot placement

To achieve an energy-efficient gait, the energetic cost of the stepping strategy, ankle strategy and hip strategy should be taken into account. Kuo (1999) studied the work-related energetic cost to actively stabilize the lateral motion of a passive walking model using these three strategies. When the model is perturbed, the energetic cost of stabilization through ankle torques or upper body motions increases quadratically with the perturbation magnitude. The energetic cost of lateral foot placement adjustment was found to increase linearly with the perturbation magnitude. Adjustments in foot placement requires only minor changes to the swing-leg motion while having a large effect on the CoM motion. This result suggests that foot placement control is not only preferable over the other control strategies because of its high

effectiveness, but also because of its lower energetic cost. Also, these results suggest that the hip strategy has not only the lowest level of control over the CoM compared to the ankle and stepping strategy, but also a relatively high energetic cost. This indicates that the hip strategy is best used only in critical balancing situations when all balancing mechanisms are required to prevent a fall.

Note that in this thesis, we have considered foot placement as the result of the desire to progress while preventing a fall, rather than to progress while consuming as little energy as possible. The latter interpretation is for example typically adopted in the Limit Cycle Walking approach, where foot placement is the result of mostly passive system dynamics. An overly strong focus on low energy consumption may draw attention away from the most important task in walking: the task of preventing a fall. We believe that it is necessary to first understand the fundamental principles of this task before focusing on how to execute it efficiently. With the capturability analysis, we can get an overview of all control strategies that will not lead to fall. Within this set of feasible control strategies, we can search for strategies that are optimal in terms of energy-efficiency or those that lead to more versatile robot behavior (as illustrated in Figure 9.1).

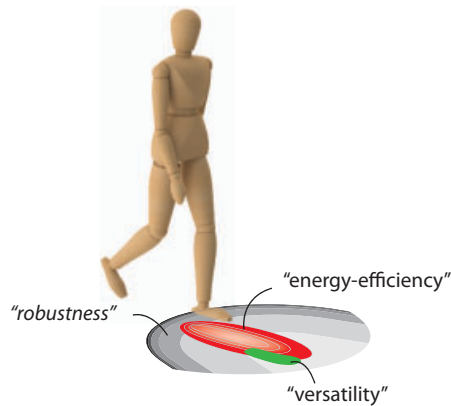


Figure 9.1: Conceptual illustration of the relation between foot placement strategies for which a fall can be prevented (i.e., 'robustness'), those that lead to a low energetic cost (i.e., 'energy-efficiency'), and those that can lead to a specific desired gait (i.e., 'versatility').

9.3.4 On the accuracy of foot placement

The accuracy of foot placement determines to a large extent how well the realized CoM motion corresponds with the desirable CoM motion. The accuracy of lateral foot placement is most critical, because the ability to correct lateral CoM deviations

through the use of the ankle strategy is very limited. To achieve a human-like lateral step variability (28 mm on average) in robotic walking, we estimate that the CoP location should be controlled at an accuracy of at least 9 mm (based on the above-mentioned 'error amplification factor' of 3). The initial foot placement location (described by the 'contact reference point' page 28) determines the possible CoP locations that can be achieved and can therefore not be too far off from the desired CoP location. It can be maximally located at a distance that is equal to the largest distance between the contact reference point and the edge of the base of support. However, this configuration severely constrains the amount of CoP modulation that can be achieved through the ankle strategy, since the CoP is located at the border of the base of support. A more preferable situation would be to place the contact reference point at a distance from the desired CoP that is about a quarter of the width of the base of support. This allows for CoP modulation in all directions through the use of ankle torques, e.g., in case of small perturbations. For a human morphology, this would correspond to a foot placement accuracy of about 10 mm.

But how to achieve that the foot (or CoP) arrives there and then, at the right location and at the right time? And, what *is* the right location and the right time? These two questions reflect the essence of the control of walking: the adequate planning and execution of foot placement strategies. These two aspects of foot placement can not be analyzed separately from each other, but are related. The execution of foot placement is for example constrained by system limitations (e.g., actuator peak torque, kinematic range) that should be taken into account when planning adequate foot placement strategies. Thus, adequate planning and execution of foot placement can be the result of a complex relation between various factors, e.g., the available level of actuation, sensing, state estimation and modeling. The following paragraphs will describe what insight this thesis has given us in all the factors that influence the overall foot placement accuracy.

On the accurate planning of foot placement

This section discusses issues related to the modeling and sensing that contribute to the overall accuracy of foot placement.

Modeling

We have derived foot placement strategies from low-dimensional linear gait models. How good is this model? We contend that for normal walking, balance control *is* low-dimensional, and sufficiently accurately modeled by the simple gait model because of several reasons. Firstly, balance can be well analyzed by focusing only on the CoM dynamics with respect to the CoP location. Secondly, modeling the system as a

linear inverted pendulum is acceptable since the angular excursions of the pendulum are small (< 20 degrees) and the true vertical displacement of the CoM in normal human walking is typically only 35 mm (Gard et al., 2004). Finally, the model was shown as a good predictor of human foot placement locations: Hof (2008) found that for lateral balance the CoP location at initial foot contact is a fixed offset from the instantaneous capture point location with a standard deviation of only 7 mm. A very similar analysis on sagittal balance found a standard deviation of 17 mm in the fore-aft direction (Millard et al., 2009). Also, a benefit of the low-dimensional model is that it can be continuously updated on the robot such that the planned foot placement strategy smoothly adapts to the state of the system.

Nevertheless, using a simple model to predict foot placement locations may result in a CoM motion that deviates from the desired or predicted path. Rebula et al. (2007) studied how well the instantaneous capture point approximated by the 3D-Linear Inverted Pendulum Model corresponds to the actual instantaneous capture point of a complex robot model. The modeled robot is commanded to step towards the approximated instantaneous capture point and it is analyzed if the robot reaches a captured state. If this is not the case, the desired step location is changed based on a learning algorithm. The reported offsets between the desired step locations based on the 3D-LIPM approximations and that on learning are in the order of 30 mm. However, it is not clear if these offsets compensate for errors introduced by the use of the simple 3D-LIPM. The reported offsets may also compensate for the errors introduced by the implemented controller that executes foot placement. This study again illustrates that planning and execution of foot placement are closely related.

We expect that in the case of moderate walking speeds, modeling errors introduced by the 3D-LIPM can easily be compensated for by ankle torques or by adaptation of the next foot placement. This situation may change for balance control when the system is 2 or 3-step capturable, for example due to a large perturbation. In this case, rapid actions are required to prevent a fall and system limitations consequently start to play a role. Control based on linear models may not be optimal or does not have the intended effect. However, one could reason that optimality and predictability are not that important in more extreme situations. Simple reflexive control strategies (such as 'step as fast as you can in the direction of the approximated instantaneous capture point') may not be optimal but still effective in preventing a fall. Similarly, the initial response to a perturbation in human walking also seems of a simple reflexive nature and was shown to be adapted and tailored to the specific perturbation when it was applied multiple times (Marigold and Patla, 2002; Nieuwenhuijzen et al., 2000). Ideally, such learning mechanisms are also incorporated in future balance control mechanisms.

Another relevant issue to discuss is the existence of two apparently conflicting findings regarding the adequate planning of foot placement strategies. In Chapter 4 of this thesis, we derived explicit foot placement strategies that allow a system to evolve from its current state to a desired future state. This desired state was assumed to be four-dimensional: the desired CoM position and velocity with respect to the CoP location in both the sagittal and the lateral plane were considered. Then, an adequate step time and an adequate step location were derived that would have the correct effect on all of these four dimensions. However, studies on foot placement in human walking found that the control of foot placement can be decomposed in two two-dimensional problems (Bauby and Kuo, 2000; Kuo, 1999; Townsend, 1985). The lateral and forward dynamics were found to be largely decoupled. This decoupled control approach was also successfully implemented on simulated and real bipeds (Hobbelen and Wisse, 2009; Kajita et al., 2001; Pratt and Pratt, 1999). How can these apparently conflicting findings be explained?

The decoupled view on foot placement control can be seen as a special case of what really is a coupled 4D problem. A decoupled approach is only possible for a constant step time. For that special case, a step width and step length can be independently selected which eventually will result in the desired state. A candidate approach is stepping at a constant offset away from the instantaneous capture point, where the system state converges to the desired state as the number of steps goes to infinity (see Chapter 4). When releasing the constraint of a constant step time, stepping strategies can be found that converge to the desired state within a fixed number of steps.

We can imagine that, for normal human or robotic walking, sustaining a constant step time can be beneficial. This does not require enforcing motions on the swing leg that are potentially far off from the passive leg dynamics. The level of control can be low and mostly equal for each step. Only in case of severe perturbations during walking do humans seem to adjust their nominal step time (van Dieën et al., 2007; Forner Cordero et al., 2003; Hof et al., 2010). It would be interesting to study how humans select their step time and step location relative to the instantaneous capture point and CoP location during more versatile behavior, such as navigating through an indoor environment. To our knowledge, such a study has not yet been conducted.

Sensing

Related to the accurate planning of foot placement is the required level of sensing that is necessary for accurate state estimation of the robot and its internal model. The accuracy of the foot placement relies on the accuracy of the estimate of the CoM position and velocity. Also, it requires estimations on where the CoP is underneath

the stance foot or where the swing foot is relative to the ground. We contend that the resolution, accuracy or update rate of modern sensors is not hampering robotic performance. Angle sensors easily achieve resolutions of $2 \cdot 10^{-4}$ rad ($1 \cdot 10^{-2}$ degrees) and inertial measurement units have an angular accuracy of about $2 \cdot 10^{-2}$ rad (1 degree) at an update rate of 250 Hz. However, there are two issues regarding sensing that require attention.

Firstly, the kinematic chain from the stance foot to the swing foot is long, easily consisting of 12 degrees of freedom. Any unmeasured elasticity or play in one of the degrees of freedom can cause significant deviations in the estimate of the state of the swing foot relative to the stance foot. This proved to be really problematic for the control of TULip. Gearbox play and compliance of structural components in the hip resulted in an inaccuracy of approximately 40 mm in the estimation of the lateral position of both feet.

Secondly, state estimation of the robot state with respect to the environment is not straightforward. To estimate the CoM velocity with respect to the ground, combined information is required from kinematic data and accelerometer data (to prevent errors due to a slipping stance foot or an accumulating integration error of accelerometer data). Often, we have used intuition to find filter parameters without having good knowledge on how well the measured CoM velocity approximates the true CoM velocity. These issues require more attention in future research.

On the accurate execution of foot placement

Accurate and fast swing leg motions are essential for the ability to handle large disturbances. The question is how to achieve this for a bipedal robot?

Since the execution of foot placement seems to be all about accurately controlling the position of the swing foot, a logical choice would be to use stiff position controlled joints and use high feedback control. However, position controlled joints are typically not shock tolerant due to their high impedance. Consequently, structural damage may occur. Also, controlling the forces that act on the CoM throughout the stance phase is not straightforward with position controlled joints. Therefore, we see benefit in using series elasticity and force control. Not only can it be beneficial during the stance phase. During the swing phase, compliant series elasticity can decouple the actuator dynamics from the structural leg dynamics, which offers simple modeling and easy control of the leg as if a perfect torque source with controllable stiffness is connected to the joint. Also, controlling forces is the most elementary way to control motion and offers optimization of the motion in terms of energy-efficiency.

However, for fast changing motions, compliant series elastic actuators are far from a perfect torque source with controllable stiffness. Fast motions require large torques, which require large deflections of the compliant element, which subsequently require large actuator motions. The result is that actuator limitations are quickly reached. Consequently, compliant series elasticity (like implemented on TULip) offers high-fidelity force control for moderate bandwidth but hampers fast and accurate trajectory tracking of joint positions.

A straightforward solution would be to increase the power of the motors to be able to enforce desired swing leg motions without reaching actuator limitations. This does obviously come at the cost of added weight and more energy use. However, fast and accurate swing leg motion does not necessarily require high power motors and high feedback gains. More 'intelligent' control can compensate for the lack of motor power. The swing leg control could be based on an accurate dynamical model of the swing leg (including actuators, compliant elements, transmissions) which enables the addition of feedforward control that can significantly improve the tracking performance (e.g., see Park et al., 2011). High levels of feedforward control in foot placement execution is perhaps most similar to the human case, where limited muscle power and muscle activation latencies do not allow for high-speed motion adaptations. Fast compensatory stepping reactions in humans were found only to be altered during the preparational and initial phase of the swing phase and immutable during the rest of the swing phase (Tripp et al., 2004). Humans demonstrate that despite actuator latencies, accurate execution foot placement can still be achieved (standard deviation of 6 mm, Hof et al., 2010).

We see most potential in series elastic actuators with increased series stiffness, sacrificing force control accuracy in favor of force control bandwidth. The increased intrinsic impedance (i.e., the impedance at a frequency above the controller bandwidth) may decrease the shock tolerance, but this could be solved by adding a more compliant series element between the ankle joint and the floor. It should be noted that advancements are also made in the field of robotic actuators with variable intrinsic impedance (e.g., Eiberger et al. (2010) and references therein), but their current low power to weight ratio of these actuators is limiting their application in bipedal robots.

The above-mentioned results illustrate the many factors that influence the accuracy of the planning of foot placement and/or the accuracy of the execution of foot placement and thereby determine the overall foot placement accuracy. This thesis described how the capturability framework can be used for accurate planning of foot placement. Accurate execution of these strategies is required to achieve robust

robotic bipedal gait. This was already demonstrated on a simulated robot but it remains a challenge to achieve this on the real robot prototype TULip.

9.4 Conclusions

This section presents the main conclusions regarding the posed research question and more general conclusions.

9.4.1 Research questions

- Regarding robustness: how can we determine where and when the foot should be placed to prevent a fall?

The timing and location of a step which can prevent a fall can be derived by analyzing the system's ability to come to a stop without falling by taking N or fewer steps. Simple models were used to approximate this ability. These models considered the effect of balance mechanisms on the center of mass position and velocity. This means that they considered the effect of ankle torques, centroidal angular momentum and foot placement on the instantaneous capture point. The resulting approximations of these simple models were shown to be sufficiently accurate for balance control of a complex bipedal robot.

- Regarding versatility: how can we determine where and when the foot should be placed to enable the system to evolve from its initial state to any desired future state?

The timing and location of a step which allow a system to evolve to a desired future state can be derived by using a dimensionally-reduced model of the system. We used a simple model that considered only the most fundamental dynamics of walking: the CoM dynamics with respect to the CoP. Because of this low-dimensional view, we could explicitly calculate the required step locations and step times for the system to evolve to the desired future state.

- Regarding energy-efficiency: how do actuator limitations influence foot placement strategies?

Firstly, actuator limitations influence foot placement by directly determining the achievable minimum step time. The minimum step time and maximum step length determine to a large extent the system's ability to prevent a fall. Secondly, simple gait models predict that actuator limitations influence human foot placement strategy selection, and not only foot placement strategy execution. Shortening of a

perturbed step is suggested to be energetically advantageous compared to the elongation of the perturbed step in case the perturbation occurs after mid-swing. The opposite is true for a perturbation that occurs before mid-swing.

9.4.2 General conclusions

The capturability framework provides a unified and practical approach to the analysis and control of robust bipedal locomotion because:

- a. The capturability analysis considers the relevant subset of the viability kernel. In other words, when analyzing fall avoidance by considering the ability to come to a stop in given number of steps without falling, all plausible and common states of bipedal locomotion are considered.
- b. The capturability analysis is theoretically not limited to a particular legged system, a controller, a specific motion, a local stability indicator, or an environment.
- c. N -step capturable states can be projected onto the ground to create N -step capture regions, which
 - conveniently visualize capturability;
 - provide practical information for controller synthesis;
 - lead to the N -step capturability margin, which quantifies capturability and provides a measure of robustness.

A low-dimensional and decoupled view on the control of walking offers:

- a. The use of simple gait models that only take the dynamics of the center of mass with respect to the contact forces between the foot and the ground into account.
- b. The possibility to approximate the whole-body dynamics of a complex system. Approximations reveal that the system's instantaneous capture point diverges away from the centroidal moment pivot point (or center of pressure location in case of zero centroidal angular momentum). It moves away along a straight line at a velocity proportional to the distance to the centroidal moment pivot point.
- c. Approximations that are shown useful for the synthesis of control strategies for a complex bipedal robot in case of
 - single-legged balancing during mild postural perturbations;
 - single-legged balancing with compensatory stepping in case of large perturbations;
 - two-legged balancing while the robot is being pushed and one of the feet is placed on a moving platform;

- walking in simulation.

9.5 Future directions

Previous chapters have presented new insight in the mechanics and control of bipedal locomotion. This section discusses future directions in this field of research.

9.5.1 On the mechanics of walking

TUlip was tailored for Limit Cycle Walking: series elastic actuators with low power and high compliance were used in combination with low levels of control. This combination of mechanics and control may be well-suited for periodic walking gaits at moderate speed but it turned out that this combination was less suitable for execution of aperiodic motions. Unfortunately, committing to a specific control methodology in bipedal robot research typically means committing to a specific hardware architecture and vice versa.

In order to properly study the mechanics and control of human-like bipedal walking, there is a strong need for a more versatile bipedal robotic research platform. A robotic platform that can not just be exclusively used for a single proof of concept, but one that can be used for research on a large variety of control ideas. The robot should be robust and easy to maintain and should therefore be able to operate over multiple trials. We see the following possibilities to achieve this.

The research platform should (initially) not be made autonomous. Of course, trying to match the human functionality in an autonomous human-like machine is an interesting challenge (e.g., illustrated by the RoboCup competition) and makes one appreciate the human body. However, to truly advance in the field of research on bipedal locomotion, the system's autonomy constrains it in too many ways. It should be tethered so that it can use off-board computing and off-board power supply.

Off-board computing does not only save weight but mostly offers more computing power, which is anticipated to speed up the research progress. It means not having to deal with a lack of memory, processing power or supported software.

Off-board power also saves weight and offers the implementation of more powerful actuators. We believe that a surplus of actuator power is essential for a future robotic prototype. Though humans perfectly well demonstrate that high power actuators are not necessary for walking, we believe that powerful actuators will quickly lead to more understanding of human walking. Taking away actuator limitations allows us to 'cheat': we can design walking controllers that are far from optimal or human-like but at least result in a stable gait. From this point on, the control approach can be

optimized to arrive at a more human-like gait with only low levels of control. This is different from an approach with low-power actuators that immediately requires complex control algorithms to get the robot operational at all.

Compare our proposed approach to the approach one adopts to learn to ride a bicycle without using the handle bars. One learns how to ride a bicycle as a child with the help of training wheels. Training wheels are used to prevent a fall, which allows one to simplify biking ('cheat'). This way the focus lies on the control of the pedals and steering. The support of the training wheels can be gradually reduced after training. Once the control is sufficiently mastered, complex upper-body motions can be learned to control the bicycle without use of the hands. Trying to achieve the same skill without such a gradual learning process, but by trial and error, is probably the slowest and most frustrating path to success.

So, we see great benefit from a robot that is specifically designed to achieve human-like swing-leg speeds and human-like foot placement accuracy through powerful force-controllable actuators. To our knowledge, there is no description in literature of a bipedal robot that is designed with these requirements in mind (though we suspect (since this information is classified) that Petman (Boston Dynamics, 2011) is a good first example of this).

9.5.2 On the control of walking

The findings in this thesis suggest that walking control only through foot placement is possible, but preferably accompanied by ankle strategies and possibly hip strategies. Nevertheless, since balance control is dictated by foot placement, a possible point of departure for the synthesis of a robust walking gait is the control of foot placement only. As a start, the CoP shift through active ankle control could be emulated by locking the ankle joint and using a slightly curved foot shape or by using a flat foot with an ankle with a fixed stiffness as demonstrated by Wisse et al. (2006). Also, we could start off by selecting a walking gait with a relatively small step time, which decreases the possible deviations from the desired CoM motion. Once walking is achieved through adequate foot placement control, one can start experimenting with the use of ankle torques and upper body motions to increase the gait robustness and fine-tune the walking motion.

To further increase the robustness of walking, there is potentially one stabilizing mechanism that can be used which is not considered in this thesis: ankle push-off with the trailing leg during and preceding the double stance phase. A study on planar walking models (Hobbelen and Wisse, 2008a) suggests that regulating the amount of push-off can improve the recovery from perturbations during gait by influencing

the CoM motion in the fore-aft direction. Biomechanical studies show that impulsive ankle push-off can also be used to elevate the body after severe perturbations and thereby provide time and clearance for proper positioning of the recovery limb and restrain the angular momentum of the upper body (Pijnappels et al., 2004, 2005). Little is known about how to control ankle push-off in three-dimensional walking, during aperiodic gaits or in case of disturbances. Further research on this subject seems desirable. Omission of ankle push-off in our presented analysis implies that our view on possible stabilizing strategies is perhaps a bit conservative.

Once a robust gait is achieved, a relevant next step would be to focus on the energetic cost of walking. The Limit Cycle Walking approach presented various control concepts which were beneficial for the energy-efficiency of walking. For example, the energetic cost of walking could be reduced by exploiting the natural dynamics of the swing leg or by employing ankle push-off with the stance ankle just before the swing foot touches the ground (Hobbelen, 2008). These control concepts could be integrated into the control algorithms presented in this thesis. However, integration of these concepts should not be attempted without a clear understanding of how they affect the capturability of the system. The capturability analysis specifies which control actions are will not lead to a fall and are therefore permissible. We are currently exploring ways to minimize the energetic cost of foot placement for a given foot placement strategy provided by the capturability analysis. We aim to use a simple swing leg model in combination with model predictive control online on the robot that can minimize the energetic cost of foot placement for a given step time and step location. A next step would be to not only optimize the execution of foot placement in terms of energetic cost, but also the planning of foot placement.

References

- Abbott, B.C., Bigland, B., and Ritchie, J.M. (1952), "The physiological cost of negative work", *The Journal of Physiology*, 117(3), p. 380.
- Abdallah, M. and Goswami, A. (2005), "A Biomechanically Motivated Two-Phase Strategy for Biped Upright Balance Control", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1996–2001.
- Aftab, Z., Wieber, P.B., and Robert, T. (2010), "Comparison of Capture Point estimation with human foot placement : Applicability and Limitations", . Proceedings of 5èmes Journées Nationales de la Robotique Humanoïde.
- Alexander, R.M. (1989), "Optimization and gaits in the locomotion of vertebrates", *Physiological Reviews*, 69(1199-1227), pp. 29–64.
- Aubin, J.P. (1991), *Viability theory*, Birkhäuser, Boston, MA.
- Aubin, J.P., Lygeros, J., Quincampoix, M., Sastry, S., and Seube, N. (2002), "Impulse Differential Inclusions: A Viability Approach to Hybrid Systems", *IEEE Transactions on Automatic Control*, 47(1), pp. 2–20.
- Bauby, C.E. and Kuo, A.D. (2000), "Active control of lateral balance in human walking", *Journal of Biomechanics*, 33(11), pp. 1433–1440.
- van den Bogert, A.J., Pavol, M.J., and Grabiner, M.D. (2002), "Response time is more important than walking speed for the ability of older adults to avoid a fall after a trip", *Journal of Biomechanics*, 35(2), pp. 199–205.
- Boston Dynamics (2011), "PETMAN - BigDog gets a Big Brother", . website.
http://www.bostondynamics.com/robot_petman.html
- van der Burg, J.C.E., Pijnappels, M., and van Dieën, J.H. (2005), "Out-of-plane trunk movements and trunk muscle activity after a trip during walking", *Experimental Brain Research*, 165(3), pp. 407–412.
- Byl, K. and Tedrake, R. (2008), "Metastable walking on stochastically rough terrain", *Proceedings of Robotics: Science and Systems IV*.
- Chen, J., Cheng, J., and Yu, G. (2001), "Virtual model control of a quadruped walking robot", *Journal of Shanghai Jiaotong University*, 35(12), pp. 1771–1775.
- Chevallereau, C., Abba, G., Aoustin, Y., Plestan, F., Westervelt, E.R., Canudas-de-Wit, C., and Grizzle, J.W. (2003), "RABBIT: a testbed for advanced control theory", *IEEE Control Systems Magazine*, 23(5), pp. 57–79.
- Chevallereau, C., Grizzle, J.W., and Shih, C. (2009), "Asymptotically Stable Walking of a Five-Link Underactuated 3-D Bipedal Robot", *IEEE Transactions on Robotics*, 25(1), pp. 37–50.
- Coleman, M.J., Garcia, M., Mombaur, K., and Ruina, A. (2001), "Prediction of stable walking for a toy that cannot stand", *Physical Review E*, 64(2), p. 022901.

- Collins, S.H., Ruina, A., Tedrake, R., and Wisse, M. (2005), "Efficient Bipedal Robots Based on Passive-Dynamic Walkers", *Science*, 307(5712), pp. 1082–1085.
- Collins, S.H., Wisse, M., and Ruina, A. (2001), "A Three-Dimensional Passive-Dynamic Walking Robot with Two Legs and Knees", *International Journal of Robotic Research*, 20(7), pp. 607–615.
- Controllab Products B.V. (2011), "20-sim", . website.
<http://www.20sim.com>
- Coros, S., Beaudoin, P., and van de Panne, M. (2010), "Generalized biped walking control", *Proceedings of SIGGRAPH*, pp. 1–9. ACM Press, New York, NY, USA.
- De Lasa, M. (2010), "Feature-Based Control for Physics-Based Character Animation", , Ph.D. thesis, University of Toronto.
- Delft Biorobotics Laboratory (2011), "DBL homepage", . website.
<http://dbl.tudelft.nl>
- Diamond Systems (2011), "Poseidon EPIC SBC", . website.
<http://www.diamondsystems.com/products/poseidon>
- Diedam, H., Dimitrov, D., Wieber, P.B., Mombaur, K., and Diehl, M. (2008), "On-line walking gait generation with adaptive foot positioning through linear model predictive control", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1121–1126.
- van Dieën, J.H., Spanjaard, M., Konemann, R., Bron, L., and Pijnappels, M. (2007), "Balance control in stepping down expected and unexpected level changes", *Journal of Biomechanics*, 40(16), pp. 3641–3649.
- Dietz, V., Quintern, J., Boos, G., and Berger, W. (1986), "Obstruction of the swing phase during gait: phase-dependent bilateral leg muscle coordination", *Brain Research*, 384(1), pp. 166–169.
- Dingwell, J.B., Cusumano, J.P., Cavanagh, P.R., and Sternad, D. (2001), "Local Dynamic Stability Versus Kinematic Variability of Continuous Overground and Treadmill Walking", *Journal of Biomechanics Engineering*, 123(1), pp. 27–32.
- Doke, J., Donelan, J.M., and Kuo, A.D. (2005), "Mechanics and energetics of swinging the human leg", *Journal of Experimental Biology*, 208(3), p. 439.
- Doke, J. and Kuo, A.D. (2007), "Energetic cost of producing cyclic muscle force, rather than work, to swing the human leg", *Journal of Experimental Biology*, 210(13), p. 2390.
- Dollar, A.M. and Herr, H. (2008), "Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art", *IEEE Transactions on Robotics*, 24(1), pp. 144–158.
- Donelan, J.M., Kram, R., and Kuo, A.D. (2002), "Mechanical work for step-to-step transitions is a major determinant of the metabolic cost of human walking", *Journal*

- of *Experimental Biology*, 205(23), pp. 3717–3727.
- DutchRobotics (2011), “Homepage of DutchRobotics”, . website.
<http://www.dutchrobotics.net>
- Eiberger, O., Haddadin, S., Weis, M., Albu-Schaffer, A., and Hirzinger, G. (2010), “On joint design with intrinsic variable compliance: Derivation of the DLR QA-joint”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1687–1694.
- Elmo Motion Control (2011), “Whistle: Highly compact and intelligent digital servo drive”, . website.
<http://www.elmomc.com/products/whistle-digital-servo-drive-main.htm>
- Eng, J.J., Winter, D.A., and Patla, A.E. (1994), “Strategies for recovery from a trip in early and late swing during human walking”, *Experimental Brain Research*, 102(2), pp. 339–349.
- Englsberger, J., Ott, C., Roa, M.A., Albu-Schäffer, A., and Hirzinger, G. (2011), “Bipedal walking control based on Capture Point dynamics”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Featherstone, R. (1987), *Robot dynamics algorithms*, Kluwer Academic Publishers, Norwell, MA, USA.
- Featherstone, R. (2008), *Rigid body dynamics algorithms*, Springer US, Boston, MA, USA.
- Forner Cordero, A., Koopman, H., and van der Helm, F.C.T. (2003), “Multiple-step strategies to recover from stumbling perturbations”, *Gait & Posture*, 18(1), pp. 47–59.
- Forner Cordero, A., Koopman, H.F., and van der Helm, F.C.T. (2005), “Energy analysis of human stumbling: the limitations of recovery”, *Gait & Posture*, 21, pp. 243–254.
- Friedmann, M., Kiener, J., Petters, S., Thhomas, D., and von Stryk, O. (2006), “Modular software architecture for teams of cooperating, heterogeneous robots”, *Proceedings of IEEE International Conference on Robotics and Biomimetics*, pp. 613–618.
- Garcia, M., Chatterjee, A., Ruina, A., and Coleman, M. (1998), “The simplest walking model: Stability, complexity, and scaling”, *ASME Journal of Biomechanical Engineering*, 120, pp. 281–288.
- Gard, S.A., Miff, S.C., and Kuo, A.D. (2004), “Comparison of kinematic and kinetic methods for computing the vertical motion of the body center of mass during walking”, *Human movement science*, 22(6), pp. 597–610.
- Geng, T., Porr, B., and Wörgötter, F. (2006), “Coupling of neural computation with physical computation for stable dynamic biped walking control”, *Neural Comput.*,

- 18(5), pp. 1156–1196.
- Goswami, A. (1999), “Postural stability of biped robots and the foot rotation indicator (FRI) point”, *International Journal of Robotic Research*, 18(6), pp. 523–533.
- Grabiner, M.D., Koh, T.J., Lundin, T.M., and Jahnigen, D.W. (1993), “Kinematics of recovery from a stumble”, *The Journal of Gerontology*, 48(3), p. M97.
- Grizzle, J., Hurst, J., Morris, B., Park, H.W., and Sreenath, K. (2009), “MABEL, a new robotic bipedal walker and runner”, *IEEE American Control Conference*, pp. 2030–2036.
- Grizzle, J.W., Abba, G., and Plestan, F. (2001), “Asymptotically stable walking for biped robots: Analysis via systems with impulse effects”, *Transactions on Automatic Control*, 46(1), pp. 51–64.
- Grizzle, J.W., Chevallereau, C., Ames, A.D., and Sinnet, R.W. (2010), “3D bipedal robotic walking: Models, feedback control, and open problems”, *NOLCOS, Bologna, Italy*.
- Guihard, M. and Gorce, P. (2002), “Dynamic control of bipeds using ankle and hip strategies”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3.
- van Ham, R., Vanderborght, B., van Damme, M., Verrelst, B., and Lefeber, D. (2007), “MACCEPA, the mechanically adjustable compliance and controllable equilibrium position actuator: Design and implementation in a biped robot”, *Robotics and Autonomous Systems*, 55(10), pp. 761–768.
- Hirai, K., Hirose, M., Haikawa, Y., and Takenaka, T. (1998), “The development of Honda humanoid robot”, *Proceedings of IEEE International Conference on Robotics and Automation*, volume 2, pp. 1321–1326.
- Hobbelen, D.G.E. (2008), “Limit Cycle Walking”, , Ph.D. thesis, Delft University of Technology.
- Hobbelen, D.G.E. and De Boer, T. (2008), “System overview of bipedal robots Flame and TULip: Tailor-made for Limit Cycle Walking”, *Proceedings of International Conference on Intelligent Robots and Systems*, pp. 2486–2491.
- Hobbelen, D.G.E. and Wisse, M. (2007a), “A Disturbance Rejection Measure for Limit Cycle Walkers: The Gait Sensitivity Norm”, *IEEE Transactions on Robotics*, 23(6), pp. 1213–1224.
- Hobbelen, D.G.E. and Wisse, M. (2007b), “Limit Cycle Walking”, M. Hackel, editor, *Humanoid Robots, Human-like Machines*, chapter 14, pp. 277–294. ITech Education and Publishing, Vienna, Austria.
- Hobbelen, D.G.E. and Wisse, M. (2008a), “Ankle Actuation for Limit Cycle Walkers”, *The International Journal of Robotics Research*, 27(6), pp. 709–735.
- Hobbelen, D.G.E. and Wisse, M. (2008b), “Swing-Leg Retraction for Limit Cycle

- Walkers Improves Disturbance Rejection", *IEEE Transactions on Robotics*, 24(2), pp. 377–389.
- Hobbelen, D.G.E. and Wisse, M. (2009), "Active Lateral Foot Placement for 3D Stabilization of a Limit Cycle Walker Prototype", *International Journal on Humanoid Robotics*, 6(1), pp. 93–116.
- Hof, A.L. (2007), "The equations of motion for a standing human reveal three mechanisms for balance", *Journal of Biomechanics*, 40(2), pp. 451 – 457.
- Hof, A.L. (2008), "The 'extrapolated center of mass' concept suggests a simple control of balance in walking", *Human Movement Science*, 27(1), pp. 112–125.
- Hof, A.L., van Bockel, R.M., Schoppen, T., and Postema, K. (2007), "Control of lateral balance in walking: Experimental findings in normal subjects and above-knee amputees", *Gait & Posture*, 25(2), pp. 250–258.
- Hof, A.L., Gazendam, M.G.J., and Sinke, W.E. (2005), "The condition for dynamic stability", *Journal of Biomechanics*, 38(1), pp. 1–8.
- Hof, A.L., Vermerris, S.M., and Gjaltema, W.A. (2010), "Balance responses to lateral perturbations in human treadmill walking", *Journal of Experimental Biology*, 213, pp. 2655–2664.
- Honda (2011), "Asimo - the world's most advanced humanoid robot", . website.
<http://asimo.honda.com/>
- Hoogvliet, P., van Duyl, W.A., de Bakker, J.V., Mulder, P.G.H., and Stam, H.J. (1997), "A model for the relation between the displacement of the ankle and the center of pressure in the frontal plane, during one-leg stance", *Gait & Posture*, 6(1), pp. 39–49.
- Horak, F.B. and Nashner, L.M. (1986), "Central programming of postural movements: adaptation to altered support-surface configurations", *Journal of Neurophysiology*, 55(6), p. 1369.
- Hsiao, E.T. and Robinovitch, S.N. (1999), "Biomechanical influences on balance recovery by stepping", *Journal of Biomechanics*, 32(10), pp. 1099–1106.
- Hsiao, E.T. and Robinovitch, S.N. (2001), "Elderly subjects' ability to recover balance with a single backward step associates with body configuration at step contact", *Journal of Gerontology Series A: Biological Sciences and Medical Sciences*, 56(1), pp. M42–47.
- Hsiao-Weckler, E.T. (2008), "Biomechanical and age-related differences in balance recovery using the tether-release method", *Journal of Electromyography and Kinesiology*, 18(2), pp. 179–187.
- Hsiao-Weckler, E.T. and Robinovitch, S.N. (2007), "The effect of step length on young and elderly women's ability to recover balance", *Clinical Biomechanics*, 22(5), pp. 574–580.

- Hu, J.J., Pratt, J.E., Chew, C.M., Herr, H.M., and Pratt, G.A. (1998), "Adaptive virtual model control of a bipedal walking robot", *Proceedings of IEEE International Joint Symp. Intell. Syst.*, pp. 245–251.
- Hu, J.J., Pratt, J.E., Chew, C.M., Herr, H.M., and Pratt, G.A. (1999), "Virtual model based adaptive dynamic control of a biped walking robot", *International Journal of Artificial Intelligence Tools*, 8(3), pp. 337–348.
- Huang, Q., Kajita, S., Koyachi, N., Kaneko, K., Yokoi, K., Arai, H., Komoriya, K., and Tanie, K. (1999), "A high stability, smooth walking pattern for a biped robot", *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pp. 65–71.
- Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., and Tanie, K. (2001), "Planning walking patterns for a biped robot", *IEEE Transactions on Robotics and Automation*, 17(3), pp. 280–289.
- Hurst, J.W., Chestnutt, J.E., and Rizzi, A.A. (2004), "An actuator with physically variable stiffness for highly dynamic legged locomotion", *Proceedings of IEEE International Conference on Robotics and Automation*, volume 5, pp. 4662–4667.
- Hyon, S.H. and Cheng, G. (2007), "Disturbance Rejection for Biped Humanoids", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2668–2675. IEEE.
- Hyon, S.H., Hale, J.G., and Cheng, G. (2007), "Full-Body Compliant Human-Humanoid Interaction: Balancing in the Presence of Unknown External Forces", *IEEE Transactions on Robotics*, 23(5), pp. 884–898.
- Hyon, S.H., Osu, R., and Otaka, Y. (2009), "Integration of multi-level postural balancing on humanoid robots", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1549–1556. IEEE.
- Kagami, S., Kitagawa, T., Nishiwaki, K., Sugihara, T., Inaba, M., and Inoue, H. (2002), "A Fast Dynamically Equilibrated Walking Trajectory Generation Method of Humanoid Robot", *Journal of Autonomous Robots*, 12(1), pp. 71–82.
- Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., and Yokoi, K. (2003), "Biped walking pattern generation by using preview control of zero-moment point", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1620–1626.
- Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Yokoi, K., and Hirukawa, H. (2002), "A realtime pattern generator for biped walking", *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pp. 31–37.
- Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., and Hirukawa, H. (2001), "The 3D Linear Inverted Pendulum Mode: A simple modeling for a biped walking pattern generation", *Proceedings of IEEE/RSJ International Conference on Intelligent*

- Robots and Systems*, volume 1, pp. 239–246.
- Kajita, S. and Tanie, K. (1991), “Study of dynamic biped locomotion on rugged terrain—derivation and application of the linear inverted pendulum mode—”, *Proceedings of IEEE International Conference on Robotics and Automation*, volume 2, pp. 1405–1411. IEEE Comput. Soc. Press.
- Kanehiro, F., Hirukawa, H., and Kajita, S. (2004), “OpenHRP: Open architecture humanoid robotics platform”, *The International Journal of Robotics Research*, 23(2), p. 155.
- Kaneko, K., Kanehiro, F., Morisawa, M., Miura, K., Nakaoka, S., and Kajita, S. (2009), “Cybernetic human HRP-4C”, *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 7–14.
- Kazerooni, H., Racine, J.L., Huang, L., and Steger, R. (2005), “On the control of the berkeley lower extremity exoskeleton (BLEEX)”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 4353–4360.
- Kemp, M. (2010), “Humanoid robot TULip: modeling and control of advanced behaviors”, *BSc project thesis 026CE2010*, University of Twente.
- King, D.L. and Zatsiorsky, V.M. (2002), “Periods of extreme ankle displacement during one-legged standing”, *Gait & Posture*, 15(2), pp. 172–179.
- Kokam (2011), “Superior Lithium Polymer Battery (SLPB)”, . website.
http://www.kokam.com/english/product/battery_main.html
- Kram, R. and Taylor, C. (1990), “Energetics of running: a new perspective”, *Nature*, 346(6281), pp. 265–267.
- Kuo, A. (2001), “A simple model of bipedal walking predicts the preferred speed-step length relationship”, *Journal of Biomechanical Engineering*, 123, p. 264.
- Kuo, A. (2002), “Energetics of Actively Powered Locomotion Using the Simplest Walking Model”, *Journal of Biomechanical Engineering*, 124, p. 113.
- Kuo, A.D. (1999), “Stabilization of Lateral Motion in Passive Dynamic Walking”, *International Journal of Robotic Research*, 18(9), pp. 917–930.
- Kuo, A.D. (2007), “The six determinants of gait and the inverted pendulum analogy: A dynamic walking perspective”, *Human Movement Science*, 26(4), pp. 617–656.
- Lagoda, C., Schouten, A.C., Stienen, A.H.A., Hekman, E.E.G., and van der Kooij, H. (2010), “Design of an electric series elastic actuated joint for robotic gait rehabilitation training”, *Proceedings of International Conference on Biomedical Robotics and Biomechatronics*, pp. 21–26.
- Lee, S.H. and Goswami, A. (2007), “Reaction mass pendulum (RMP): An explicit model for centroidal angular momentum of humanoid robots”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 4667–4672. IEEE.
- Luchies, C.W., Alexander, N.B., Schultz, A.B., and Ashton-Miller, J. (1994), “Step-

- ping responses of young and old adults to postural disturbances: kinematics ", *Journal of the American Geriatrics Society*, 42(5), p. 506.
- Maki, B.E. and McIlroy, W.E. (1997), "The role of limb movements in maintaining upright stance: the "change-in-support" strategy", *Physical Therapy*, 77(5), pp. 488–507.
- Maki, B.E. and McIlroy, W.E. (1999), "The control of foot placement during compensatory stepping reactions: does speed of response take precedence over stability?", *IEEE Transactions on Rehabilitation Engineering*, 7(1), pp. 80–90.
- Manchester, I.R., Mettin, U., Iida, F., and Tedrake, R. (2009), "Stable dynamic walking over rough terrain: Theory and experiment", *Proceedings of International Symposium on Robotic Research*.
- Marigold, D.S. and Patla, A.E. (2002), "Strategies for dynamic stability during locomotion on a slippery surface: effects of prior experience and knowledge", *Journal of Neurophysiology*, 88(1), p. 339.
- Masani, K., Vette, A.H., and Popovic, M.R. (2006), "Controlling balance during quiet standing: Proportional and derivative controller generates preceding motor command to body sway position observed in experiments", *Gait & Posture*, 23(2), pp. 164–172.
- McGeer, T. (1990a), "Passive dynamic walking", *The International Journal of Robotics Research*, 9(2), p. 62.
- McGeer, T. (1990b), "Passive walking with knees", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1640–1645.
- McIlroy, W.E. and Maki, B.E. (1996), "Age-related changes in compensatory stepping in response to unpredictable perturbations", *Journal of Gerontology Series A: Biological Sciences and Medical Sciences*, 51(6), pp. 289–296.
- MESA (2011), "MESA motion control cards", . website.
<http://www.mesanet.com/motioncardinfo.html>
- Millard, M., Wight, D.L., McPhee, J., Kubica, E.G., and Wang, D.W.L. (2009), "Human Foot Placement and Balance in the Sagittal Plane", *Journal of Biomechanical Engineering*, 131(12), p. 121001.
- Mirtich, B.V. (1996), "Impulse-based dynamic simulation of rigid body systems", , Ph.D. thesis, University of California, Berkeley.
- Morimoto, J., Nakanishi, J., Endo, G., Cheng, G., Atkeson, C.G., and Zeglin, G. (2005), "Poincaré-Map-Based Reinforcement Learning For Biped Walking", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2381–2386.
- Morisawa, M., Harada, K., Kajita, S., Kaneko, K., Sola, J., Yoshida, E., Mansard, N., Yokoi, K., and Laumond, J.P. (2009), "Reactive stepping to prevent falling

- for humanoids", *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 528–534. IEEE.
- Morisawa, M., Kanehiro, F., Kaneko, K., Mansard, N., Sola, J., Yoshida, E., Yokoi, K., and Laumond, J. (2010), "Combining suppression of the disturbance and reactive stepping for recovering balance", *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, pp. 3150–3156.
- Nenchev, D.N. and Nishio, A. (2008), "Ankle and hip strategies for balance recovery of a biped subjected to an impact", *Robotica*, 26(05), pp. 643–653.
- Neuhaus, P.D., Noorden, J.H., Craig, T.J., Torres, T., Kirschbaum, J., and Pratt, J.E. (2011), "Design and evaluation of Mina: A robotic orthosis for paraplegics", *Proceedings of International Conference on Rehabilitation Robotics*, pp. 1–8.
- Nieuwenhuijzen, P., Schillings, A., van Galen, G., and Duysens, J. (2000), "Modulation of the startle response during human gait", *Journal of Neurophysiology*, 84(1), p. 65.
- Nishiwaki, K. and Kagami, S. (2010), "Strategies for Adjusting the ZMP Reference Trajectory for Maintaining Balance in Humanoid Walking", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 4230–4236.
- Nishtwaki, K., Nagasaka, K., Inaba, M., and Inoue, H. (1999), "Generation of reactive stepping motion for a humanoid by dynamically stable mixture of pre-designed motions", *Conference Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, volume 6, pp. 902–907.
- Okumura, Y., Tawara, T., Endo, K., Furuta, T., and Shimizu, M. (2003), "Realtime ZMP compensation for biped walking robot using adaptive inertia force control", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pp. 335–339.
- Orin, D.E. and Goswami, A. (2008), "Centroidal Momentum Matrix of a humanoid robot: Structure and properties", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 653–659.
- Össur (2011), "Website, Bionic Technology by Össur", .
<http://www.ossur.com>
- Ott, C., Baumgartner, C., Mayr, J., Fuchs, M., Burger, R., Lee, D., Eiberger, O., Albu-Schaffer, A., Grebenstein, M., and Hirzinger, G. (2010), "Development of a biped robot with torque controlled joints", *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 167–173.
- Otten, E. (1999), "Balancing on a narrow ridge: biomechanics and control", *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 354(1385), p. 869.
- Owaki, D., Koyama, M., Yamaguchi, S., Kubo, S., and Ishiguro, A. (2010), "A

- two-dimensional passive dynamic running biped with knees", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 5237–5242.
- Pai, Y.C. and Patton, J. (1997), "Center of mass velocity-position predictions for balance control", *Journal of Biomechanics*, 30(4), pp. 347–354.
- Pai, Y.C., Rogers, M.W., Patton, J., Cain, T.D., and Hanke, T.A. (1998), "Static versus dynamic predictions of protective stepping following waist-pull perturbations in young and older adults", *Journal of Biomechanics*, 31(12), pp. 1111–1118.
- Paluska, D.J. (2000), "Design of a Humanoid Biped for Walking Research", M. eng. thesis, Massachusetts Institute of Technology.
- Park, H.W., Sreenath, K., Hurst, J.W., and Grizzle, J.W. (2011), "Identification of a Bipedal Robot with a Compliant Drivetrain", *IEEE Control Systems*, 31(2), pp. 63–88.
- Pavol, M.J., Owings, T.M., Foley, K.T., and Grabiner, M.D. (2001), "Mechanisms leading to a fall from an induced trip in healthy older adults", *Journal of Gerontology Series A: Biological Sciences and Medical Sciences*, 56(7), pp. M428–437.
- Perry, J. (1992), *Gait Analysis: Normal and Pathological Function*, Slack, Thorofare, NJ, USA.
- Petters, S., Thomas, D., and Von Stryk, O. (2007), "RoboFrame - a modular software framework for lightweight autonomous robots", *Proceedings of Workshop on Measures and Procedures for the Evaluation of Robot Architectures and Middleware of the International Conference on Intelligent Robots and Systems*. Citeseer.
- Pijnappels, M., Bobbert, M., and van Dieën, J. (2004), "Contribution of the support limb in control of angular momentum after tripping", *Journal of Biomechanics*, 37(12), pp. 1811–1818.
- Pijnappels, M., Bobbert, M.F., and van Dieën, J.H. (2005), "How early reactions in the support limb contribute to balance recovery after tripping", *Journal of Biomechanics*, 38, pp. 627–634.
- Pijnappels, M., Kingma, I., Wezenberg, D., and Reurink, G and van Dieën, J.H. (2010), "Armed against falls: the contribution of arm movements to balance recovery after tripping", *Experimental Brain Research*, 201(4), pp. 689–699.
- Pijnappels, M., Reeves, N., Maganaris, C., and van Dieën, J. (2008), "Tripping without falling; lower limb strength, a limitation for balance recovery and a target for training in the elderly", *Journal of Electromyography and Kinesiology*, 18(2), pp. 188–196.
- Plestan, F., Grizzle, J.W., Westervelt, E.R., and Abba, G. (2003), "Stable walking of a 7-DOF biped robot", *IEEE Transactions on Robotics and Automation*, 19(4), pp. 653–668.
- Popovic, M.B., Goswami, A., and Herr, H.M. (2005), "Ground Reference Points in

- Legged Locomotion: Definitions, Biological Trajectories and Control Implications", *International Journal of Robotic Research*, 24(12), pp. 1013–1032.
- Pratt, G.A. (2000a), "Legged Robots at MIT: what's new since Raibert?", *IEEE Robotics and Automation Magazine*, 7(3), pp. 15–19.
- Pratt, G.A. and Williamson, M.M. (1995), "Series elastic actuators", *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, volume 1, pp. 399–406.
- Pratt, J.E. (2000b), "Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots", , Ph.D. thesis, Massachusetts Institute of Technology.
- Pratt, J.E., Carff, J., Drakunov, S.V., and Goswami, A. (2006), "Capture Point: A Step toward Humanoid Push Recovery", *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 200–207. IEEE.
- Pratt, J.E., Chew, C.M., Torres, A., Dilworth, P., and Pratt, G.A. (2001), "Virtual model control: an intuitive approach for bipedal locomotion", *International Journal of Robotic Research*, 20(2), pp. 129–143.
- Pratt, J.E. and Drakunov, S.V. (2007), "Derivation and Application of a Conserved Orbital Energy for the Inverted Pendulum Bipedal Walking Model", *Proceedings of IEEE International Conference on Robotics and Automation*.
- Pratt, J.E., Koolen, T., De Boer, T., Rebula, J., Cotton, S., Carff, J., Johnson, M., and Neuhaus, P. (2011), "Capturability-Based Analysis and Control of Legged Locomotion, Part 3: Application to M2V2, a Lower Body Humanoid", Submitted for publication.
- Pratt, J.E., Krupp, B., Ragusila, V., Rebula, J., Koolen, T., van Nieuwenhuizen, N., Shake, C., Craig, T., Taylor, J., Watkins, G., Neuhaus, P., Johnson, M., Shooter, S., Buffinton, K., Canas, F., Carff, J., and Howell, W. (2009), "The Yobotics-IHMC Lower Body Humanoid Robot", *Proceedings of International Conference on Intelligent Robots and Systems*, pp. 410–411.
- Pratt, J.E. and Pratt, G. (1999), "Exploiting natural dynamics in the control of a 3d bipedal walking simulation", *Proceedings of of International Conference on Climbing and Walking Robots (CLAWAR99)*.
- Pratt, J.E. and Pratt, G.A. (1998), "Intuitive control of a planar bipedal walking robot", *Proceedings of IEEE International Conference on Robotics and Automation*, volume 3, pp. 2014–2021.
- Pratt, J.E. and Tedrake, R. (2006), "Velocity-Based Stability Margins for Fast Bipedal Walking", M. Diehl and K. Mombaur, editors, *Fast Motions in Biomechanics and Robotics*, volume 340 of *Lecture Notes in Control and Information Sciences*, chapter 14, pp. 299–324. Springer Berlin Heidelberg.

- Raibert, M.H. (1986), *Legged robots that balance*, Massachusetts Institute of Technology.
- Randazzo, M., Fumagalli, M., Nori, F., Natale, L., Metta, G., and Sandini, G. (2011), "A comparison between joint level torque sensing and proximal F/T sensor torque estimation: implementation on the iCub", *Proceedings of International Conference on Intelligent Robots and Systems*.
- Raytheon Sarcos (2011), "www.raytheon.com", . website.
<http://www.raytheon.com>
- Rebula, J.R., Cañas, F., Pratt, J.E., and Goswami, A. (2007), "Learning capture points for humanoid push recovery", *Proceedings of IEEE-RAS International Conference on Humanoid Robots*.
- RoboCup (2011), "RoboCup, pushing the state-of-the-art", . website.
<http://www.robocup.org/>
- Roos, P.E., McGuigan, M.P., Kerwin, D.G., and Trewartha, G. (2008), "The role of arm movement in early trip recovery in younger and older adults", *Gait & Posture*, 27(2), pp. 352–356.
- ROS (2011), "Robot Operating System ROS", . website.
<http://www.ros.org/wiki/>
- Sabourin, C. and Bruneau, O. (2005), "Robustness of the dynamic walk of a biped robot subjected to disturbing external forces by using CMAC neural networks", *Robotics and Autonomous Systems*, 51(2-3), pp. 81 – 99.
- Sakata, K., Inoue, K., Takubo, T., Arai, T., and Mae, Y. (2004), "Wheelchair user support system using humanoid robots: system concept and experiments on pushing wheelchair", *SICE Annual Conference*, volume 2, pp. 1650–1655.
- Scancon (2011), "Mini hollow shaft encoder", . website.
<http://www.scancon.dk/scancon/upload/2RMHF.pdf>
- Schiavi, R., Grioli, G., Sen, S., and Bicchi, A. (2008), "VSA-II: A novel prototype of variable stiffness actuator for safe and performing robots interacting with humans", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2171–2176.
- Schillings, A., van Wezel, B., and Duysens, J. (1996), "Mechanically induced stumbling during human treadmill walking", *Journal of Neuroscience Methods*, 67(1), pp. 11–17.
- Schillings, A., van Wezel, B., Mulder, T., and Duysens, J. (1999), "Widespread short-latency stretch reflexes and their modulation during stumbling over obstacles", *Brain research*, 816(2), pp. 480–486.
- Schillings, A., van Wezel, B., Mulder, T., and Duysens, J. (2000), "Muscular Responses and Movement Strategies During Stumbling Over Obstacles", *Journal of*

- Neurophysiology*, 83(4), pp. 2093–2102.
- Sipser, M. (2005), *Introduction to the Theory of Computation*, Course Technology.
- Smeesters, C., Hayes, W.C., and McMahon, T.A. (2001), “The threshold trip duration for which recovery is no longer possible is associated with strength and reaction time”, *Journal of Biomechanics*, 34(5), pp. 589–595.
- Sreenath, K., Park, H.W., Poulakakis, I., and Grizzle, J. (2010), “A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on MABEL”, *International Journal of Robotics Research*, -, pp. -, published online.
- Stephens, B.J. (2007a), “Humanoid push recovery”, *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 589–595. IEEE.
- Stephens, B.J. (2007b), “Integral control of humanoid balance”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4020–4027. IEEE.
- Stephens, B.J. and Atkeson, C.G. (2010), “Dynamic balance force control for compliant humanoid robots”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Sugihara, T. (2010), “Consistent Biped Step Control with COM-ZMP Oscillation Based on Successive Phase Estimation in Dynamics Morphing”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 4224–4229. IEEE.
- Surveyor (2011), “Surveyor Stereo Vision System (SVS)”, . website.
<http://www.surveyor.com/stereo/>
- Takahashi, Y. (2009), “Systematic controller design for limit cycle walking”, , Msc thesis, Delft University of Technology.
- Tan, F., Fu, C., and Chen, K. (2010), “Biped blind walking on changing slope with reflex control system”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1709–1714.
- Tekscan (2011), “FlexiForce Load and Force Sensors and Systems”, . website.
<http://www.tekscan.com/flexiforce.html>
- Tellez, R., Ferro, F., Garcia, S., Gomez, E., Jorge, E., Mora, D., Pinyol, D., Oliver, J., Torres, O., Velazquez, J. et al. (2008), “Reem-B: An autonomous lightweight human-size humanoid robot”, *Proceedings of IEEE-RAS International conference on Humanoid Robots*, pp. 462–8.
- Thelen, D.G., Wojcik, L.A., Schultz, A.B., and Ashton-Miller, James A and Alexander, N.B. (1997), “Age differences in using a rapid step to regain balance during a forward fall”, *Journal of Gerontology Series A: Biological Sciences and Medical Sciences*, 52(1), pp. M8–13.
- Townsend, M.A. (1985), “Biped gait stabilization via foot placement”, *Journal of Biomechanics*, 18(1), pp. 21–38.

- Tripp, B.P., McIlroy, W.E., and Maki, B.E. (2004), "Online mutability of step direction during rapid stepping reactions evoked by postural perturbation", *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 12(1), pp. 140–152.
- Tsagarakis, N.G., Li, Z., Saglia, J., and Caldwell, D.G. (2011), "The design of the lower body of the compliant humanoid robot *ŞcCubT*", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2035–2040.
- Veneman, J.F., Ekkelenkamp, R., Kruidhof, R., van Der Helm, F.C.T., and van Der Kooij, H. (2006), "A series elastic-and bowden-cable-based actuation system for use as torque actuator in exoskeleton-type robots", *The International Journal of Robotics Research*, 25(3), p. 261.
- Vukobratovic, M. and Borovac, B. (2004), "Zero-Moment Point — Thirty Five Years of its Life", *International Journal of Robotic Research*, 1(1), pp. 157–173.
- Vukobratovic, M. and Juricic, D. (1969), "Contribution to the Synthesis of Biped Gait", *IEEE Transactions on Biomedical Engineering*, BME-16(1), pp. 1–6.
- Vukobratovic, M. and Stepanenko, J. (1972), "On the stability of anthropomorphic systems", *Math. Biosci.*, 15, pp. 1–37.
- Weerdesteyn, V., Nienhuis, B., Mulder, T., and Duysens, J. (2005), "Older women strongly prefer stride lengthening to shortening in avoiding obstacles", *Experimental Brain Research*, 161(1), pp. 39–46.
- Westervelt, E., Grizzle, J., and Koditschek, D.E. (2003), "Hybrid zero dynamics of planar biped walkers", *Transactions on Automatic Control*, 48(1), pp. 42–56.
- Westervelt, E.R., Buche, G., and Grizzle, J.W. (2004), "Experimental Validation of a Framework for the Design of Controllers that Induce Stable Walking in Planar Bipeds", *International Journal of Robotic Research*, 23(6), pp. 559–582.
- Wieber, P.B. (2000), "Constrained dynamics and parametrized control in biped walking", *Proceedings of International Symp. Math. Th. Netw. Syst.* SIAM, Perpignan France.
- Wieber, P.B. (2002), "On the stability of walking systems", *Proceedings of International Workshop Humanoid Hum. Friendly Robot.*, pp. 53–59. Tsukuba, Japan.
- Wight, D.L. (2008), "A Foot Placement Strategy for Robust Bipedal Gait Control", , Ph.D. thesis, University of Waterloo.
- Wight, D.L., Kubica, E.G., and Wang, D.W.L. (2008), "Introduction of the Foot Placement Estimator: A Dynamic Measure of Balance for Bipedal Robotics", *Journal of Computational Nonlinear Dynamics*, 3(1), 011009.
- Winter, D.A. (1990), *Biomechanics and Motor Control of Human Movement*, Wiley, 2nd edition.
- Wisse, M., Atkeson, C.G., and Kloimwieder, D.K. (2005a), "Swing leg retraction

- helps biped walking stability”, *Proceedings of IEEE-RAS International conference on Humanoid Robots*.
- Wisse, M., Feliksdal, G., Frankenhuyzen, J., and Moyer, B. (2007), “Passive-based walking robot-Denise, a simple, efficient, and lightweight biped”, *IEEE Robotics and Automation Magazine*, 14(2), pp. 52–62.
- Wisse, M., Hobbelen, D.G.E., Rotteveel, R.J.J., Anderson, S.O., and Zeglin, G.J. (2006), “Ankle springs instead of arc-shaped feet for passive dynamic walkers”, *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, pp. 110–116.
- Wisse, M., Schwab, A., and van Der Helm, F. (2004), “Passive dynamic walking model with upper body”, *Robotica*, 22(06), pp. 681–688.
- Wisse, M., Schwab, A.L., van der Linde, R.Q., and van der Helm, F.C.T. (2005b), “How to keep from falling forward: elementary swing leg action for passive dynamic walkers”, *IEEE Transactions on Robotics*, 21(3), pp. 393–401.
- Wojcik, L.A., Thelen, D.G., Schultz, A.B., Ashton-Miller, J.A., and Alexander, N.B. (2001), “Age and gender differences in peak lower extremity joint torques and ranges of motion used during single-step balance recovery from a forward fall”, *Journal of Biomechanics*, 34(1), pp. 67–73.
- Wolf, S. and Hirzinger, G. (2008), “A new variable stiffness design: Matching requirements of the next robot generation”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1741–1746.
- Xsens (2011), “Miniature Attitude and Heading Sensor”, . website.
<http://www.xsens.com/en/general/mti>
- Yin, K., Loken, K., and van de Panne, M. (2007), “SIMBICON: Simple Biped Locomotion Control”, *Proceedings of SIGGRAPH*, p. 105. ACM Press, New York, NY, USA.
- Yobotics, Inc (2011), “Yobotics Home. Legged Robots, Simulation, Actuators.”, . website.
<http://www.yobotics.com/>

Acknowledgements

Research on such a multi-disciplinary topic as robotics is not possible without the help of many.

I would like to thank my supervisor Martijn Wisse. The development of a new robot has been an enormous adventure and without your enthusiasm and optimism we would not have come this far. Thank you for the great cooperation. Thanks as well to Frans van der Helm, my promotor. Although you were more distantly involved, it always was motivating and inspiring to talk, either at the office or on the bike.

Throughout the years I have had the pleasure to work with many people. In the first place, a lot of people have contributed to the development of TULip. I would like to thank every student that has contributed to this research through the DutchRobotics foundation. I have continuously been amazed about how much enthusiasm a robot can arouse with people. I would like to mention several people specifically, as they have played a crucial role in the development of TULip: Tim van Bentum, Johannes Bertens, Mehdi Chitichian, Jeroen Fransman, Bram Hendriksen, Philip Heijkoop, Gijs van der Hoorn, Sebastiaan Kiemel, Twan Koolen, Jesper Smith, Arjan Smorenberg, Yutaro Takahashi, Erik de Vries. You have all contributed to this thesis by helping out with numerous different things, from designing parts to developing software and finding interested students or sponsors. I have greatly enjoyed working with you and learned a lot. I would also like to thank Guus Liqui Lung, Eelko van Breda, Boris Lenseigne en Jan van Frankenhuyzen from the DBL lab for their contribution. Thanks also to Henk Neijmeijer, Thom Warmerdam, Dragan Kostic, Edwin Dertien, Gijs van Oort and Stefano Stramigioli from Eindhoven and Twente for their enthusiasm and support of the DutchRobotics initiative. Also, I would like to thank Jerry Pratt, Twan Koolen and John Rebula for the many very interesting Skype meetings. After two and a half years we might get the two papers published after all!

Additionally, I would like to thank my colleagues at the DBL lab for the great ambience and the enjoyable lunches. My roommates Wietse van Dijk en Shiqian Wang for the exchanging of ideas about human and robotic walking. A big thanks to Erik and Daniël for the good times at work and outside the workplace, it has been great!

Finally, I would like to thank my parents, my sisters, my girlfriend and my friends for their support and encouragement.

Thank you all. It was a grand adventure!

Samenvatting

Voetplaatsing tijdens robotisch tweebenig voortbewegen

Menselijk lopen is opmerkelijk robuust, veelzijdig en efficiënt: mensen hebben het vermogen om grote onverwachte verstoringen op te vangen, een grote variëteit aan loopbewegingen uit te voeren en om daarbij weinig energie te verbruiken. Een tweebenige lopende robot die op al deze vlakken goed presteert is nog niet ontwikkeld. Sommige robots zijn in staat veelzijdige loopbewegingen uit te voeren, andere zijn efficiënt; geen enkele robot is robuust: alle robots verliezen vaak hun balans. De gebrekkige loopprestatie van robots bellemert hun toepasbaarheid in het dagelijks leven. Ook illustreert het dat de fundamentele beginselen van lopen niet goed worden begrepen. Het doel van dit proefschrift is om het begrip van de mechanica en aansturing van lopen te vergroten en om daarmee de prestatie van robotisch tweebenig lopen te vergroten. Een groter begrip is tevens nodig voor de ontwikkeling van robotische apparaten die mensen met een verminderd loopvermogen helpen of die de loopprestatie van valide personen verbeteren.

Lopen gaat in essentie over het vermogen de positie en snelheid van het massamiddelpunt van het lichaam te kunnen sturen. Sturing van het massamiddelpunt vereist het regelen van de krachten die via de voet uitgeoefend worden op het massamiddelpunt. De krachten die door het contact van de voet met de grond worden uitgeoefend kunnen tot op zekere hoogte gemanipuleerd worden door krachten in de enkel en bewegingen van het bovenlichaam, maar worden bovenal bepaald door de plaatsing van de voet ten opzichte van het massamiddelpunt. De beperkte invloed die de enkel en het bovenlichaam hebben op de contactkrachten en daarmee het massamiddelpunt worden het best geïllustreerd als men probeert te balanceren op één voet zonder een stap te nemen. Bij een kleine verstoring raakt men uit balans en moet er een stap genomen worden om een val te voorkomen. Dit laat zien dat balans houden tijdens het lopen afhankelijk is van de regeling van voetplaatsing (het plaatsen en timen van een stap). We concentreren ons daarom vooral op voetplaatsing bij het aansturen van robotische loopbewegingen.

Door deze focus onderscheidt dit onderzoek zich van andere onderzoeken in de robotica. Bij het aansturen van lopen gebaseerd op 'Zero Moment Point' wordt de toestand van de robot aangepast om een vooraf bepaalde voetplaatsing te bereiken. In Limit Cycle Walking wordt de voetplaatsing voornamelijk bepaald door passieve systeemodynamica. Dit proefschrift presenteert voetplaatsingstrategieën waarvan de tijd en plaats aangepast kunnen worden, die een expliciete functie zijn tussen de initiële toestand van de robot en de gewenste toestand van de robot en die relatief

weinig rekenkracht vereisen en daardoor real-time toepasbaar zijn op de robot. De bijdrage van dit proefschrift aan het onderzoek naar tweebenig lopen bestaat uit: een theoretisch kader, simulatiestudies en experimenten op een robot prototype. Deze bijdrage geeft inzicht in hoe sturing van voetplaatsing de robuustheid, veelzijdigheid en efficiëntie van een tweebenige loopbeweging kan vergroten.

Met betrekking op het onderzoek naar robuustheid wordt het theoretisch kader van *capturability* geïntroduceerd om acties die een val kunnen voorkomen te analyseren of synthetiseren. Het vermijden van een val wordt geanalyseerd door *N-step capturability*: het vermogen van het systeem om uiteindelijk tot stilstand te komen zonder te vallen door middel van het nemen van N of minder stappen, met inachtneming van zijn dynamica en actuatorlimieten. Met gebruik van simpele loopmodellen wordt een inschatting gemaakt van de *capturability* van complexe modellen. Zo wordt geïllustreerd hoe voetplaatsing, enkelkrachten en bewegingen van het bovenlichaam het massamiddelpunt beïnvloeden en bijdragen aan *N-step capturability*. *N-step capture regions*, die aangeven waar het systeem kan stappen om de mogelijkheid van tot stilstand komen te behouden, kunnen op de grond afgebeeld worden. Het oppervlak van deze gebieden kan gebruikt worden als een maat voor robuustheid.

Met betrekking op het onderzoek naar veelzijdigheid worden voetplaatsingstrategieën geformuleerd die het mogelijk maken voor het systeem om zijn toestand in een zo klein mogelijk aantal stappen te laten evolueren naar de gewenste toestand. Simulaties met simpele modellen laten zien hoe deze voetplaatsingstrategieën gebruikt kunnen worden om de loopsnelheid of looprichting te veranderen.

Met betrekking tot het onderzoek naar efficiëntie worden door simpele loopmodellen menselijke voetplaatsingstrategieën na struikelen vertoond wanneer voetplaatsing geoptimaliseerd wordt naar één van de volgende kostfuncties: maximaal koppel, vermogen, impuls, en koppel gedeeld door tijd. Met betrekking tot robotische aansturing impliceren deze resultaten dat actuatorlimieten in acht genomen moeten worden bij zowel het uitvoeren als het plannen van voetplaatsingstrategieën.

Met betrekking tot het uitvoeren van robotische experimenten zijn de concepten van het 'capturability' kader geïntegreerd in de aansturing van een robot. Het gebruik van een simpel (laagdimensionaal) loopmodel is nuttig gebleken voor het robuust regelen van een robot. Een dergelijk model houdt alleen rekening met de dynamica van het massamiddelpunt ten opzichte van het drukpunt onder de voet. De toepassing van dit model, samen met de op krachten gebaseerde aansturingsstrategieën leiden tot robuust robot gedrag: balans blijft behouden wanneer de robot staat en geduwd wordt terwijl één van de voeten op een bewegend oppervlak wordt geplaatst. Succesvolle toepassing wordt ook getoond met betrekking tot balanceren op één been

en het nemen van stappen om een duw te compenseren en balans te behouden, en (gesimuleerd) lopen.

De voornaamste conclusie is dat het analyseren van lopen als een combinatie van ontkoppelde en laagdimensionale taken, simpele en nuttige aansturings-regels kunnen verschaffen voor het aansturen van robotisch tweebeinig lopen. De belangrijkste aansturingstaak in lopen is het regelen van de positie en snelheid van het massamiddelpunt door voetplaatsing. Deze laagdimensionale opvatting van lopen maakt het afleiden van voetplaatsingstrategieën mogelijk die ervoor zorgen het systeem een val kan voorkomen. Deze set van robuuste voetplaatsingstrategieën geven de grenzen aan waarbinnen voetplaatsing aangepast kan worden om meer veelzijdig en efficiënt gedrag te bereiken.

Curriculum vitae

October 24, 1981

Born in Kortenhoef, the Netherlands.

1999-2006

M.Sc. study Mechanical Engineering at Eindhoven University of Technology, Eindhoven, the Netherlands. Graduated at the Control Systems Technology group. Final project conducted at Philips Research, titled 'Laboratory Study on Needle-Tissue Interaction: Towards the Development of an Instrument for Automatic Venipuncture'. The project concerned the mechatronic design and development of a prototype instrument for automatic venipuncture.

2006-2012

Ph.D. study at Mechanical engineering at Delft University of Technology, Delft, the Netherlands. Department of Biomechanical Engineering. The thesis project was titled 'Foot placement; in robotic bipedal locomotion' and concerned the development and control of a humanoid robot.