

Humanoid Robot Balance Control using Center of Mass Height Variation

B.J. van Hofslot

Humanoid Robot Balance Control using Center of Mass Height Variation

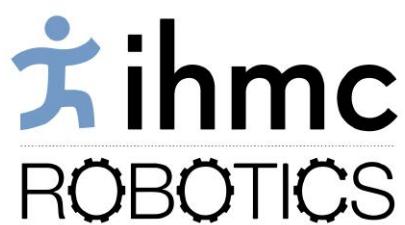
MASTER OF SCIENCE THESIS

For the degree of Master of Science in Systems and Control at Delft
University of Technology

B.J. van Hofslot

February 23, 2019

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of
Technology



The work in this thesis was supported by the Institute for Human and Machine Cognition.
Their cooperation is hereby gratefully acknowledged.



Copyright © Cognitive Robotics
All rights reserved.



Abstract

This research considers using center of mass (CoM) height variation as an input for balance control on a humanoid robot. Traditional balance strategies for humanoid robots are taking a step, control of the center of pressure (CoP) location, a result of the ‘ankle strategy’, and changing the angular momentum about the CoM, for example by a ‘hip strategy’. For humanoid robots, a common assumption behind these strategies is that the CoM height is predefined. However, CoM height changes can be used as an input for balance control, as for example can be observed during the landing of an athlete after a long jump.

The first contributions of this work are bounds on the initial states for the variable height inverted pendulum (VHIP) from which convergence is possible to a stopped state, also known as capture regions. First, only a unilateral contact constraint is considered; negative CoM acceleration cannot be smaller than gravitational acceleration. Second, CoM height constraints are added to the model, after which a capture region can still be computed in closed-form. Third, vertical force constraints are added, after which capture regions are computed numerically using a bang-bang control law. The last capture region bridges the transition to the applied part of this work.

The second contribution is a control law on vertical acceleration, suitable for application on a humanoid robot using a momentum-based control framework. Push recovery is tested on NASA’s Valkyrie humanoid robot while the robot is standing. In simulation, an increase in recoverable push of 9% can be observed when comparing to a controller that only uses CoP, when pushing the back of the robot. On hardware, an average increase of 7% can be observed for this push direction using a load sensor. Additionally, tests are conducted on hardware on Boston Dynamics’ Atlas using a medicine ball on a rope, but no improvement in recovery is observed. The control method for standing push recovery is also extended for use while the robot is walking. For Valkyrie in simulation, recovery improved the most compared with a predefined height approach for a push applied in the first part of the single support state for rear and frontal push directions. Additionally, a hardware result on Atlas while walking is briefly presented.

Table of Contents

Preface & Acknowledgments	ix
1 Introduction	1
1-1 Center of Mass Height Variation	2
1-2 Research Objective	2
1-3 Contributions	2
1-4 Thesis Outline	3
2 Background	5
2-1 Legged System Preliminaries	5
2-1-1 Human Balance Strategies	5
2-1-2 Ground Reference Points	6
2-1-3 Inverted Pendulum-Based Models	7
2-1-4 Orbital Energy & the Capture Problem	9
2-2 Humanoid Robotics at IHMC	11
2-2-1 Robots	11
2-2-2 Planning	11
2-2-3 Instantaneous Capture Point Control	13
2-2-4 Momentum-Based Whole-Body Control	13
2-3 Related Work	15
2-3-1 Dynamic Planning	16
2-3-2 Balance Control	17
3 Variable Height Inverted Pendulum Capture Regions	19
3-1 Unilateral Contact Constraint	19
3-2 Height Constraints	22
3-2-1 Maximum Height	22

3-2-2 Minimum Height	23
3-2-3 Bounds on Region	24
3-3 Vertical Force Constraints	25
3-4 Capturability Comparison	27
3-4-1 Comparison without Angular Momentum	28
3-4-2 Comparison with Angular Momentum	29
3-5 Discussion	31
4 Kinematically Constrained Orbital Energy Trajectories	33
4-1 Constraint Matrix with Final Velocity	33
4-1-1 Maximum Height Constraint	35
4-1-2 Maximum Leg Length Constraint	36
4-2 Discussion	37
5 Application: Standing	39
5-1 Method	39
5-2 Results	41
5-2-1 Valkyrie Simulation	41
5-2-2 Valkyrie Hardware	43
5-2-3 Atlas Hardware	48
5-2-4 Comparison with Capture Regions	51
5-3 Discussion	52
6 Application: Walking	55
6-1 Experimental Setup	55
6-2 Method	57
6-2-1 Avoiding Generating Additional Angular Momentum Rate	57
6-2-2 Decision Variables	58
6-2-3 Actions	59
6-3 Results	62
6-3-1 Valkyrie Simulation	62
6-3-2 Atlas Hardware	66
6-4 Discussion	69
7 Conclusion	71
7-1 Recommendations	72
7-1-1 Extending the Proposed Approach	72
7-1-2 Outlook	74
A Test Parameters	75
B 2D Analysis from Model to Robot	77
Bibliography	87
List of Acronyms	91

List of Figures

2-1	Balance strategies for a standing human; (A) shows the ‘ankle strategy’, (B) shows the ‘hip strategy’, (C) shows the ‘suspensory strategy’ and (D) shows the stepping strategy. Adopted from [15].	6
2-2	Ground reference points for different modeling choices. (a) The blue diamond points out the zero moment point (ZMP)/CoP location. As no angular momentum is considered in the model, this point is also the centroidal moment pivot (CMP). (b) A flywheel with inertia is added to the model, the blue circle with cross points out the CMP location and the blue diamond the ZMP/CoP location.	7
2-3	The three-dimensional space (3D) motion of a linear inverted pendulum (LIP) model.	8
2-4	The 3D motion of a VHIP model.	9
2-5	Visualization of the path to the LIP capture point (LIPCP).	10
2-6	(a) Atlas [28] and (b) Valkyrie [29] walking over an uneven cinder block field at the Institute for Human and Machine Cognition (IHMC).	12
2-7	Approximation of the wrench cone with basis vectors β_{ij} for ground contact point i . The linear part of the ground reaction wrench, f_i in the drawing, is a positive multiplication of the basis vectors and lies inside the wrench cone. Adopted from [8].	15
2-8	High-level overview of the control framework. Adopted from [8].	16
3-1	Visualization of feasible final point-mass locations for the unilateral contact constrained capture region (gray area). The values $\dot{x}_0 = 1.0 \text{ [m/s]}$, $z_0 = 1.0 \text{ [m]}$, $\dot{z}_0 = 0.0 \text{ [m/s]}$ and $g = 9.81 \text{ [m/s}^2]$ are used. The thin black lines are made with the method presented in [13] and visualize possible intermediate trajectories.	21
3-2	Visualization of feasible final point-mass locations for the analytic capture regions. The values $\dot{x}_0 = 1.0 \text{ [m/s]}$, $z_0 = 1.0 \text{ [m]}$, $\dot{z}_0 = 0.0 \text{ [m/s]}$ and $g = 9.81 \text{ [m/s}^2]$ are used. The light gray area shows the feasible final points in the unilateral contact constrained capture region (Section 3-7). The dark gray area shows the feasible final points in the height constrained capture region (Lemma 1) for $0.7 < z < 1.1 \text{ [m]}$. The dotted plots are made with the <i>orbital energy controller</i> of [13] and show that the final points are inside the height constrained region.	26

3-3	Vertical force constrained capture positions relative to the height constrained bounds. The acceleration $\ddot{z}_c = \ddot{z}_{c,1} = \ddot{z}_{c,2} $ if $\ddot{z}_c \leq g$ and else the negative 'bang' is set to $-g$. The values $\dot{x}_0 = 1.0$ [m/s], $z_0 = 1.0$ [m], $\dot{z}_0 = 0.0$ [m/s] and $g = 9.81$ [m/s ²] are used.	27
3-4	Reachable dimensionless capture positions for $\dot{x}'_0 = 1$ (left) and capture velocities for $x'_0 = 1$ (right) without angular momentum. Deviations from $z' = 1$ show which minimum or maximum height constraint is used in the computation of the capture points, $z'_0 = 1$	29
3-5	Comparison of capture region with angular momentum versus vertical force constrained capture. The height constraints are set equal, $\delta z'_{min} = \delta z'_{max}$ and $I'_y = \frac{1}{6}$. The solid plots show the first bound on the region and the dotted plots the second bound.	30
4-1	Resulting polynomials as output from Algorithm 1 with two different constraints on the final value. The dash-dotted line shows the height constraint $z_{max} = 1.1$ [m]. Initial conditions are $x_0 = -0.25$ [m], $\dot{x}_0 = 1.0$ [m/s], $z_0 = 1.0$ [m] and $\dot{z}_0 = 0$ [m/s]. Blue plot: $\dot{x}_f = 0.552$ [m/s], red plot: $\dot{x}_f = 0.576$ [m/s], $g = 9.81$ [m/s ²].	36
4-2	Resulting polynomials as output from Algorithm 2 with two different initial values under the final height constraint $z_f = 1.0$. The dash-dotted line shows the length constraint $l_{max} = 1.15$ [m]. Other initial conditions are $\dot{x}_0 = 1.4$ [m/s], $z_0 = 1.0$ [m] and $\dot{z}_0 = 0.0$ [m/s]. Blue plot: $\dot{x}_f = 1.107$ [m/s], red plot: $\dot{x}_f = 0.724$ [m/s], $g = 9.81$ [m/s ²].	37
4-3	Position plot (left) versus the resulting vertical acceleration (right) with $x_0 = -0.3$ [m], $z_0 = 1.0$ [m], $\dot{x}_0 = 1.0$ [m/s], $\dot{z}_0 = 0.0$ [m/s] and $g = 9.81$ [m/s ²].	38
5-1	Polar plot of maximum recoverable pushes with an increment of 5 degrees. 0 degrees is a push from the back. The radius is the normalized impulse i	42
5-2	Phase plot of horizontal CoM motion in the sagittal plane for a push of $i = 0.271$ [m/s] (solid) and a push of $i = 0.295$ [m/s] (dotted).	43
5-3	Time response plots for a push of $i = 0.271$ [m/s] for the default setup (black) and the vertical motion controller (blue). 'Achieved' is the value of the variable after the quadratic program (QP) found a solution. The gray area is where the push is applied.	44
5-4	(a) Push head of push stick with a rubber surface, steel plate, load sensor and aluminum stick. (b) Tests setup, where the author applies a push using the push stick on Valkyrie.	45
5-5	Average impulse of 12 pushes, where the CoM came closer than 5 [mm] from the polygon edge for the (a) default setup and (b) vertical motion controller. (c) Two picks of pushes, where the force profile and integrated force applied on both setups were similar.	45
5-6	Phase plot of sagittal CoM motion for a push of $i = 0.261$ [m/s].	45
5-7	Time-lapse of Valkyrie recovering from a push using vertical motion (top row) and using the default controller setup (bottom row). The letters below the columns match with the letters next to the yellow lines in Figure 5-8. The push rod tip is encircled in red.	46
5-8	Time response plots for a push of $i = 0.261$ [m/s] for the default setup (black) and the vertical motion controller (blue). All joint torques, except for the back, are the average over left and right.	47

5-9	Experimental setup for hardware tests on Atlas.	48
5-10	Release mechanism for the medicine ball tests.	49
5-11	CoP tracking during the two ‘bangs’ of the vertical motion control law. In the left column, $k_{\dot{q}} = 15.0$. In the right column, $k_{\dot{q}} = 40.0$. For illustration of the ‘bangs’, $\dot{\mathbf{I}}_{d,z}$ is graphed.	50
5-12	Phase plots of 20 Atlas hardware tests for each setup (thin transparent) with the averaged data for each setup (bold).	51
5-13	Difference in recovery of Valkyrie and Atlas in simulation and on hardware between the vertical motion controller and the default setup, plotted in the capture velocity plot in Figure 3-4.	52
6-1	Test setup for push recovery during walking in simulation. The limited foothold options show that footstep adjustment is not available as a balance strategy. . .	56
6-2	Trajectories during single support in the horizontal plane (gray dotted lines). The gray area is the current, right, footstep position where the push will be applied. .	56
6-3	Explanatory drawing of the ground reaction force (GRF) resulting from the default desired momentum rate (black) versus the modified desired momentum rate for CoM height variation (red). The thin arrows show the part of the GRF that intersects with the CoM, which is used for height changes. The equal scalar offset a shows that the same angular momentum rate will be requested about the CoM.	58
6-4	Explanatory visualizations of ϕ and δ for the configuration at start of single support with different instantaneous capture point (ICP) error directions.	59
6-5	Height constraints throughout single support for the vertical motion controller while the robot is walking.	61
6-6	Polar plots of the maximum recoverable impulses i (radius) for the default controller (black) and the vertical motion controller (blue) for pushes applied at different moments in single support. The right foot is the support foot and 0 degrees corresponds with a push from the back. The colors of the radial lines show the actions used by the vertical motion control law.	64
6-7	Polar plots of the maximum recoverable impulses i (radius) for the default controller (black) and vertical motion controller (blue). The right foot is the support foot and 0 degrees corresponds with a rear push. A comparison is shown for when the desired CMP is constrained to be inside the polygon, and when the desired CMP is allowed to leave the polygon 0.05 [m].	65
6-8	Time responses after a rear push of $i = 0.156$ [m/s] and a frontal push of 0.315 [m/s] are applied on Valkyrie for the default setup (black) and the vertical motion controller (blue). The push is applied in the dark gray area and the light gray area shows when the robot is in double support.	66
6-9	Visualization in 3D of the CoM and $\mathbf{r}_{cmp,d}$ trajectory during the walking push recovery test (top). The yellow lines are the pendulums projected on the xz -plane. The letters above the yellow lines correspond with the letters below the images in Figure 6-10 and show the $\mathbf{r}_{cmp,d}$ - \mathbf{c} configurations. At a , the push is applied and at b the push is released. Also, $\dot{\mathbf{I}}_{d,z}$ over time is shown (bottom).	67
6-10	Time-lapse of Atlas recovering from a push using the positive alignment action. The letters below the images match with the corresponding letters above the yellow lines in Figure 6-10. The push rod tip is encircled in red.	68

List of Tables

6-1 Stepping parameters for the walking tests.	55
A-1 Control framework parameters for the Valkyrie tests.	75

Preface & Acknowledgments

Having a background in mechanical engineering, I have always been motivated in closing the gap between theory and application on a physical system during my master's in control systems. The topic of humanoid robotics offers a very interesting, challenging, platform to dedicate my motivation to. The complex multi-body system of a humanoid robot copes with nonlinearity, hybrid dynamics, actuation limitations and plays with your human intuition. Also, humanoid robots are still physically far behind of what a human can do, which proves that there is large room for improvement. I hope in the future, robots will be able to save lives. I believe reaching out to IHMC was the best decision to learn from and contribute to this field of research.

I would like to thank IHMC, for giving me the opportunity to conduct research at the robotics lab. I am particularly grateful for the supervision that was given to me by Dr. Robert Griffin, Dr. Sylvain Bertrand and Dr. Jerry Pratt. I would like to thank everybody else at the robotics lab for their advice and the joy I experienced of working at the lab.

I would also like to thank Dr. Javier Alonso-Mora from Delft University of Technology, for supervising me throughout the year I was abroad. Also, I would like to thank the assessment committee members Prof. Dr. Martijn Wisse, Dr. Tamas Keviczky and Dr. Carlos Hernandez Corbato.

I would like to thank my mother, father and two brothers, for always being supportive.

Delft, University of Technology
February 23, 2019

B.J. van Hofslot

“Playing soccer is very simple, but the hardest thing there is, is playing soccer in a simple way.”

— *Johan Cruyff*

Chapter 1

Introduction

Humans are often exposed to dangerous tasks. This is the case, for example, in disaster response: exploring a radioactive environment to help people after a nuclear disaster, or entering a building on fire to save a life. Another example is an exploration space mission to Mars, where a human committing to such a task is in risk of harm, but also costs a lot of time and money. In all these situations, the physical versatility of the human, such as the ability to grasp with hands and to walk over uneven terrain, is needed. Substitution of a human-like machine in such scenarios, which has similar benefits as a human, could be very beneficial.

Replacing the physical human with a machine is not a new topic. There already exist ancient texts, that describe a human-like machine [1]. As technology became more advanced, the topic of humanoid robotics became more popular in the last decades. However, a humanoid robot that has the same physical capabilities as an average human being still does not exist. One of those physical capabilities is maintaining balance. Humans are capable of not falling over in various terrains, configurations and subject to disturbances, while robots are frequently not. Commonly used balancing strategies for humanoid robots are taking a step, control of the center of pressure (CoP) ('ankle strategies') and, to a lesser extent, change of body angular momentum (for example: 'hip strategies').

A common assumption behind these strategies is that the center of mass (CoM) height of the robot is predefined and not to be used in balancing tasks. By constraining the height, a degree of freedom of the system is defined and the dynamics of the system are known. Furthermore, if height changes are small, the system can be approximated with a linear inverted pendulum (LIP) model [2] in the generation of a dynamic reference plan for the robot. An important advantage of this model is that, by its linearity, closed-form solutions exist to the dynamics, as for example the LIP orbital energy [3]. The linear dynamics of the LIP are a saddle point with one stable eigenvalue and one unstable eigenvalue. The unstable mode has been referred to as the LIP capture point (LIPCP) [4], the instantaneous capture point (ICP) [5], the extrapolated center of mass [6] and the divergent component of motion (DCM) [7], which all relate to the LIP orbital energy.

1-1 Center of Mass Height Variation

In addition to a LIP-based dynamical plan, CoM height reference trajectories for the robot are traditionally generated for kinematic feasibility [8, 9] or more human-like motion, but are disturbances on the LIP model considered in dynamic planning. These disturbances are commonly rejected with the traditional balance control strategies. Efforts have been made in incorporating CoM height variation in the dynamic reference plan. Examples are the variable height inverted pendulum (VHIP) orbital energy [10], the three-dimensional space (3D) DCM [11] and the time-varying DCM [12]. The presented methods either introduce artificial constraints [10] or cope with larger computation times [12].

Recently, the use of the CoM height variation came in sight with the particular goal to improve balance control. Using a VHIP model, height changes of the pendulum tip can influence the horizontal dynamics, which can be used as a control input for balance. Some work has considered this control input by formulating a model predictive control (MPC) law [13, 14] for computing capture trajectories: trajectories that will result in the point-mass to stop. In both publications, the objective is to let the point-mass converge within the current ground contact without taking additional steps, based on the initial configuration and velocity. This is also known as ‘0-step’ capture [5]. In MPC, fast computation times are even more needed than in dynamic planning, as a reference trajectory is evaluated every control tick. Control rates are commonly between 200 [Hz] and 800 [Hz] on humanoid robots [8, 9]. The methods therefore introduce artificial constraints to solve the problem online.

Relying on the mentioned work, no hardware results have been shown yet in using CoM height variation for balance control. Also, if a method is proposed, no comparison is made with constant height approaches in simulation. Therefore, it is difficult to judge the differences in performance between the proposed control law and for example a LIP-based controller in an applied setting. In this report, comparison between predefined height approaches and approaches that use CoM height variation for balance will be central in the evaluation of the presented theory and methods.

1-2 Research Objective

The objective of this work is to improve balance of a humanoid robot using vertical CoM motion. To measure ‘improving balance’, a high-level differentiation is made between:

- **Theory:** analysis of the variable height inverted pendulum (VHIP) and comparison with the linear inverted pendulum (LIP).
- **Application:** comparison of results obtained after application of a control law on a robot, using a commonly used control framework to transcribe pendulum-based control commands to the joints of the robot. In this control law, fast computation times are required.

1-3 Contributions

The contributions of this work are novel capture regions for the variable height inverted pendulum (VHIP) model, which addresses the theoretical part of the research objective. Also, results are presented on hardware using CoM height variation for balance control, which addresses the applied part of the research objective. More specifically, relating to theoretical contribution, this work presents:

- **VHIP capture regions:** bounds on the initial states of a VHIP model, from which convergence of the unstable mode to a stopped state is possible. The regions are obtained by incrementally adding constraints to the VHIP model. First, only unilateral contact constraints are considered. After this, height constraints are added and finally, vertical force constraints are added.
- **Kinematically constrained orbital energy trajectories:** a MPC law that takes kinematic constraints into account, extending the method presented in [13]. This method is not further used in application, as the polynomial function used in the control law is found to be overly constraining the VHIP dynamics.

Relating to the objective of comparison of applied results, this work presents:

- **A method that uses CoM height variation for balancing while standing.** This is a bang-bang control law on vertical acceleration. Push recovery tests are conducted on NASA's Valkyrie humanoid robot and Boston Dynamics' Atlas. Results are compared with results of a constant CoM height control approach.
- **A method that uses CoM height variation for balancing while walking.** This is an extension of the standing method for walking scenarios. This method is tested on Valkyrie in simulation and is compared with a predefined CoM height approach. Additionally, one hardware result on Atlas is presented.

1-4 Thesis Outline

The remainder of this thesis is structured as follows. In Chapter 2, a brief background is given on commonly used expressions in legged systems research, on humanoid robotics at the Institute for Human and Machine Cognition (IHMC) and on related work to CoM height variation. In Chapter 3, capture regions are derived for the VHIP model. In Chapter 4, a two-dimensional space (2D) MPC law is derived, which is not further used in application. In Chapter 5, a control law is presented and push recovery is tested on NASA's Valkyrie while the robot is standing, with additional results shown on Boston Dynamics' Atlas. In Chapter 6, this control law is extended for the use during walking and push recovery is tested on Valkyrie during walking in simulation. Additionally, a hardware result on Atlas is presented. The final chapter, Chapter 7, presents the conclusion and recommendations for future work.

Chapter 2

Background

In this chapter, a brief background is given on legged systems, humanoid robotics at the Institute for Human and Machine Cognition (IHMC) and related work to center of mass (CoM) height variation.

2-1 Legged System Preliminaries

In this section, commonly used expressions and background related to legged systems are briefly presented.

2-1-1 Human Balance Strategies

As humanoid robots are a derivation of humans, human balance strategies are briefly discussed. In Figure 2-1, balance strategies are shown for a standing human. The ‘ankle strategy’, ‘hip strategy’ and stepping strategy are most commonly considered as a balance strategy. However, the figure includes the ‘suspensory strategy’ [15], which is less commonly considered. With the ‘suspensory strategy’, the human is in a slightly lower configuration with bent knees to have more control authority of the ankles.

For balancing, height variation is often not considered for a standing human. A reason for this might be that often the assumption is made that the legs are straight [16]. The goal of the ‘suspensory strategy’ however, gives an interesting insight to the problem. Using height changes for balancing, the control authority of the ankles will change. Therefore, height variations for balance control could be a trade-off between the application of additional force and the gain in height.

For dynamic cases, like walking or jumping, CoM height variations for balance on a human can be observed in the landing after a long jump. Bending the legs and lowering the height is crucial for not falling backwards and to land above the feet.

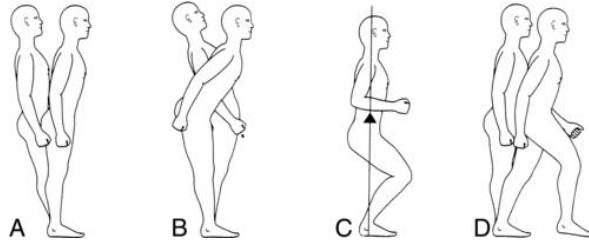


Figure 2-1: Balance strategies for a standing human; (A) shows the ‘ankle strategy’, (B) shows the ‘hip strategy’, (C) shows the ‘suspensory strategy’ and (D) shows the stepping strategy. Adopted from [15].

2-1-2 Ground Reference Points

In biped locomotion, the dynamics of the system are often simplified by considering the forces resulting from ground contact, the CoM location and the angular moment about the CoM. The contact forces are commonly summed up in a single ground reaction force (GRF), coming from a point of application in the supporting area of the system. Ground reference points are used to describe the dynamics of the system in a single point, using GRF and CoM states.

Zero Moment Point & Center of Pressure

The point on the ground where the resulting GRF does not produce any moment in the horizontal plane at the point of application, is referred to as the zero moment point (ZMP) [17]. By definition, this is the point where the part of the GRF that does not cause angular momentum about the CoM intersects with the ground surface. The ZMP was initially introduced in [18]. The ZMP is formulated as:

$$\mathbf{r}_{zmp} = \mathbf{c}_{xy} - \frac{\mathbf{F}_{gr,xy}}{F_{gr,z}} z + \frac{\boldsymbol{\tau}_c}{F_{gr,z}}, \quad (2-1)$$

where $\mathbf{r}_{zmp} = [x_{zmp}, y_{zmp}]^T$ is the ZMP location, $\mathbf{F}_{gr,xy} = [F_{gr,x}, F_{gr,y}]^T$ and $F_{gr,z}$ are the horizontal and vertical components of the GRF respectively, $\mathbf{c}_{xy} = [x, y]^T$ and z are the components of the CoM Cartesian position and $\boldsymbol{\tau}_c = [-\tau_y, \tau_x]^T$ is the torque about the CoM.

In Figure 2-2, the location of the ZMP is visualized for two modeling choices, using both a connection between the ankle and the CoM as a prismatic joint. In Figure 2-2a, no inertia is considered about the CoM and the GRF \mathbf{F}_{gr} coming from the ZMP intersect with the mass m . The difference in position between the ankle location and the ZMP is affected by the ankle torque $\boldsymbol{\tau}_{ankle}$. In Figure 2-2b, body inertia of the system is approximated by adding a flywheel with inertia $\mathbf{I} = [I_y, I_x]^T$ to the model. The GRF, coming from the ZMP, is pointed away from the CoM by using the body torque $\boldsymbol{\tau}_c$ as input. The dynamics of m are depending on \mathbf{F}_{gr} and $\mathbf{g} = [0, 0, -g]^T$, which is the gravity vector with g the gravitational constant.

The center of pressure (CoP) coincides during walking over flat ground with the ZMP [19]. The two points however are not equal in more complex environments. The CoP is restricted to be located in the support polygon, while the ZMP is restricted to be located on the ground plane [17]. Traditionally, the CoP is a measured quantity from a force pressure plate under the foot. In this thesis, the CoP location is denoted as $\mathbf{r}_{cop} = [x_{cop}, y_{cop}]^T$ and considered equal to \mathbf{r}_{zmp} , as a flat contact surface is used as reference.

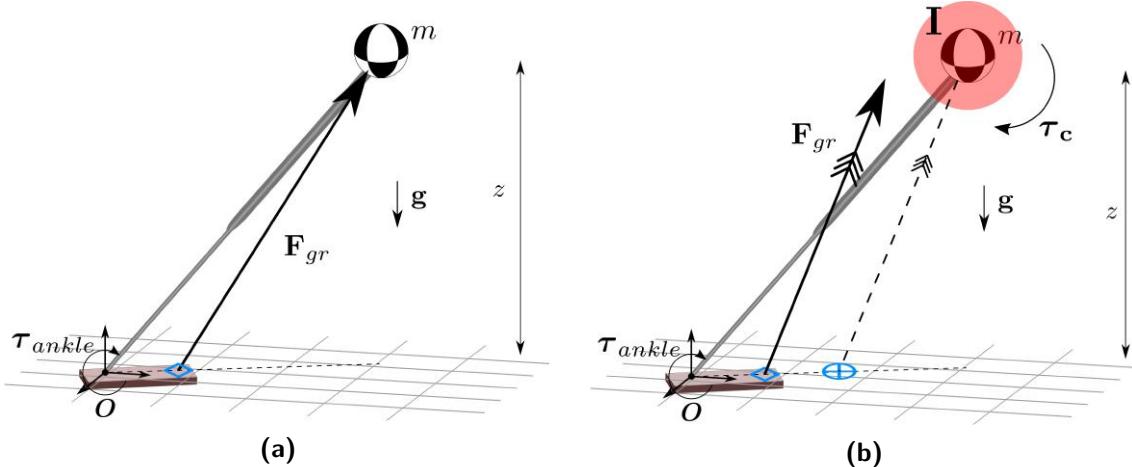


Figure 2-2: Ground reference points for different modeling choices. (a) The blue diamond points out the ZMP/CoP location. As no angular momentum is considered in the model, this point is also the centroidal moment pivot (CMP). (b) A flywheel with inertia is added to the model, the blue circle with cross points out the CMP location and the blue diamond the ZMP/CoP location.

Centroidal Moment Pivot

The centroidal moment pivot (CMP) includes, unlike the ZMP and CoP, angular momentum about the CoM [20]. This is defined as the point where a line passing through the CoM, parallel to the GRF intersects with the ground surface. Unlike the CoP, the CMP is not constrained to lie inside the support polygon. The CMP is defined as:

$$\mathbf{r}_{cmp} = \mathbf{c}_{xy} - \frac{\mathbf{F}_{gr,xy}}{F_{gr,z}} z, \quad (2-2)$$

where $\mathbf{r}_{cmp} = [x_{cmp}, y_{cmp}]^T$ is the CMP location. In Figure 2-2b, the difference between the ZMP and CMP is graphically explained. Without body inertia in the model, the points coincide, as is depicted in Figure 2-2a. Equivalently, if $\tau_c = 0$, the points coincide as well.

2-1-3 Inverted Pendulum-Based Models

The line between a ground reference point and the CoM can be modeled as an inverted pendulum. In this thesis, this line is also called the *virtual leg*.

Linear Inverted Pendulum Model

For its fast, closed-form solutions, the linear inverted pendulum (LIP) model [2] is widely used in walking research and especially in legged robotics. The LIP equations of motion are:

$$\ddot{\mathbf{c}}_{xy} = \frac{g}{l} \mathbf{c}_{xy}, \quad (2-3)$$

where $\ddot{\mathbf{c}}_{xy}$ is the horizontal CoM acceleration. At any horizontal position, a constant leg length is considered and the motion is at a constant height $l = z_0$. In Figure 2-3, the three-dimensional space (3D) motion is visualized if the CoM is relatively far from the base. The

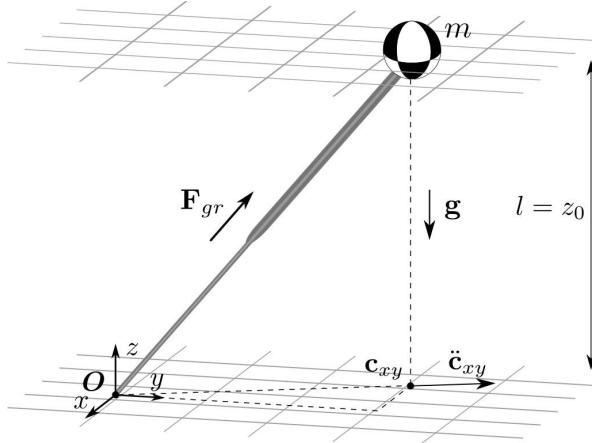


Figure 2-3: The 3D motion of a LIP model.

pendulum base lies in the origin. Because the LIP assumption holds, the vertical component of \mathbf{F}_{gr} cancels out gravity acceleration: $F_{gr,z} = mg$.

To model body inertia of the robot, sometimes a flywheel is added to the LIP [4, 21, 5]. By controlling the torque applied on the flywheel, a control authority over the CoM dynamics becomes available.

Height Varying Models

Unlike the LIP, height variation of the CoM can be included in modeling of the virtual leg. Three examples of such models are:

- The, not linearized, inverted pendulum model [16];
- The spring-loaded inverted pendulum (SLIP) model [22];
- A pendulum with prismatic joint, not constrained to maintain a constant height: the variable height inverted pendulum (VHIP) [10].

The inverted pendulum model is often used in human motion research, as in [16]. The advantages of a LIP, like fast, closed-form solutions to the dynamics, are often not needed, as there is no dynamic planning and control involved. The SLIP model originates from hopping and running robots [23]. Deviations from the nominal height or pendulum length are modeled as mass-spring dynamics.

Throughout this report, special focus is given to the VHIP. In Figure 2-4, the VHIP is depicted. The dynamics of the point-mass can be written in two ways. One is as a function of the GRF in two-dimensional space (2D):

$$m\ddot{x} = \mathbf{F}_{gr,2D} \frac{x}{\sqrt{x^2 + z^2}}, \quad (2-4)$$

$$m\ddot{z} = \mathbf{F}_{gr,2D} \frac{z}{\sqrt{x^2 + z^2}} - mg, \quad (2-5)$$

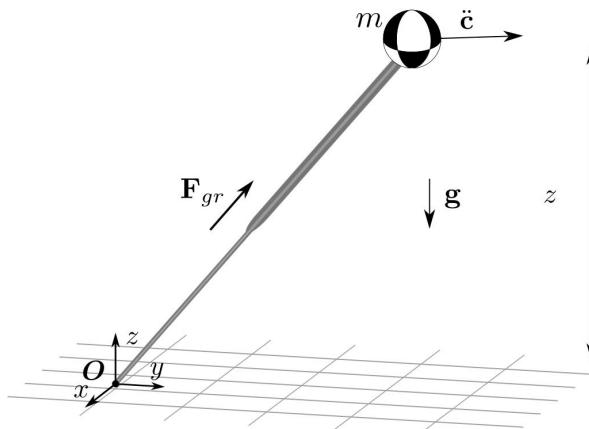


Figure 2-4: The 3D motion of a VHIP model.

where $\mathbf{F}_{gr,2D} = [F_{gr,x}, F_{gr,z}]^T$ is the GRF of the 2D model. Work that uses this equation are, for example, [10] and [13].

Another way of writing the dynamics of the VHIP is as a function of the vertical acceleration \ddot{z} :

$$\ddot{\mathbf{c}} = \frac{g + \ddot{z}}{z} \mathbf{c} + \mathbf{g}. \quad (2-6)$$

This dynamical equation can be seen as a linear time-varying system. Examples of work that uses the latter dynamical description for the VHIP are [12] and [14]. Note that the two models are identical in 2D.

2-1-4 Orbital Energy & the Capture Problem

An advantage of the LIP is that closed-form solutions to the dynamics exist. The LIP orbital energy is an example of such a closed-form solution. This energy can be used to determine the ability of the pendulum to converge to its unstable equilibrium: the capture problem [4], [5].

Linear Inverted Pendulum Orbital Energy

The LIP orbital energy is originally derived in [3] and shows one of the main advantages of the use of a LIP model. This energy reads as follows:

$$E_{LIP} = \int (\ddot{x} - \frac{g}{z_0} x) \dot{x} dt = \frac{1}{2} \dot{x}^2 - \frac{g}{2z_0} x^2, \quad (2-7)$$

where E_{LIP} is the LIP orbital energy. This orbital energy is a conserved quantity if no contact change occurs. Note that the expression resembles kinetic and potential energy: one term depends on velocity, the other on position. If $E_{LIP} > 0$, the point-mass will cross the horizontal position of the pendulum base with its current velocity. If $E_{LIP} < 0$, the point-mass will not cross the pendulum base and will have a turning point where the velocity changes direction.

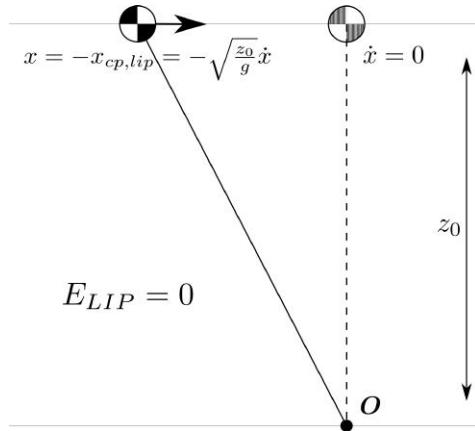


Figure 2-5: Visualization of the path to the LIPCP.

Capture Point & Capture Region

More than a decade later than the first mention of the LIP orbital energy, the LIP capture point (LIPCP) was presented in [4]. Setting $E_{LIP} = 0$ and taking the square root of Equation (2-7) gives:

$$x_{cp,lip} = \sqrt{\frac{z_0}{g}} \dot{x}, \quad (2-8)$$

where $x_{cp,lip}$ is the ‘capture point’, in this thesis referred to as the LIPCP. If the CoP is held constant at the LIPCP, the velocity of the point-mass will be exactly driven to zero when it is above the CoP. In Figure 2-5, a 2D visual explanation is given of this point. Because of the LIP model, the trajectory is at a constant height z_0 . The LIPCP will be used for comparison with the novel VHIP capture regions presented later in this report.

A capture region will traditionally appear if the LIP plus flywheel model is used [4, 21, 5]. As the control authority over the flywheel can change the CoM dynamics, a set of capture points become reachable, which is referred to as the capture region in [4].

Instantaneous Capture Point

The instantaneous capture point (ICP) was introduced in [5], which gives a slightly different description of the LIPCP:

$$\xi = \mathbf{c}_{xy} + \sqrt{\frac{z_0}{g}} \dot{\mathbf{c}}_{xy}, \quad (2-9)$$

where $\xi = [\xi_x, \xi_y]^T$ is the ICP. In this way, the LIPCP is written in environmental coordinates and can be seen as a point where to step in the environment to come to a stop. Other expressions similar to the ICP are the extrapolated center of mass [6] and the divergent component of motion (DCM) [7].

For planning and control, the time derivative is often taken of the ICP: the ICP dynamics [5]. This time derivative can be written as a function of the current ICP location and a ground reference point:

$$\dot{\xi} = \sqrt{\frac{g}{z_0}} (\xi - \mathbf{r}), \quad (2-10)$$

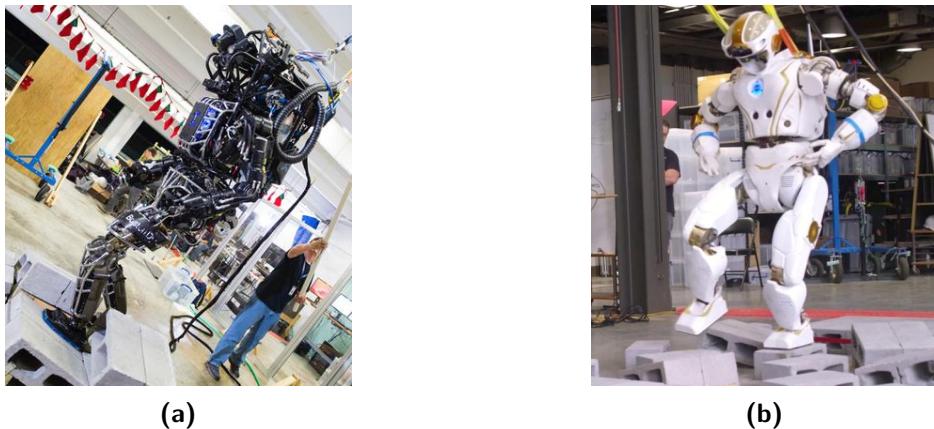


Figure 2-6: (a) Atlas [28] and (b) Valkyrie [29] walking over an uneven cinder block field at IHMC.

where $\dot{\xi}$ is the ICP velocity and \mathbf{r} is a ground reference point, depending on modeling choices as discussed in Section 2-1-2. If angular momentum is included, the CMP is often used [24] and if not, the CoP or ZMP can be used [25].

Orbital Energy with Height Variation

Allowing height variation of the CoM, an expression for orbital energy is more difficult to derive than its linear counterpart. Examples of attempts to include CoM height variations in the solution to the dynamics are the time-varying DCM [12], the VHIP orbital energy under a virtual constraint [10] and the height varying boundedness condition [14]. This work is discussed in Section 2-3, since they are highly related to the research of focus.

2-2 Humanoid Robotics at IHMC

To support the methods and results presented later in this thesis, this section presents a brief background on humanoid robotics at IHMC. Most algorithms are written in Java and simulations are run in Simulation Construction Set (SCS) [26].

2-2-1 Robots

There are two humanoid robots present at the institute at the moment of writing: Boston Dynamics' Atlas [8, 9] and NASA's Valkyrie [27]. An important difference between the two robots is that Atlas is hydraulically actuated, while Valkyrie relies on electric series elastic actuators. The actuation of Atlas is in general more powerful, which allows the robot to take higher steps. Valkyrie on the other hand has more precise torque sensing, which is important for torque control on the robot. This ties to a similarity between the robots: both robots are torque controlled. Using a control framework, the actuators of the robots can be controlled using measured and reference torques. In Figure 2-6, the two robots are shown.

2-2-2 Planning

In contrary to running, with walking there is a state in every cycle with two feet in contact with the ground. This is the double support state and the state where only one leg is on the ground is the single support state. Additionally, other states within either the double support or single support state are considered, like toe-off in the transition from double to single support. In single support, the leg in contact with ground is the support leg and the foot taking a step is the swing leg. The transition between those states and the duration in each state play an important role in the generation of a dynamic plan. Planning of the robots motion is conducted by separating footstep planning from dynamic planning. The dynamic plan in this case is an ICP reference trajectory [24].

Footstep Planning

Footstep planning is the generation of a sequence of footsteps for the robot to follow. A Light Detection and Ranging sensor on the head of the robot provides terrain information. One way to generate a footstep plan is to let the user define each footstep via a graphical user interface [8]. Via relatively simple algorithmic checks on for example kinematic reachability, the user interface can show whether a footstep is feasible or not. To make this process more autonomous, recently developments have been made in the creation of a footstep planner based on an A* search algorithm.

Instantaneous Capture Point Planning

The generation of a dynamic reference plan for the robot is conducted by computing an ICP reference trajectory. This trajectory relies on the solution to the linear differential equation of the ICP dynamics of Equation (2-10):

$$\xi(t) = e^{\omega_0 t} (\xi_0 - \mathbf{r}_0) + \mathbf{r}_0, \quad (2-11)$$

where $\omega_0 = \sqrt{\frac{g}{z_0}}$ is the natural frequency of the LIP and ξ_0 and \mathbf{r}_0 are the initial ICP and ground reference point location, also called the *knot-points*. This equation assumes that the location of the ground reference point is constant.

Under the assumption of constant ground reference point locations, the ground reference knot-points, multiple methods have been developed over the years and improvements are still being made. The most traditional ICP reference trajectory is calculated with a single ZMP knot-point for each footstep [25]. For each ZMP knot-point, an ICP knot-point is computed by integrating the ICP dynamics backwards in time from the last footstep to the first. Using Equation (2-11), the local ICP reference value can be computed at any time instance within the plan.

The above mentioned method is extended in [30], where multiple CMP knot-points per foothold are considered and single support and double support transitions are interpolated using splines. In the most recent improvements, continuous CMP reference trajectories are used for the generation of the ICP plan [24]. An estimate of the angular momentum generated during the walking motion is incorporated in the generation of the CMP reference trajectory. At the time of writing, the latter method is the one currently in use at IHMC, and which is used for the experiments in Chapter 6.

2-2-3 Instantaneous Capture Point Control

Based on a CMP and ICP reference trajectory, the following proportional control law is used to generate a desired CMP [8]:

$$\mathbf{r}_{cmp,d} = \mathbf{r}_{cmp,r} + \mathbf{k}_\xi (\boldsymbol{\xi} - \boldsymbol{\xi}_r), \quad (2-12)$$

where $\mathbf{r}_{cmp,d}$ is the desired CMP, $\mathbf{r}_{cmp,r}$ and $\boldsymbol{\xi}_r$ are the reference CMP and ICP from the ICP planner respectively, $\boldsymbol{\xi}$ is the estimated ICP and \mathbf{k}_ξ is the ICP gain. From $\mathbf{r}_{cmp,d}$, the desired horizontal linear momentum rate of change is computed:

$$\dot{\mathbf{i}}_{d,xy} = \frac{\mathbf{c}_{xy} - \mathbf{r}_{cmp,d}}{z_0} mg, \quad (2-13)$$

where $\dot{\mathbf{i}}_{d,xy}$ is the desired horizontal linear momentum rate of change, which is the desired horizontal force on the CoM of the robot. Note that this value is computed based on the LIP equations of motion. This value is sent to the whole-body quadratic program (QP).

2-2-4 Momentum-Based Whole-Body Control

The desired horizontal linear momentum rate of change $\dot{\mathbf{i}}_{d,xy}$, the output of ICP control, is one of the inputs for the the whole-body QP. The whole-body QP finds desired joint accelerations and desired GRFs, which are translated to desired joint torques by a Newton-Euler inverse dynamics algorithm.

Centroidal Dynamics

The constraint on the dynamics of the robot in the whole-body QP is based on centroidal dynamics [31]. Centroidal dynamics describe the dynamics on and about the CoM of the robot as a result of external forces like gravity and GRF. To explain the CoM dynamics of the robotic chain of the humanoid, a short introduction is given to joint to end-effector mapping. The mapping between joint velocities and end-effector motion plays a crucial role in any robotic system:

$$\mathbf{T} = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v} \end{bmatrix} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \in \mathbb{R}^6, \quad (2-14)$$

where $\dot{\mathbf{q}}$ are the joint velocities, $\mathbf{J}(\mathbf{q})$ is the Jacobian that maps joint velocities to the end-effector twist \mathbf{T} . The twist consist of the angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$ and the linear velocity $\mathbf{v} \in \mathbb{R}^3$. A basis for momentum-based whole-body control is the use of centroidal momentum:

$$\mathbf{h} = \begin{bmatrix} \mathbf{k} \\ \mathbf{l} \end{bmatrix} = \mathbf{A}(\mathbf{q})\dot{\mathbf{q}} \in \mathbb{R}^6, \quad (2-15)$$

where $\mathbf{A} = \mathbf{I}\mathbf{J}$, as introduced in [31], is the inertia matrix \mathbf{I} times the Jacobian. The centroidal momentum \mathbf{h} consists of the angular part $\mathbf{k} \in \mathbb{R}^3$ and the linear part $\mathbf{l} \in \mathbb{R}^3$. The time derivative of the centroidal momentum, the centroidal momentum rate of change, is the constraint on the dynamics of the robot in the QP currently in use [8]:

$$\dot{\mathbf{h}} = \begin{bmatrix} \dot{\mathbf{k}} \\ \dot{\mathbf{l}} \end{bmatrix} = \mathbf{A}\ddot{\mathbf{q}} + \dot{\mathbf{A}}\dot{\mathbf{q}} = \mathbf{W}_g + \sum_i \mathbf{W}_{gr,i} + \sum_i \mathbf{W}_{ext,i}, \quad (2-16)$$

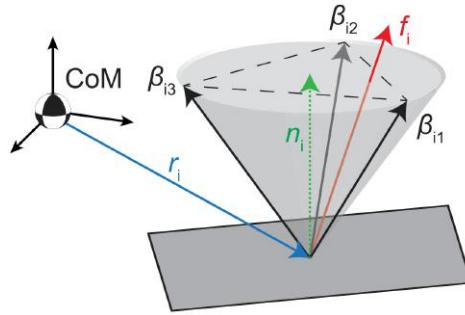


Figure 2-7: Approximation of the wrench cone with basis vectors β_{ij} for ground contact point i . The linear part of the ground reaction wrench, f_i in the drawing, is a positive multiplication of the basis vectors and lies inside the wrench cone. Adopted from [8].

where \mathbf{W}_g is the gravitational wrench and $\sum_i \mathbf{W}_{gr,i}$ the wrench exerted by the total GRF, as a sum of the wrenches of each ground contact point considered. The other external wrenches $\sum_i \mathbf{W}_{ext,i}$ are caused by other contacts than ground. These are considered zero in this thesis: $\sum_i \mathbf{W}_{ext,i} = \mathbf{0}$.

Whole-Body Quadratic Program

The whole-body QP [8] optimizes between momentum rate objectives and motion objectives to find desired joint accelerations and desired GRF. The optimization is formulated as follows:

$$\begin{aligned} \min_{\ddot{\mathbf{q}}_d, \boldsymbol{\rho}} \quad & J_{\dot{\mathbf{h}}_d} + J_{\mathbf{J}_m} + J_{\boldsymbol{\rho}} + J_{\ddot{\mathbf{q}}_d}, \\ \text{s.t.} \quad & \mathbf{A}\ddot{\mathbf{q}}_d + \dot{\mathbf{A}}\dot{\mathbf{q}} = \mathbf{W}_g + \mathbf{Q}\boldsymbol{\rho} + \sum_i \mathbf{W}_{ext,i}, \\ & \boldsymbol{\rho}_{min} \leq \boldsymbol{\rho}, \\ & \ddot{\mathbf{q}}_{min} \leq \ddot{\mathbf{q}}_d \leq \ddot{\mathbf{q}}_{max}, \end{aligned} \quad (2-17)$$

where $\ddot{\mathbf{q}}_d$ are the desired joint accelerations, $\mathbf{Q}\boldsymbol{\rho} = \sum_i \mathbf{W}_{gr,i}$ is the basis vector matrix \mathbf{Q} times the basis vector multiplier $\boldsymbol{\rho}$. The basis vector matrix \mathbf{Q} consists of all basis vectors of the wrench cones from each ground contact point. In Figure 2-7, the wrench cone for a single ground contact point is visually explained. Currently, there are 4 ground contact points considered for each foot of the robot. The minimum $\boldsymbol{\rho}_{min}$ has to be at least zero, because of the unilateral contact constraint; the robot can only push with its feet on the ground. The total cost J is composed of the following cost terms:

$$\begin{aligned} \text{Momentum rate objective: } & J_{\dot{\mathbf{h}}_d} = \mathbf{R}_{\dot{\mathbf{h}}_d} \|\mathbf{A}\ddot{\mathbf{q}}_d + \dot{\mathbf{A}}\dot{\mathbf{q}} - \dot{\mathbf{h}}_d\|^2, \\ \text{Motion objective: } & J_{\mathbf{J}_m} = \mathbf{R}_{\mathbf{J}_m} \|\mathbf{J}_m \ddot{\mathbf{q}}_d - \mathbf{p}\|^2, \\ \text{Contact force regularization: } & J_{\boldsymbol{\rho}} = \mathbf{R}_{\boldsymbol{\rho}} \|\boldsymbol{\rho}\|^2, \\ \text{Joint acceleration regularization: } & J_{\ddot{\mathbf{q}}_d} = \mathbf{R}_{\ddot{\mathbf{q}}_d} \|\ddot{\mathbf{q}}_d\|^2, \end{aligned}$$

where the weighting terms \mathbf{R} can have a selecting function as well. For example, the centroidal momentum rate of change objective $J_{\dot{\mathbf{h}}_d}$ only consists of the linear part and is only affected by the desired linear momentum rate of change $\dot{\mathbf{i}}_d$. The motion task Jacobian $\mathbf{J}_m = [\mathbf{J}_1^T \dots \mathbf{J}_N^T]^T$ consists of all N concatenated Jacobians that map either joint acceleration to end-effector motion, as in Equation (2-14), or joint acceleration to joint acceleration.

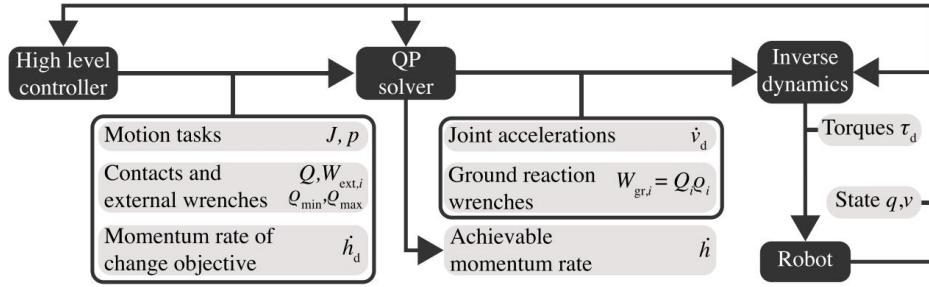


Figure 2-8: High-level overview of the control framework. Adopted from [8].

The motion objective vector $\mathbf{p} = [\mathbf{p}_1^T \dots \mathbf{p}_N^T]^T$ consists of PD controlled desired accelerations, coming for example from trajectory tracking of the swing leg or maintaining the upper body orientation.

An important note considering the research of focus is the generation of vertical CoM motion of the robot. The desired linear momentum rate of change $\dot{\mathbf{i}}_d$ consists, next to its horizontal component of Equation (2-13), of a vertical component:

$$\dot{\mathbf{i}}_{d,z} = m(k_p(z_r - z) - k_d\dot{z}), \quad (2-18)$$

where k_p and k_d are the PD height control gains and z_r is the reference height coming from a reference trajectory. Decision variables for this trajectory are, for example, kinematic reachability and height changes in terrain, but maintaining the robot's balance is *not* a part of those decision variables.

Inverse Dynamics

To translate desired joint accelerations and end-effector wrenches to desired joint torques, a recursive Newton-Euler inverse dynamics algorithm is used. Desired joint torques τ_d are calculated using the solution of whole-body QP, $\ddot{\mathbf{q}}_d$ and ρ , as follows:

$$\tau_d = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{J}^T \mathbf{W}_{gr}. \quad (2-19)$$

These desired joint torques are used by the torque controllers for each actuator.

A high-level overview of the control framework is shown in Figure 2-8. The 'high level controller' block consists, for example, of the ICP controller presented in Section 2-2-3 and the height control law of Equation (2-18).

2-3 Related Work

In this short literature survey, the scope is not limited to only the use of vertical CoM motion in balance control. Other goals of improvement, like improving dynamic planning for motions over rough terrain or for more human-like motions, are discussed as well. The methods used for dynamic planning can be closely related to model predictive control (MPC) and can be insightful for the problem considered in this thesis.

Traditionally, vertical CoM motion is generated through PD control as in Equation (2-18) [32, 8]. A dynamic reference plan exists, often based on the LIP model, and height variations are considered as disturbances on the model used in the dynamical plan. Reasons to use height variations here include the guarantee of kinematic feasibility: height variation allows the robot to step up platforms, and allows the robot to take larger steps. A noteworthy, more unique, example of height variation in non-predictive control is walking with straighter legs as in [33]. The motivation in this work is to let the robot walk more human-like, which could have more underlying benefits, such as kinematic reachability and metabolic energy consumption [34].

2-3-1 Dynamic Planning

Because the constant height assumption of the LIP is constraining the dynamics of the robotic system, efforts have been made to incorporate CoM height variation in the generation of a dynamic plan. Instead of using a LIP model, a more complicated model is used. Expected height variations of the CoM can be incorporated in the dynamic planning problem, which improves the dynamic feasibility of the plan. In theory, the reference dynamics are closer to the real dynamics of the robot. Deviations from the LIP in the CoM reference trajectory can be incorporated in the plan. These deviations can come, for example, from an uneven terrain or a human-like walking pattern.

An example of incorporating height variations in terrain in the dynamic planning problem can be found in [11], which is an extension of ICP planning as in Section 2-2-2. Additional reference points, similar to ground reference points as in Section 2-1-2, are designed and used in the planning method. The drawback of this method is that still a linearized model is considered and the trajectories between footsteps for the dynamical plan are constrained to be straight lines.

In [12], this latter aspect is improved by introducing the time-varying DCM. The natural frequency of the LIP was made time-varying, such that the DCM became time-varying. However, a closed-form solution using this method was not available anymore, requiring the dynamic plan to be computed numerically.

The methods presented in [35] and [36] also show the objectives of walking with straighter legs in the dynamic planning problem. The objective in the optimization in [35] is to let the robot walk with the straightest legs as possible at any time. In [36], a 2D walking pattern is generated for walking with straighter legs. The third dimension is added, under the assumption that the dynamics in the sagittal plane and the coronal plane can be decoupled.

2-3-2 Balance Control

Instead of coping with height variations in the dynamic planning problem, the use of CoM height variation was recently considered as an input for online disturbance rejection. Unlike dynamic planning, in which trajectories are computed in advance, with balance control a control input needs to be computed within the control rate. When going from the LIP to the VHIP, the problem arises of losing the explicit solution to the dynamics. Therefore, especially in MPC, computation times are a challenge, as reference trajectories have to be replanned every control tick.

In this section, the following three publications that consider CoM height variation as control input for balance are discussed. The work proposed by: Koolen, Posa & Tedrake [13], Gao, Jia & Fu [37] and Caron & Mallein [14].

Koolen et al. [13] propose a 2D MPC law, based on the VHIP orbital energy proposed in [10]. This energy reads as follows:

$$E_{VHIP} = \frac{1}{2} \dot{x}^2 \bar{f}^2(x) + g x^2 f(x) - 3g \int_{x_f}^x f(\xi) \xi d\xi = \frac{1}{2} \dot{x}_f^2 \bar{f}^2(x_f) + g x_f^2 f(x_f), \quad (2-20)$$

where E_{VHIP} is the VHIP orbital energy under a virtual constraint. The virtual constraint $z = f(x)$ is used to make a closed-form solution possible for the energy. Furthermore, $\bar{f}(x) = f(x) - f'(x)x$. Unlike its LIP cousin, this 2D orbital energy allows for CoM height variation. Note that filling in a constant value for the function, $f(x) = z_0$, rewrites to the LIP orbital energy.

The function $f(x)$ is constrained to be a cubic polynomial by Koolen et al. [13]. Using this description, four constraints are presented, which are used in a matrix to solve for the polynomial constants. There is one constraint on the final height, one constraint on the initial height, one on the initial direction and one constraint on conservation of E_{VHIP} . The final position of the polynomial trajectory is above the CoP and the final velocity is zero, such that the resulting polynomial trajectory is a capture trajectory: the trajectory that results in the system to come to a stop.

Comparable with the LIP-based ‘capture regions’ in [4, 5] and ‘stable regions’ in [21], Koolen et al. investigate ‘regions of attraction’ for the MPC law. Traditional ‘capture regions’ consider inertia about the CoM to control the LIP. The VHIP ‘regions of attraction’ use CoM height variation as a control input. The ‘regions of attraction’ show an interesting insight in capture regions of the VHIP model. By constraining the height trajectory to be a polynomial function, the MPC law has an explicit solution. However, this comes at the cost of the trajectories, and thus also the control inputs, to be constrained to the polynomial shape. Therefore, also the regions of attraction are constrained to the polynomial shape, which can affect the regions. Furthermore, no kinematic constraints, such as a minimum and maximum height, are considered in the problem. Therefore, the control law is not implementable on a real robot without modification.

Gao et al. [37] present different 2D multi-step strategies to use vertical motion in balance control. An example is lowering the CoM height in the current step, to exert more force and raise the CoM in the next step. In this way, the pendulum can stop closer to the current position than with a LIP trajectory. To account for the vertical CoM motion, the natural frequency of the LIP is adjusted with the added vertical acceleration. To make the dynamical model closed-form solvable, a constant height is considered, as the deviations from the initial height are considered to be relatively small.

Caron & Mallein [14] propose a 3D MPC law, based on a height varying version of the boundedness condition from [38]. In [38], the boundedness condition is based on the LIP and presents a similar expression as the ICP, but takes a time-varying ground reference point trajectory into account. By using a time varying natural frequency of the pendulum $\omega(t)$, Caron & Mallein [14] combine the boundedness condition with the time-varying DCM of [12]. By the nonlinearity of the problem, it is initially hard to solve real-time. The problem

function is reformulated and written as a function of an inverse of time, rather than as a function of forward time. Also, the capture trajectory is divided in 10 segments, which are considered to have a piece-wise time-invariant natural frequency. The resulting problem is solved using a nonlinear solver.

Even though the work presented in [14] is the closest to application (3D control law, kinematic and force constraints), no applied results are shown. Additionally, for the presented control laws, no comparison is made with a traditional constant height control law. Therefore, the question remains what the advantages are of the presented MPC law compared to using a LIP-based proportional controller for example. In Chapter 5 and Chapter 6 of this report, CoM height variations in balance control are compared with constant height approaches on humanoid robots.

Comparable with the ‘regions of attraction’ [13], capture regions are presented for the VHIP model in the next chapter. Like in the approach presented in [14], constraints on the kinematics and dynamics of the system are approximated with constraints on vertical position and vertical acceleration.

Chapter 3

Variable Height Inverted Pendulum Capture Regions

Using simple models for a walking humanoid robot, it has been a common approach to study point-foot placements where it is possible to come to a stop [4, 21, 5]. Such a point-foot location is referred to as a capture point in [4]. In the case of the linear inverted pendulum (LIP) model, only one capture point exists: the LIP capture point (LIPCP). If a torque controlled flywheel is added to this model, a control input is available and reaching a set of capture points will become possible. This set of capture points is referred to as a capture region, also called a stability region in [21]. A control input however can also become available without addition of a flywheel to the LIP.

Taking away the height constraint of the LIP, results in the variable height inverted pendulum (VHIP). Height variations can be used to influence the capture region. Equation (2-6), written in two-dimensional space (2D), reads as follows:

$$\ddot{x} = \frac{g + \ddot{z}}{z} x. \quad (3-1)$$

In this chapter, capture regions are derived by adding constraints to the model above. Recovery is considered within the current step, which is equivalent to ‘0-step’ capture [5]. A capture point is denoted as a positive value, while the initial state of the above model is set to $x_0 < 0$, $z_0 > 0$ and $\dot{x}_0 > 0$. Thus, if x_0 is a capture point, this is denoted as $x_{cp} = -x_0$. For comparison with the LIP and for simplicity, the initial vertical velocity is set to zero: $\dot{z}_0 = 0$.

3-1 Unilateral Contact Constraint

The assumption is made that the robot can only push on the ground with its legs. This introduces a constraint of contact unilaterality. The downward acceleration of the point-mass is constrained by gravity acceleration, and the constraint $\ddot{z} \geq -g$ is added to the VHIP model of Equation (3-1).

Neglecting torque limits, the velocity of the point-mass can be instantly changed by an impact of the VHIP. Writing Equation (3-1) in terms of an instant velocity change gives:

$$\delta\dot{x}_I = \frac{\delta\dot{z}_I}{z} x, \quad (3-2)$$

where $\delta\dot{x}_I$ and $\delta\dot{z}_I$ are the horizontal and vertical velocity change, resulting from an impact by the VHIP.

To find the first bound on the unilateral contact constrained capture region, the closest possible capture point is explored. Using Equation (3-2), if an initial impact of the leg directly stops the horizontal motion of the mass, $\dot{x}_0 + \delta\dot{x}_I$ is zero. Using this, the first bound on the region is found by considering an infinite impact of the leg:

Unilateral Contact Constrained First Bound

$$\lim_{\delta\dot{z}_I \rightarrow \infty} x_{cp,uni,1} = \frac{z_0}{\delta\dot{z}_I} \dot{x}_0 = 0, \quad (3-3)$$

where $x_{cp,uni,1}$ is the first bound on the unilateral contact constrained capture region. When, in the horizontal plane, the point-mass approaching the base is infinitely close to the base, an infinite vertical impact by the VHIP is needed to stop the horizontal motion.

For the second bound on the region, the minimum possible vertical acceleration of $\ddot{z} = -g$ is used. With this vertical acceleration, the dynamics of the point-mass depend on gravitational forces only: the point-mass follows a *ballistic* trajectory. After this ballistic trajectory, the mass can be stopped by an impact of the VHIP just before $z = 0$. The time after which the point-mass touches the ground is:

$$t_{bal} = \sqrt{\frac{2z_0}{g}}. \quad (3-4)$$

The horizontal location at this time is:

$$x_{bal} = \dot{x}_0 t_{bal} = \dot{x}_0 \sqrt{\frac{2z_0}{g}}, \quad (3-5)$$

where x_{bal} is the ballistic touchdown point. This point is the second bound on the unilateral contact constrained capture region. In theory, the VHIP can apply an impact just before the point-mass touches the ground, assuming the VHIP is aligned with the velocity of the point-mass at that moment. Also, this point has a special relationship with the LIPCP:

Unilateral Contact Constrained Second Bound

$$x_{cp,uni,2} = x_{bal} = \sqrt{\frac{2z_0}{g}} \dot{x}_0 = \sqrt{2} x_{cp,lip}, \quad (3-6)$$

where $x_{cp,uni,2}$ is the second bound on the unilateral contact constrained capture region.

The capture region spanned between these two bounds reads as follows:

$$x_{cp,uni} \in \left(0, \sqrt{\frac{2z_0}{g}} \dot{x}_0 \right), \quad (3-7)$$

where $x_{cp,uni}$ is a unilateral contact constrained capture position. In Figure 3-1, this region and the LIPCP are visualized. With gray plots, made with the method presented in [13], possible intermediate trajectories are shown. It can be observed how the trajectories can become very high, when approaching the left side of the bound.

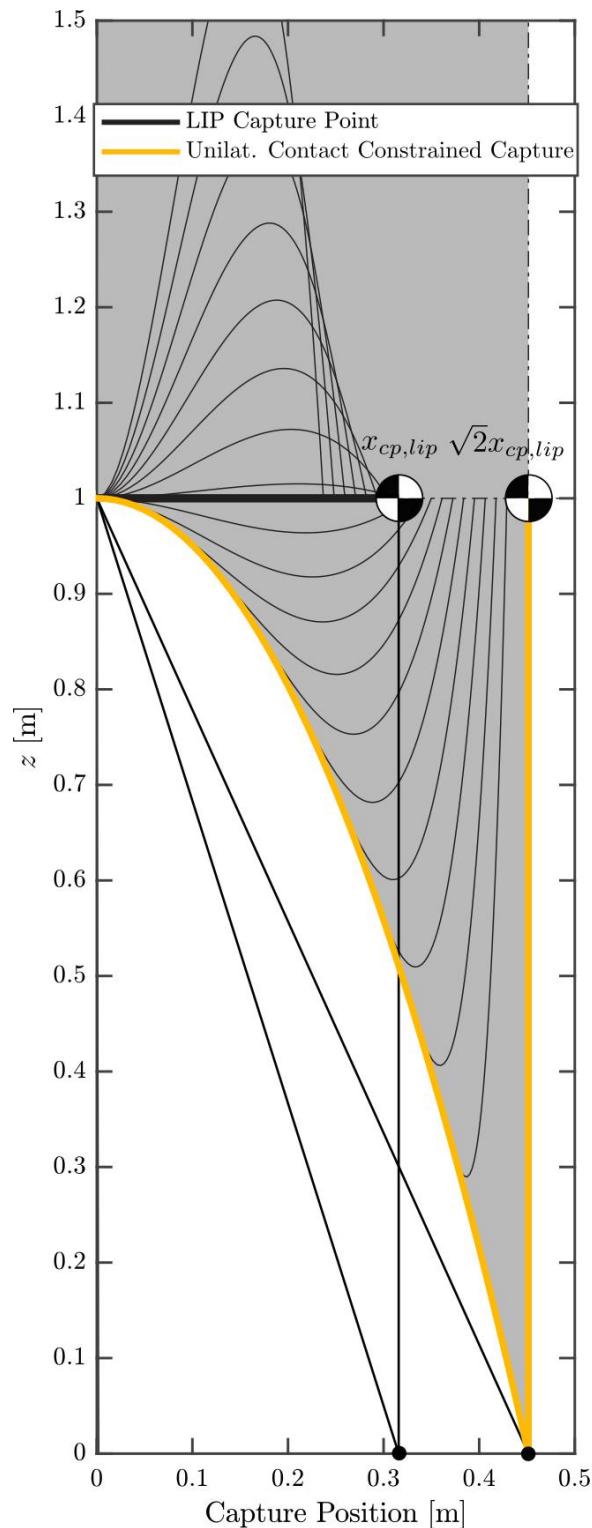


Figure 3-1: Visualization of feasible final point-mass locations for the unilateral contact constrained capture region (gray area). The values $\dot{x}_0 = 1.0$ [m/s], $z_0 = 1.0$ [m], $\dot{z}_0 = 0.0$ [m/s] and $g = 9.81$ [m/s²] are used. The thin black lines are made with the method presented in [13] and visualize possible intermediate trajectories.

3-2 Height Constraints

To further constrain the VHIP model, kinematic constraints are taken into account. Under the assumption that kinematic limits of the robotic system can be approximated with a minimum and maximum center of mass (CoM) height, in this section a minimum and maximum height constraint are added to the unilateral contact constrained capture region of the previous section. The capture points that will be presented are a combination of impacts, ballistic trajectories and LIP capture trajectories, such that closed-form solutions become available.

3-2-1 Maximum Height

With an initial vertical impact by the leg of such magnitude that the resulting apex of the point-mass does not violate the maximum height constraint, a capture position under a maximum height constraint can be derived. This capture point is computed using the velocity change by the vertical impact first, followed by a ballistic trajectory, after which the LIPCP at the maximum height is used.

To calculate the allowed size of the initial impact, the following equality of kinetic and potential energy is used:

$$\frac{1}{2}m\delta\dot{z}_I^2 = mg\delta z_{max}, \quad (3-8)$$

$$\delta\dot{z}_I = \sqrt{2g\delta z_{max}}, \quad (3-9)$$

where $\delta z_{max} = z_{max} - z_0$ is the difference between the current height and the maximum height z_{max} . The initial horizontal velocity is influenced at the moment of the impact as follows:

$$\dot{x}_{0,I} = \dot{x}_0 - \frac{x_{cp,z_{max}}}{z_0}\delta\dot{z}_I, \quad (3-10)$$

where $\dot{x}_{0,I}$ is the remaining horizontal velocity after the impact and $x_{cp,z_{max}}$ is the capture position following the maximum height constraint, to be determined. Note that at the moment when z reaches z_{max} for the first time after the ballistic trajectory, $\dot{x}_{0,I}$ is unchanged as no virtual leg force is used. The duration of this ballistic trajectory is given by:

$$t_{\dot{z}>0} = \frac{\delta\dot{z}_I}{g}, \quad (3-11)$$

where $t_{\dot{z}>0}$ is the time that the height velocity is greater than zero. The capture position under the maximum height constraint is calculated as follows:

$$x_{cp,z_{max}} = \left(t_{\dot{z}>0} + \sqrt{\frac{z_{max}}{g}} \right) \dot{x}_{0,I}, \quad (3-12)$$

$$= \left(\frac{\delta\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}} \right) \left(\dot{x}_0 - \frac{x_{cp,z_{max}}}{z_0} \delta\dot{z}_I \right). \quad (3-13)$$

Taking $x_{cp,z_{max}}$ to the left-hand side leads to:

$$\left(1 + \left(\frac{\delta\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}}\right) \frac{\delta\dot{z}_I}{z_0}\right) x_{cp,z_{max}} = \left(\frac{\delta\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}}\right) \dot{x}_0, \quad (3-14)$$

$$x_{cp,z_{max}} = \frac{\left(\frac{\delta\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}}\right)}{1 + \left(\frac{\delta\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}}\right) \frac{\delta\dot{z}_I}{z_0}} \dot{x}_0. \quad (3-15)$$

Finally, writing this point in terms of the initial state gives:

$$x_{cp,z_{max}} = \frac{\frac{\sqrt{2g\delta z_{max}}}{g} + \sqrt{\frac{z_{max}}{g}}}{1 + \left(\frac{\sqrt{2g\delta z_{max}}}{g} + \sqrt{\frac{z_{max}}{g}}\right) \frac{\sqrt{2g\delta z_{max}}}{z_0}} \dot{x}_0. \quad (3-16)$$

Simplifying leads to:

Maximum Height Constrained Capture Point

$$x_{cp,z_{max}} = \frac{z_0(\sqrt{2\delta z_{max}} + \sqrt{z_{max}})}{\sqrt{g}(z_0 + 2\delta z_{max} + \sqrt{2z_{max}\delta z_{max}})} \dot{x}_0. \quad (3-17)$$

3-2-2 Minimum Height

First the impact influenced capture point is presented, which is used to formulate a capture point following the minimum height constraint.

Impact Influenced Capture Point

Temporarily, the constraint on zero initial vertical velocity is neglected and the initial vertical velocity is set to a negative value. Under the assumption that this vertical velocity is directly driven to zero after an impact of the leg, the resulting capture position is computed as follows:

$$x_{cp,I} = \sqrt{\frac{z}{g}}(\dot{x}_0 + \delta\dot{x}_I), \quad (3-18)$$

where $x_{cp,I}$ is the impact influenced capture point. Writing the added velocity $\delta\dot{x}_I$ in terms of added vertical velocity gives:

$$x_{cp,I} = \sqrt{\frac{z_0}{g}}\left(\dot{x}_0 - \frac{x_{cp,I}}{z_0}\delta\dot{z}_I\right). \quad (3-19)$$

Under the assumption that the vertical velocity is driven to zero instantaneously by the impact, $\delta\dot{z}_I = -\dot{z}_0$. Taking $x_{cp,I}$ to the left-hand side leads to:

$$x_{cp,I} = \frac{\sqrt{\frac{z_0}{g}}}{1 + \frac{\delta\dot{z}_I}{z_0}\sqrt{\frac{z_0}{g}}} \dot{x}_0, \quad (3-20)$$

and simplifying gives:

Impact Influenced Capture Point

$$x_{cp,I} = \frac{z_0}{\sqrt{z_0g} - \dot{z}_0} \dot{x}_0. \quad (3-21)$$

Minimum Height Constrained Capture Point

Considering a minimum height constraint on the CoM, a point can be computed that follows a ballistic trajectory initially, after which the impact influenced capture point is computed at the minimum height. Using Equation (3-4), the vertical velocity after a ballistic fall until the minimum height constraint reads as follows:

$$\dot{z}_{z_{min}} = -gt_{bal,z_{min}} = -g\sqrt{\frac{2\delta z_{min}}{g}} = -\sqrt{2g\delta z_{min}}, \quad (3-22)$$

where $t_{bal,z_{min}}$ and $\dot{z}_{z_{min}}$ are the time and vertical velocity at the minimum height constraint and $\delta z_{min} = z_0 - z_{min}$. The impact influenced capture point of Equation (3-20) after the ballistic fall reads as follows:

$$x_{cp,I}(z_{min}, \dot{z}_{z_{min}}) = \frac{z_{min}}{\sqrt{z_{min}g} + \sqrt{2g\delta z_{min}}} \dot{x}_0. \quad (3-23)$$

The horizontal position after the ballistic fall is $x = \sqrt{\frac{2\delta z_{min}}{g}} \dot{x}_0$. Using this and Equation (3-23), the capture point following the minimum height constraint is computed as follows:

Minimum Height Constrained Capture Point

$$x_{cp,z_{min}} = \left(\sqrt{\frac{2\delta z_{min}}{g}} + \frac{z_{min}}{\sqrt{z_{min}g} + \sqrt{2g\delta z_{min}}} \right) \dot{x}_0, \quad (3-24)$$

where $x_{cp,z_{min}}$ is the capture point following the minimum height constraint.

3-2-3 Bounds on Region

The capture positions $x_{cp,z_{max}}$ and $x_{cp,z_{min}}$ are also the outer bounds on the height constrained capture region.

Lemma 1. *Considering the VHIP dynamics of Equation (2-6), $\dot{z}_0 = 0$, minimum height constraint z_{min} and maximum height constraint z_{max} , $x_{cp,z_{min}}$ and $x_{cp,z_{max}}$ are the outer bounds on the height constrained capture region.*

Proof. For any capture position x_{cp} , $x\dot{x} < 0$ [13] and $0 > x_0 \geq -x_{bal}$ from Equation (3-7). The configuration is used that $x \leq 0, \forall t$ and $x \rightarrow 0$ along any trajectory. From the VHIP dynamical equation, and $z > 0$, it follows that any $\ddot{z} > -g$ will slow \dot{x} down. Showing that $\frac{x}{z} \rightarrow 0, \forall t$ will prove that $\ddot{z} = -g$ for the longest possible time t will lead to the farthest capture point x_{cp} , and a maximum \ddot{z} at the earliest possible t will lead to the closest x_{cp} .

For $\ddot{z} = 0$, z remains constant and $\frac{x}{z} \rightarrow 0$. For $\ddot{z} > 0$, z will grow and $\frac{x}{z} \rightarrow 0$. If $\ddot{z} < 0$, it can be shown with the derivative of $\frac{x}{z}$ that this is always increasing:

$$\frac{d\frac{x}{z}}{dt} = \frac{z\dot{x} - x\dot{z}}{z^2}, \quad (3-25)$$

where $x \leq 0$ and $z\dot{x} \geq 0$. Taking the extreme case $\ddot{z} = -g$ leads to:

$$z\dot{x} - x\dot{z} = (z_0 - \frac{1}{2}gt^2)\dot{x}_0 + (x_0 + \dot{x}_0 t)gt, \quad (3-26)$$

$$= (z_0 + \frac{1}{2}gt^2)\dot{x}_0 + x_0 gt. \quad (3-27)$$

Noting that all terms are positive except for x_0 , which has the largest negative value for $x_0 = -x_{bal}$. Filling this in gives:

$$(z_0 + \frac{1}{2}gt^2)\dot{x}_0 - \sqrt{\frac{2z_0}{g}}\dot{x}_0 gt = \dot{x}_0 \left(\sqrt{\frac{1}{2}gt} - \sqrt{z_0} \right)^2, \quad (3-28)$$

which is always greater than or equal to zero for all t . Therefore, $x_{cp,z_{max}}$ is the first bound and $x_{cp,z_{min}}$ is the second bound on the height constrained capture region. \square

In Figure 3-2, the discussed capture regions are visualized. The LIP capture point lies inside the height constrained capture region, which lies inside the unilateral contact constrained capture region. Again, possible intermediate trajectories are shown with the method presented in [13] and show that the final points are located inside the height constrained capture region, if the same minimum and maximum height constraints z_{min} and z_{max} are considered.

3-3 Vertical Force Constraints

Because the height constrained capture region does not take joint torque limits into account, the VHIP dynamics can be further constrained to narrow the gap between model and robot. Under the assumption that the robot specific limitations on joint torques can be approximated with a minimum and maximum vertical force on the CoM, constraints on the minimum and maximum vertical acceleration are added to the VHIP dynamics. From Lemma 1, any vertical acceleration extremum at the earliest convenience will lead to staying closer to a height constrained bound, which makes the capture region larger. Therefore, a bang-bang control law on vertical acceleration is introduced.

In [4, 21, 5], a bang-bang control law is used to regulate the angular momentum in the body of model. Instead of using this strategy, a bang-bang control law is used to regulate the vertical acceleration:

$$\ddot{z} = \ddot{z}_{c,1}H(t) - (\ddot{z}_{c,1} - \ddot{z}_{c,2})H(t - t_1) - \ddot{z}_{c,2}H(t - t_2), \quad (3-29)$$

where $\ddot{z}_{c,1}$ is the first and $\ddot{z}_{c,2}$ the second constant control input of the bang-bang law and have opposite signs. $H(\cdot)$ is the Heaviside step function and

$$t_1 = \sqrt{\frac{2(z_{const} - z_0)}{\ddot{z}_{c,1} - \frac{\ddot{z}_{c,1}^2}{\ddot{z}_{c,2}}}}, \quad (3-30)$$

which is the solution of:

$$z_0 + \frac{1}{2}\ddot{z}_{c,1}t_1^2 - \frac{1}{2}\frac{(\ddot{z}_{c,1}t_1)^2}{\ddot{z}_{c,2}} = z_{const}, \quad (3-31)$$

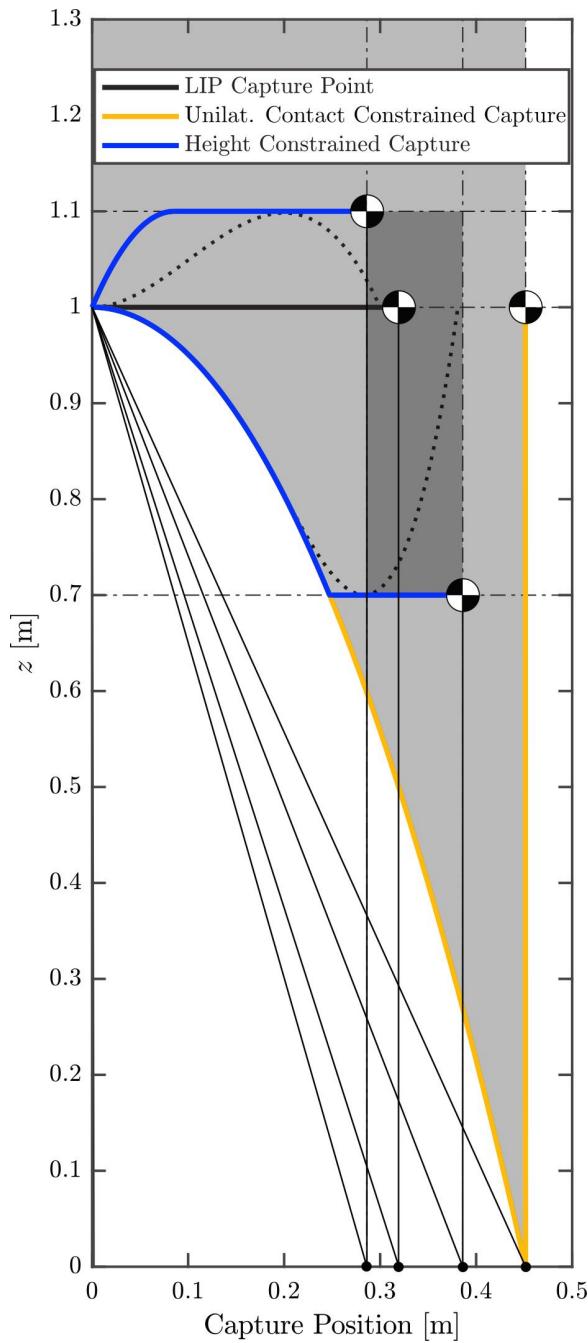


Figure 3-2: Visualization of feasible final point-mass locations for the analytic capture regions. The values $\dot{x}_0 = 1.0 \text{ [m/s]}$, $z_0 = 1.0 \text{ [m]}$, $\dot{z}_0 = 0.0 \text{ [m/s]}$ and $g = 9.81 \text{ [m/s}^2]$ are used. The light gray area shows the feasible final points in the unilateral contact constrained capture region (Section 3-7). The dark gray area shows the feasible final points in the height constrained capture region (Lemma 1) for $0.7 < z < 1.1 \text{ [m]}$. The dotted plots are made with the *orbital energy controller* of [13] and show that the final points are inside the height constrained region.

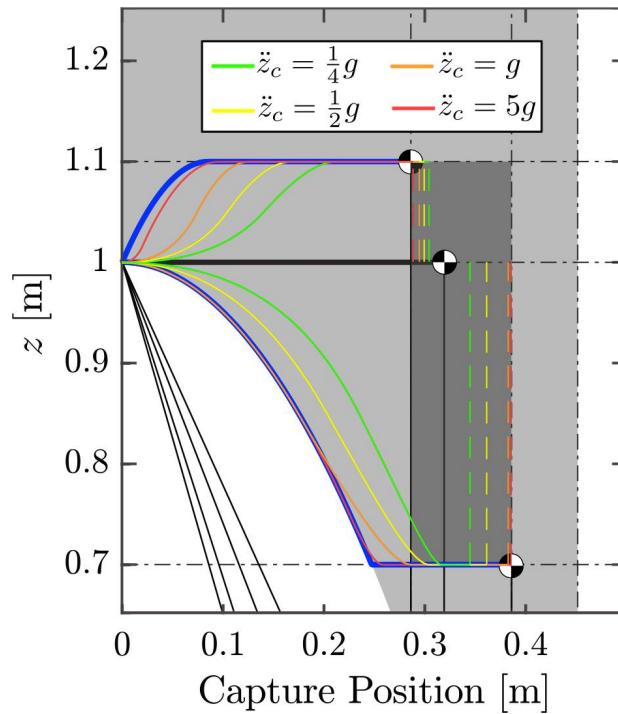


Figure 3-3: Vertical force constrained capture positions relative to the height constrained bounds. The acceleration $\ddot{z}_c = |\ddot{z}_{c,1}| = |\ddot{z}_{c,2}|$ if $\ddot{z}_c \leq g$ and else the negative ‘bang’ is set to $-g$. The values $\dot{x}_0 = 1.0$ [m/s], $z_0 = 1.0$ [m], $\dot{z}_0 = 0.0$ [m/s] and $g = 9.81$ [m/s²] are used.

where $z_{const} = z_{min}$ if $\ddot{z}_{c,1} < 0$ and $z_{const} = z_{max}$ otherwise. The time $t_2 = (1 - \frac{\ddot{z}_{c,1}}{\ddot{z}_{c,2}})t_1$, as the second ‘bang’ needs to drive the vertical velocity resulting from the first bang to zero.

By inserting a constraint on vertical acceleration, the dynamical solution of the VHIP is found via numerical integration. A capture position is found using a optimization similar to a binary search. In this optimization, instead of finding an index, the search is stopped when a specified cost function is below a certain threshold. There is iterated over the initial position, given the initial velocity. The cost function used is x^2 at the time instance that $\dot{x} = 0$. If \dot{x} is never zero, which happens when the point-mass crosses the pendulum base, the cost is set to a high value. The authors of [37] present analytic solutions using vertical acceleration, but consider a constant height in the model. For comparison with applied results later in this thesis, this constant height assumption is not considered.

In Figure 3-3, simulation results are shown in perspective with the earlier presented capture regions. Note that when the bang-bang control inputs \ddot{z}_c are larger, both trajectory and capture position come closer to the height constrained bounds.

3-4 Capturability Comparison

In this section, a comparison is made between the presented capture positions and the LIPCP, as well as a comparison with the LIP plus flywheel capture regions. As in [4, 21, 5], a dimensional analysis [39] is performed when comparing the capture regions. The following

parameters are used for dimensionless position and velocity:

$$x' = \frac{x}{z_0}, \quad \dot{x}' = \frac{1}{\sqrt{gz_0}} \dot{x}. \quad (3-32)$$

The following dimensionless height and vertical acceleration are used for the VHIP:

$$z' = \frac{z}{z_0}, \quad \ddot{z}' = \frac{\ddot{z}}{g}, \quad (3-33)$$

and the following dimensionless inertia and CoM torque are used for the LIP plus flywheel model:

$$I'_y = \frac{I_y}{mz_0^2}, \quad \tau'_y = \frac{\tau_y}{mgz_0}. \quad (3-34)$$

Instead of the CoM torque, a dimensionless centroidal moment pivot (CMP) offset is used:

$$x'_{cmp} = \frac{x_{cmp}}{z_0} = x'_{cop} + \delta x'_{cmp}, \quad (3-35)$$

$$\delta x'_{cmp} = \tau'_y, \quad (3-36)$$

where $\delta x'_{cmp}$ is the dimensionless CMP offset. The dimensionless center of pressure (CoP) x'_{cop} is assumed to be saturated and at a constant location.

3-4-1 Comparison without Angular Momentum

A comparison is made between the LIPCP, the height constrained capture positions and the vertical force constrained capture positions. The dimensionless vertical accelerations of each ‘bang’ for the vertical force constrained capture positions are set to equal dimensionless magnitude \ddot{z}'_c . For comparison, a rough estimate is made of realistic values of vertical forces that are achievable on both humans and robots.

First, an approximation is made of what would be achievable for a human being. A human jumping vertically with maximum effort generates approximately $2mg$ ground reaction force [40]. If the assumption is made that this value can also be used in recovery, the value $\ddot{z}'_c = 1$ can be taken for a human. Second, an approximation is made of what is possible on robots. On hardware experiments on NASA’s Valkyrie in Chapter 6 is found that $\ddot{z}'_c = \frac{1}{4}$ is near the maximum what is applicable on Valkyrie. Larger accelerations would result in the robot to shake and did not improve recovery. On hardware experiments on Boston Dynamics’ Atlas, accelerations of approximately $\ddot{z}'_c = \frac{1}{2}$ are still applicable.

In the left plot in Figure 3-4, the height constrained bounds are shown, together with the approximations of realistic vertical acceleration constraints on humans and on Valkyrie and Atlas for the vertical force constrained capture positions. The vertical force constrained capture positions are computed for every dimensionless height change $\delta z' = 0.02$. Deviations from $z' = 1$ show which values for the constraints z'_{min} and z'_{max} are used in the computation of the capture points. Note how the capture positions relate differently under a minimum height constraint than under a maximum height constraint. Also note how the capture positions linking to the approximation of allowed vertical acceleration for the robots, seem to approach a minimum and maximum value quite soon after changing height. The point [1, 1] in the plot is the LIPCP.

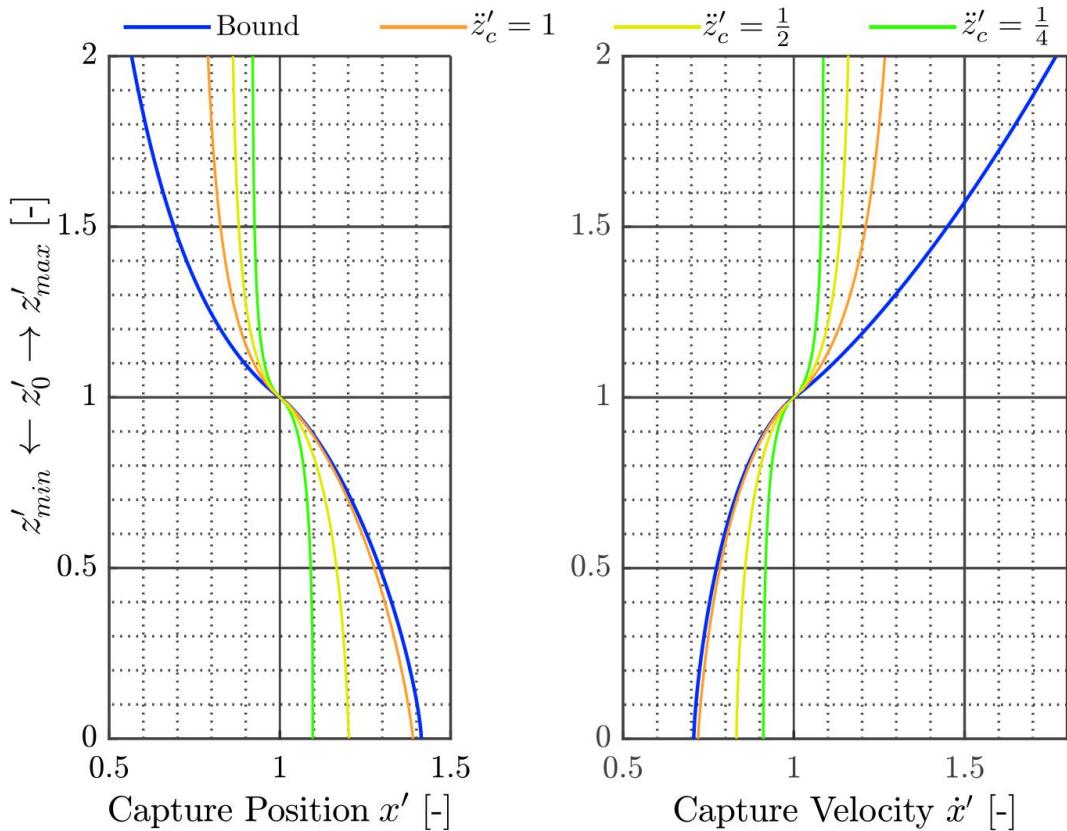


Figure 3-4: Reachable dimensionless capture positions for $\dot{x}'_0 = 1$ (left) and capture velocities for $x'_0 = 1$ (right) without angular momentum. Deviations from $z' = 1$ show which minimum or maximum height constraint is used in the computation of the capture points, $z'_0 = 1$.

In the right plot, the *capture velocity* is shown. This is the initial velocity, after which the resulting capture point is equal to the LIPCP. This is an inverse capture problem: instead of finding a point-foot location where the point-mass would come to a stop, an initial velocity is found that will come to a stop for a fixed point-foot location. The differences between the capture velocities of the regions that consider height variation and the capture velocity of the LIP can be seen as a difference in robustness for an applied disturbance. The capture velocity is used later in this report for comparison with applied results, where push recovery is tested on Valkyrie and Atlas.

3-4-2 Comparison with Angular Momentum

An estimation of the effects of angular momentum strategies can be performed as in [5]. Angular momentum strategies, like a ‘hip strategy’, have a pay back time; any angular velocity generated by the strategy has to be driven back to zero before a physical limit is reached. Therefore, the body torque can be estimated with a bang-bang control law. Considering a LIP plus flywheel model, as proposed in the work mentioned, the following bang-bang control law is used to account for angular momentum:

$$\delta x'_{cmp} = (1 - 2e^{-\delta t'_{bang}} + e^{-2\delta t'_{bang}})\delta x'_{cmp,max}, \quad (3-37)$$

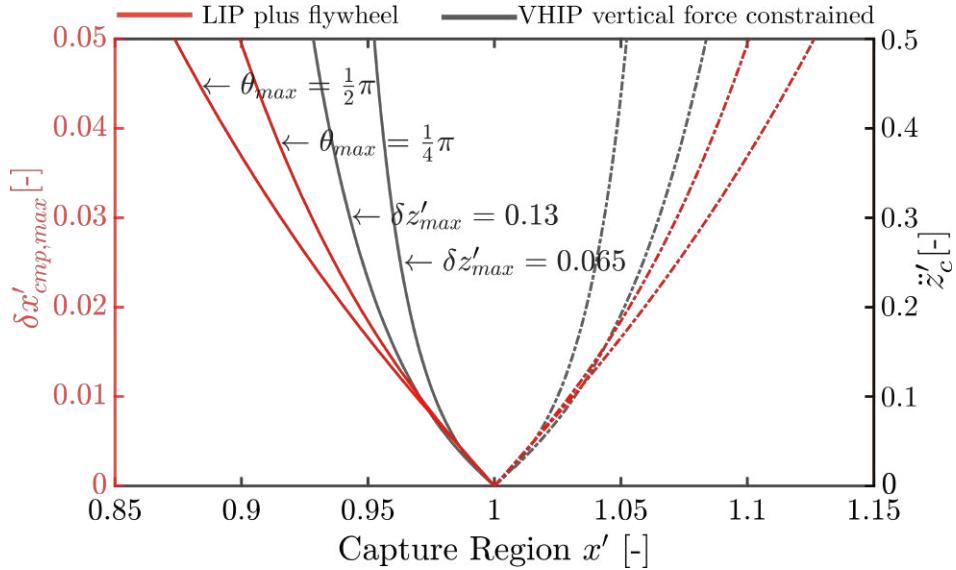


Figure 3-5: Comparison of capture region with angular momentum versus vertical force constrained capture. The height constraints are set equal, $\delta z'_{min} = \delta z'_{max}$ and $I'_y = \frac{1}{6}$. The solid plots show the first bound on the region and the dotted plots the second bound.

where $\delta x'_{cmp,max}$ is the maximum dimensionless offset of the CMP with the CoP and $\delta t'_{bang}$ is the time of each ‘bang’ of the control law. The average dimensionless CMP offset to account for in computation of the LIPCP is $\delta x'_{cmp}$. The time $\delta t'_{bang}$ can be determined by:

$$\delta t'_{bang} = \sqrt{\frac{I'_y \theta_{max}}{\delta x'_{cmp,max}}}, \quad (3-38)$$

where θ_{max} the maximum allowed body lunge relative to the vertical. Note that $I'_y = 0$ for a point-mass and $I'_y = 1$ for a disc with all its mass on the edge, with its radius equal to the CoM height above the ground.

A rough estimate is made of the dimensionless inertia of a standing human or robot used in recovery. The assumption is made that the hip is the CoM position [41] and that the total body length is two times the CoM height. Furthermore, only the body above the hip is assumed to be used in recovery, which is modeled as a beam with half the total body mass rotating around the hip. The inertia of this beam is $I_{beam} = \frac{1}{3}(\frac{1}{2}m)L^2$. Note that the length L is equal to the CoM height z_0 . Those assumptions result in the following dimensionless inertia:

$$I'_y = \frac{\frac{1}{3}(\frac{1}{2}m)z_0^2}{mz_0^2} = \frac{1}{6}. \quad (3-39)$$

Even though the inertia used in recovery is approximated with a calculation based on a beam, still a flywheel model is assumed. Rotation of a beam around its end would change the CoM height of the system and would violate the LIP model.

In Figure 3-5, vertical force constrained capture regions are compared with LIP plus flywheel capture regions. The maximum allowed vertical CoM height change $\delta z'_{max} = 0.065$ is approximately the same value as is used on NASA’s Valkyrie later in this report. The range for $\delta x'_{cmp,max}$ is commonly used at the Institute for Human and Machine Cognition (IHMC) [42].

3-5 Discussion

Chapter 3 presented capture positions and capture regions for the variable height inverted pendulum (VHIP) model, by adding constraints to the VHIP model. By comparing the capture positions, it can be observed that addition of constraints to the model affect the capture region. Comparisons were made with the LIPCP and a high-level comparison is made with angular momentum strategies, which bring CoM height variations for balance in perspective with other balancing strategies.

The unilateral contact and height constrained capture regions consider impacts of the VHIP. This has some resemblance with the work presented in [16], where the influences of impacts on the energetics of the inverted pendulum are investigated. The impact influenced capture point presented in this chapter, shows how an impact influences the LIPCP instead of the inverted pendulum.

For the vertical force constrained capture regions, joint torque limits of the robot are approximated with a constant constraint on minimum and maximum vertical force. This approximation of the torque limit can be closer to the real torque limit if height changes are relatively small, as the change in configuration of the robotic chain is relatively small. For large changes in height, a more complicated function might be worth considering as constraint on vertical force over time. Additionally, it could be interesting to explore closed-form solutions for the vertical force constrained capture points. In this chapter, the points were found numerically. A closed-form expression allows for faster computation times, which could allow for the use in control.

A constraint that was not considered in this chapter is a constraint on ground friction. The unilateral contact constraint constrains the vertical component of the ground reaction force (GRF) to be larger than zero. The vertical force constraint limits the magnitude of the GRF. However, a constraint on the direction of the GRF, like a friction cone constraint, was not considered. As no inertia is used in the model however, the direction of the GRF is just depending on the VHIP configuration.

However, if CoM height variation and angular momentum strategies would be combined, a constraint on ground friction might be worth considering. A VHIP plus flywheel model can be formulated as follows:

$$\ddot{x} = \frac{g + \ddot{z}}{z}x - \frac{1}{mz}\tau_y. \quad (3-40)$$

If the input torque τ_y would be used during a ballistic trajectory, where $\ddot{z} = -g$, the GRF would become parallel to the ground plane. This would require infinite ground friction, which is in practice never the case.

Also, this chapter presented ‘0-step’ capture regions, unlike the work presented in [5, 37], where multiple steps are considered in the computation of the capture region. However, in a similar manner the capture points in this chapter were derived, expressions can be derived for multi-step strategies. Using a combination of impacts, ballistic trajectories and LIP trajectories, a multi-step strategy can be computed closed-form.

In the next chapter, the height constrained capture point z_{max} is used to compute the step size for a model predictive control (MPC) law.

Chapter 4

Kinematically Constrained Orbital Energy Trajectories

The height constrained capture bounds cannot be used directly in control, as they consider impacts with no constraints on force. The force constrained bang-bang control law has to be integrated numerically to obtain future state information. Therefore, it is interesting to explore an additional method that has knowledge over future states, without the drawback of numerical integration. This chapter proposes an extension to the work of Koolen, Posa & Tedrake [13], which relies for an important part on the work by Pratt & Drakunov [10]: the *orbital energy* of the variable height inverted pendulum (VHIP).

4-1 Constraint Matrix with Final Velocity

In [13], the final horizontal velocity \dot{x}_f is set to zero, as this leads to a *capture trajectory*: a trajectory that leads to convergence of the unstable equilibrium of the VHIP. However, the trajectories have no constraints on kinematics and can become unrealistically high above the ground, as shown in Figure 3-1. To take kinematic limits of the system into account, constraints on the configuration of the VHIP have to be added. In this section, the final horizontal velocity of the VHIP orbital energy is left undetermined, which is used to add kinematic constraints to the orbital energy trajectory.

First, the orbital energy is derived for nonzero final velocity. Using the VHIP orbital energy of Equation (2-20), but setting the final horizontal position x_f zero gives:

$$\frac{1}{2}\dot{x}^2\bar{f}^2(x) + gx^2f(x) - 3g \int_0^x f(\xi)\xi d\xi = \frac{1}{2}\dot{x}_f^2\bar{f}^2(0). \quad (4-1)$$

Note that $\bar{f}^2(0) = (f(0) - f'(0) \cdot 0)^2 = f(0)^2$. The additional nonzero term on the right-hand

side can be added in the constraint matrix presented in [13]. The constraint matrix reads as:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & x_0 & x_0^2 & x_0^3 \\ 0 & 1 & 2x_0 & 3x_0^2 \\ \frac{3}{2}gx_0^2 & gx_0^3 & \frac{3}{4}gx_0^4 & \frac{3}{5}gx_0^5 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}}_{\mathbf{c}} = \underbrace{\begin{bmatrix} z_f \\ z_0 \\ \dot{z}_0 \\ k \end{bmatrix}}_{\mathbf{b}}, \quad (4-2)$$

where \mathbf{A} is the constraint matrix, \mathbf{c} is the vector containing the polynomial constants and \mathbf{b} is the constraint vector, giving a constraint on final height, initial height, initial direction and conservation of orbital energy respectively. The parameter k in \mathbf{b} is derived as follows. The integral term in the orbital energy equation, written as a cubic polynomial, read as: $3g \int_0^{x_0} f(\xi) \xi d\xi = 3g \sum_{i=0}^3 \frac{1}{i+2} c_i x_0^{i+2}$. The last constraint of Equation (4-2) reads after addition of the final velocity term as:

$$3g \int_0^{x_0} f(\xi) \xi d\xi = \frac{1}{2} \dot{x}_0^2 \bar{f}^2(x_0) + g x_0^2 f(x_0) - \frac{1}{2} \dot{x}_f^2 \bar{f}^2(0), \quad (4-3)$$

and in terms of the cubic polynomial as:

$$3g \sum_{i=0}^3 \frac{1}{i+2} c_i x_0^{i+2} = \frac{1}{2} (\dot{x}_0 z_0 - \dot{x}_0 x_0)^2 + g x_0^2 z_0 - \frac{1}{2} z_f^2 \dot{x}_f^2, \quad (4-4)$$

$$= k, \quad (4-5)$$

where k is the last entry of vector \mathbf{b} .

Using the polynomial description and having a nonzero final velocity \dot{x}_f , the final height constraint z_f results in a final vertical velocity \dot{z}_f relative to the gradient of the polynomial at $x_f = 0$:

$$f'(0) = \frac{\dot{z}_f}{\dot{x}_f}, \quad (4-6)$$

$$\dot{z}_f = f'(0) \dot{x}_f. \quad (4-7)$$

It could be undesirable that \dot{z}_f is depending on \dot{x}_f . When $\dot{x}_f \neq 0$, additional future actions have to be taken to come to a stop, like taking a step. High vertical velocities can result in, for example, high impacts, which can be undesirable. Therefore, it could be useful to have zero vertical velocity at this point. The constraint $f(0) = z_f$ in the first row in Equation (4-2) can be replaced with the following constraint on the final gradient:

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & x_0 & x_0^2 & x_0^3 \\ 0 & 1 & 2x_0 & 3x_0^2 \\ \frac{3}{2}gx_0^2 & gx_0^3 & \frac{3}{4}gx_0^4 & \frac{3}{5}gx_0^5 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}}_{\mathbf{c}} = \underbrace{\begin{bmatrix} \frac{\dot{z}_f}{\dot{x}_f} \\ z_0 \\ \dot{z}_0 \\ k \end{bmatrix}}_{\mathbf{b}}. \quad (4-8)$$

Note that z_f still appear in k . For the results presented in the following sections, z_f is set equal to the constraint on maximum height or maximum leg length.

4-1-1 Maximum Height Constraint

Using the previous two descriptions of the problem constraints with an undetermined final velocity \dot{x}_f , Equation (4-2) and Equation (4-8), a height constraint is added to the problem. Algorithm 1 shows how the polynomial values can be found under this constraint. The maximum height of the trajectory being optimized over is at location $x_{z_{peak}}$, which is the location in the trajectory where the derivative of the polynomial is zero. The initial guess for the final horizontal velocity is determined based on the linear inverted pendulum (LIP) orbital energy.

In Figure 4-1, two resulting polynomials are shown from Algorithm 1: one with the final height constraint and one for the constraint on the final gradient. Notice that the maximum of $f(x)$ in both plots lies on the highest x -value of the extrema of the polynomial functions. Therefore, in Algorithm 1, the solution corresponding to this peak is used in the computation of the location $x_{z_{peak}}$ of the maximum height in the trajectory. In the figure, the entire polynomial is shown for explanatory reasons. In control however, only the part depicted with a solid line would be used, as this is the part of the function after the initial condition.

Algorithm 1 Find cubic polynomial constants under height constraint

```

1:  $\dot{x}_f \leftarrow \sqrt{\dot{x}_0^2 - \frac{g}{z_0}x_0^2}$                                 ▷ Initial guess based on  $E_{LIP}$ 
2: repeat
3:    $\mathbf{c} \leftarrow \mathbf{A}^{-1}\mathbf{b}(\dot{x}_f)$                                      ▷ Find polynomial constants
4:    $x_{z_{peak}} \leftarrow \frac{-2c_2 - \sqrt{4c_2^2 - 12c_3c_1}}{6c_3}$           ▷ Traj. peak lies on highest  $x$ 
5:    $z_{peak} \leftarrow c_0 + c_1x_{z_{peak}} + c_2x_{z_{peak}}^2 + c_3x_{z_{peak}}^3$     ▷ Corresponding height
6:    $\dot{x}_f \leftarrow \dot{x}_f + \alpha$                                               ▷ Velocity increment
7: until  $z_{peak} < z_{max}$ 
8: return  $\mathbf{c}$ 

```

It is challenging to find a local gradient of the function \dot{x}_f for optimization purposes. Therefore, there is not chosen to determine the size of increment α based on the gradient of \dot{x}_f . Using the final velocity of the LIP orbital energy and a final velocity after the influence of an impact as in $x_{cp,z_{max}}$, an increment can be derived. The final velocity according to the LIP orbital energy is:

$$\dot{x}_{f,lip} = \sqrt{\dot{x}_0^2 - \frac{g}{z_0}x_0^2}, \quad (4-9)$$

where $\dot{x}_{f,lip}$ is the final velocity according to the LIP model. The final velocity of a height constrained bound as in $x_{cp,z_{max}}$ can be calculated as follows:

$$\dot{x}_{f,z_{max}} = \sqrt{\dot{x}_{0,I}^2 - \frac{g}{z_{max}}(x_0 + \frac{\dot{z}_I}{g}\dot{x}_{0,I})^2}, \quad (4-10)$$

where $\dot{x}_{0,I} = \dot{x}_0 + \frac{x_0}{z_0}\dot{z}_I$ is the initial horizontal velocity influenced by the impact by the VHIP and $\dot{z}_I = \sqrt{2g(z_{max} - z_0)}$ is the vertical velocity resulting from the impact of the leg such that the constraint z_{max} is not violated. The velocity $\dot{x}_{f,z_{max}}$ is the final velocity when a similar trajectory as in $x_{cp,z_{max}}$ would be followed. The increment α is calculated by a linear

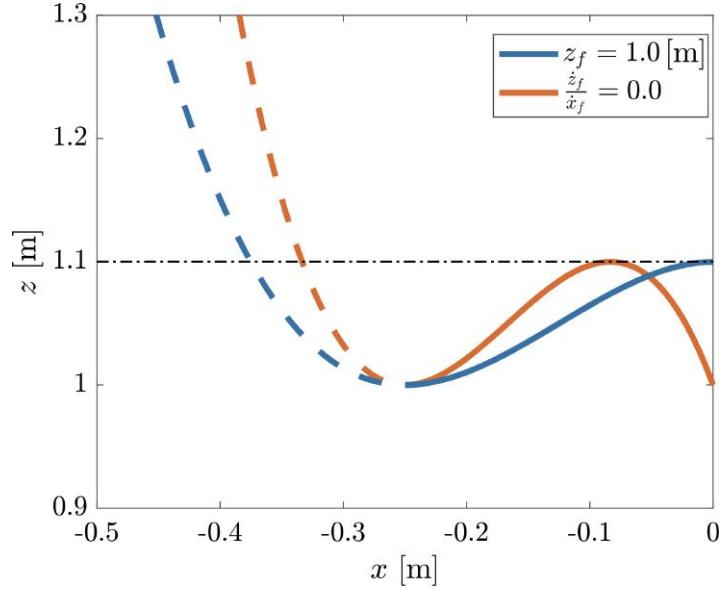


Figure 4-1: Resulting polynomials as output from Algorithm 1 with two different constraints on the final value. The dash-dotted line shows the height constraint $z_{max} = 1.1$ [m]. Initial conditions are $x_0 = -0.25$ [m], $\dot{x}_0 = 1.0$ [m/s], $z_0 = 1.0$ [m] and $\dot{z}_0 = 0$ [m/s]. Blue plot: $\dot{x}_f = 0.552$ [m/s], red plot: $\dot{x}_f = 0.576$ [m/s], $g = 9.81$ [m/s²].

interpolation of the velocity differences:

$$\alpha = \frac{\dot{x}_{f,lip} - \dot{x}_{f,z_{max}}}{N}, \quad (4-11)$$

where the value $N = 30$ resulted in reasonable precision, as shown in for example Figure 4-1. Also, this value resulted in an average computation time of 0.51 [ms] in Java, which could be suitable for a controller running on 250 [Hz] as is used at the Institute for Human and Machine Cognition (IHMC).

4-1-2 Maximum Leg Length Constraint

Instead of a constraint on the height of the VHIP, also a constraint on the length can be added, which can simulate the maximum leg length. The virtual leg length as a function of x can be expressed using the Pythagorean theorem:

$$l^2(x) = f(x)^2 + x^2, \quad (4-12)$$

where l is the length of the VHIP. The solution to $f(x)^2$ is:

$$f(x)^2 = \left(\sum_{n=0}^3 c_n x^n \right)^2 = \sum_{n=0}^3 c_n^2 x^{2n} + 2 \sum_{\substack{i+j=n \\ i < j}}^5 c_i c_j x^n. \quad (4-13)$$

However, as the gradient of $l^2(x)$ needs to be computed to obtain the locations of the maxima, $f(x)^2$ is approximated with:

$$f(x)^2 \approx \sum_{n=0}^3 c_n^2 x^{2n}.$$

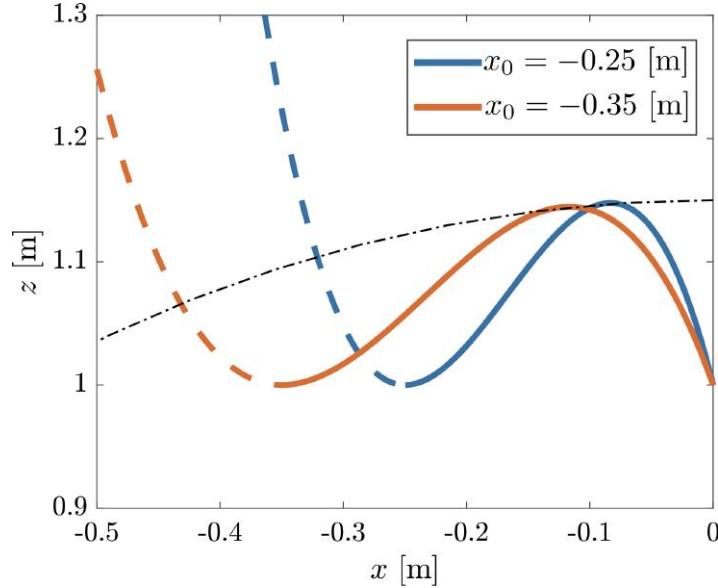


Figure 4-2: Resulting polynomials as output from Algorithm 2 with two different initial values under the final height constraint $z_f = 1.0$. The dash-dotted line shows the length constraint $l_{max} = 1.15$ [m]. Other initial conditions are $\dot{x}_0 = 1.4$ [m/s], $z_0 = 1.0$ [m] and $\dot{z}_0 = 0.0$ [m/s]. Blue plot: $\dot{x}_f = 1.107$ [m/s], red plot: $\dot{x}_f = 0.724$ [m/s], $g = 9.81$ [m/s²].

With this formulation of the squared function, the squared horizontal position $x_{l_{peak}}^2$ in $l(x)^2$ where the approximated maximum pendulum length l_{peak} lies, is computed as follows:

$$\frac{d(f(x_{l_{peak}})^2 + x_{l_{peak}}^2)}{dx} = 0, \quad (4-14)$$

$$\frac{d(f(x_{l_{peak}})^2 + x_{l_{peak}}^2)}{dx} \frac{1}{x_{l_{peak}}} = 0, \quad (4-15)$$

$$6c_3^2x_{l_{peak}}^4 + 4c_2^2x_{l_{peak}}^2 + 2 + 2c_1^2 = 0, \quad (4-16)$$

$$6c_3^2(x_{l_{peak}}^2)^2 + 4c_2^2(x_{l_{peak}}^2) + 2 + 2c_1^2 = 0, \quad (4-17)$$

where $x_{l_{peak}}^2$ is the squared location of the approximated maximum pendulum length l_{peak} .

Again, the value of the peak lies on the highest x value of the two locations where the gradient is zero, this time in $l(x)^2$. Algorithm 2 shows how the polynomial constants can be found under the leg length constraint. In Figure 4-2, it can be seen that the resulting polynomials do not violate the maximum length constraint.

4-2 Discussion

Chapter 4 presented a method for finding VHIP orbital energy trajectories under kinematic constraints, extending the work in [13]. The horizontal velocity of the orbital energy at the final position $x_f = 0$ was left undetermined, such that this could be used as a variable in a search. Choosing the search increment using the knowledge obtained in Chapter 3, constrained trajectories could be generated in a time that is suitable for online use.

Algorithm 2 Find cubic polynomial constants under leg length constraint

```

1:  $\dot{x}_f \leftarrow \sqrt{\dot{x}_0^2 - \frac{g}{z_0} x_0^2}$                                 ▷ Initial guess based on  $E_{LIP}$ 
2: repeat
3:    $\mathbf{c} \leftarrow \mathbf{A}^{-1}\mathbf{b}(\dot{x}_f)$                                          ▷ Find polynomial constants
4:    $x_{l_{peak}}^2 \leftarrow \frac{-4c_2^2 + \sqrt{16c_2^4 - 24c_3^2(2+2c_1^2)}}{12c_3^2}$           ▷  $d(f(x)^2 + x^2)/dx = 0$ 
5:    $x_{l_{peak}} \leftarrow -|\sqrt{x_{l_{peak}}^2}|$                                          ▷ Complex solutions
6:    $l_{peak}^2 \leftarrow x_{l_{peak}}^2 + (c_0 + c_1 x_{l_{peak}} + c_2 x_{l_{peak}}^2 + c_3 x_{l_{peak}}^3)^2$ 
7:    $\dot{x}_f \leftarrow \dot{x}_f + \alpha$                                                  ▷ Velocity increment
8: until  $l_{peak}^2 < l_{max}^2$ 
9: return  $\mathbf{c}$ 

```

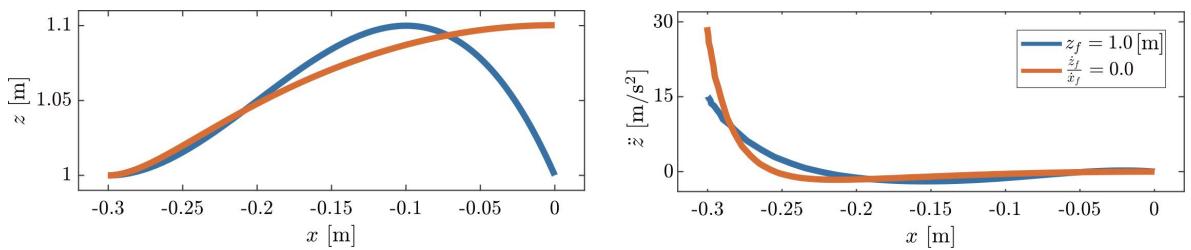


Figure 4-3: Position plot (left) versus the resulting vertical acceleration (right) with $x_0 = -0.3$ [m], $z_0 = 1.0$ [m], $\dot{x}_0 = 1.0$ [m/s], $\dot{z}_0 = 0.0$ [m/s] and $g = 9.81$ [m/s²].

Noticeable is that the final gradient constraint gives a higher final velocity \dot{x}_f compared to the default height constraint in Figure 4-1. This is caused by the polynomial shapes, as the blue plot, under constraint z_f , has a steeper gradient in the first part of the trajectory after x_0 . This is the crucial part in the trajectory for an acceleration or deceleration action in x -direction, as is proven in the previous chapter.

Another drawback of the constraint of the polynomial shape is that derivatives of the function, and thus also accelerations, depend on the function itself. In Figure 4-3, two trajectories are shown for $\dot{x}_f = 0$ [m/s], together with a corresponding plot of the vertical accelerations. The vertical acceleration has a shape that is desirable for a stopping behavior: high acceleration in the beginning of the trajectory and lower later. However, the acceleration in the beginning of the trajectory is unrealistically high, assuming the approximations of possible vertical acceleration on humans and robots in the previous chapter. This is even the case for the initial position that is relatively close to the LIP capture point (LIPCP): $x_0 = -0.3$ [m], as in the example figure. Also, the vertical acceleration barely goes below zero. Therefore, there is no high deceleration in the trajectory, which could result in more height change than for example the vertical acceleration profile of the bang-bang control law for the vertical force constrained capture region presented in the previous chapter.

For those reasons, the methods proposed in this chapter are not used in application in Chapter 5 and Chapter 6.

Chapter 5

Application: Standing

In this chapter, a bang-bang control law on vertical acceleration is implemented in the momentum-based control framework and tested on NASA’s Valkyrie [27] humanoid robot. Push recovery is tested while the robot is standing. To measure the improvement in balance control, there is compared with a controller that uses a constant reference height. Additionally, results are presented using an alternative test setup using Boston Dynamics’ Atlas humanoid robot.

5-1 Method

As the polynomial trajectories of the previous chapter are constraining the vertical acceleration, these are not used in application. Instead, a bang-bang control law for vertical acceleration is designed, similar to the bang-bang control law for the vertical force constrained capture positions in Section 3-3.

The bang-bang control law is activated when the use of center of pressure (CoP) is saturated. This the case if the CoP is on the polygon edge. The desired centroidal moment pivot (CMP) $\mathbf{r}_{cmp,d}$ of Equation (2-12) is constrained to be inside the support polygon. Therefore, it is assumed that the angular momentum rate of change is zero, $\mathbf{k}_y = 0$ [Nm], and that $\mathbf{r}_{cmp,d}$ coincides with a desired CoP $\mathbf{r}_{cop,d}$. The center of mass (CoM) height is initially controlled using the default CoM height PD control law of Equation (2-18). While the robot is standing still, the control law controls to the default, nominal height:

$$\ddot{z}_d = k_p(z_0 - z) - k_d \dot{z}, \quad (5-1)$$

where \ddot{z}_d is the desired vertical acceleration.

Because $\mathbf{r}_{cop,d}$ is proportionally controlled based on the instantaneous capture point (ICP) error ξ_e , the bang-bang controller is activated when ξ_e is larger than a minimum ICP error threshold $\xi_{e,min}$, which can be used for tuning. For the development of a controller for a standing robot, the initial ‘bang’ is always upward, as $\mathbf{r}_{cop,d}$ can be placed in the direction of

the ICP error. This placement of $\mathbf{r}_{cop,d}$ is possible, as the CoM is inside the support polygon while the robot is standing, and thus when the push will be applied.

The first and the second acceleration ‘bang’ of the control law are jerk limited, unlike the control law for the vertical force constrained capture positions in Section 3-3. Also, the duration of each ‘bang’ is not predefined, but determined online. Determining the duration online, allows for more control of the vertical dynamics, as the current state can be constantly evaluated on violation of the maximum height constraint. A predefined duration as in Section 3-3 does not use any feedback and is strictly feedforward. In relation to these differences in control design, the following parameters are used:

- \ddot{z}_{max} : maximum allowed vertical CoM jerk;
- $\alpha_{\dot{z}_c}$: parameter to scale down expected \dot{z}_c for the second ‘bang’, due to jerk limits.

The control sequence for the bang-bang control law reads as follows. For the first ‘bang’, the desired acceleration is set to $\ddot{z}_d = \ddot{z}_c$. The transition from the first ‘bang’ to the second is if the predicted CoM height violates the maximum height constraint z_{max} . This prediction of violation of the constraint is computed as follows, using a conservation of energy:

$$m\hat{g}z + \text{sgn}(\dot{z})\frac{1}{2}\dot{z}^2 > m\hat{g}z_{max}, \quad (5-2)$$

$$z + \text{sgn}(\dot{z})\frac{1}{2}\frac{\dot{z}^2}{\hat{g}} > z_{max}, \quad (5-3)$$

where \hat{g} is the predicted downward acceleration of the second ‘bang’. Using the acceleration constant \dot{z}_c of the second bang and the scale down parameter $\alpha_{\dot{z}_c}$ for the predicted downward acceleration, gives:

$$z + \text{sgn}(\dot{z})\frac{1}{2}\frac{\dot{z}^2}{\alpha_{\dot{z}_c}\dot{z}_c} > z_{max}. \quad (5-4)$$

When this condition is met, the second ‘bang’ is activated and $\ddot{z}_d = -\ddot{z}_c$. The second ‘bang’ continues until the vertical velocity changes direction: $\dot{z} < 0$. This results in the CoM height to be controlled to z_{max} until the controller turns off:

$$\ddot{z}_d = k_p(z_{max} - z) - k_d\dot{z}. \quad (5-5)$$

Controlling to z_{max} allows the robot to apply more force on the ground, compared to controlling directly to z_0 again. The controller turns off if the ICP error ξ_e is at a small value, a measure for stability. If the controller is turned off, the CoM height is controlled to the default height and the reference height is set to $z_r = z_0$. Finally, the rate of \dot{z}_d is limited with the maximum allowed jerk \ddot{z}_{max} and the desired vertical momentum rate is computed as $\dot{\mathbf{i}}_{d,z} = m\ddot{z}_d$. The desired horizontal linear momentum rate is adjusted for the added vertical acceleration as follows:

$$\dot{\mathbf{i}}_{d,xy} = \frac{\mathbf{c}_{xy} - \mathbf{r}_{cop,d}}{z}(mg + \dot{\mathbf{i}}_{d,z}). \quad (5-6)$$

In the remainder of this chapter, this control law will be compared with the default, constant height, control law that uses CoP only. The two control setups will be referred to as:

- **Vertical motion:** the control law presented in this section. Results are depicted in blue, unless stated different.
- **Default:** the control law as presented in Section 2-2. Results are depicted in black, unless stated different.

5-2 Results

Tests are conducted by applying a push at chest height on Valkyrie. The control framework parameters and settings used for the tests on Valkyrie can be found in Appendix A. Additionally, hardware tests are conducted on Atlas, but no clear improvement in push recovery is observed. To measure the impulse applied on the robot, a normalized impulse i is used:

$$i = \frac{\int F_{push} dt_{push}}{m_{robot}}, \quad (5-7)$$

where F_{push} is the push force and m_{robot} the robot mass.

5-2-1 Valkyrie Simulation

For Valkyrie in simulation, the push duration is set to $t_{push} = 0.15$ [s], as on hardware tests it is found that approximately the same push duration can be applied. The push force profile in simulation is constant. The ICP error parameter is set to $\xi_{e,min} = 0.03$ [m], as with this error $\mathbf{r}_{cop,d}$ is near the polygon edge. A value of $\ddot{z}_c = 2.4$ [m/s²] and $\dot{z}_{max} = 80.0$ [m/s³] are used, as this was also found to work well on hardware. A value of $\alpha_{\dot{z}_c} = 0.4$ is used for the prediction of the vertical CoM motion. An initial height of $z_0 = 1.0$ [m] is used and a maximum height of $z_{max} = 1.065$ [m]. Valkyrie has a total mass of $m_{robot} = 127.3$ [kg].

Maximum Recoverable Push

A search is used in simulation to find the maximum push force where the robot could still recover from. This search is conducted every 5 degrees change in horizontal push direction. In Figure 5-1, a polar plot is shown for the maximum recoverable impulses i for the default setup and the vertical motion controller. A push in the back corresponds with 0 degrees and a frontal push with 180 degrees.

The recovery improved when using the vertical motion controller after the default controller in push directions coming from the back or the side of the robot. Remarkably, push recovery is worse for push directions coming from the front of the robot. According to the variable height inverted pendulum (VHIP)-linear inverted pendulum (LIP) comparison, the direction of the disturbance should not make a difference. Compared to the default setup, the vertical motion controller recovered of a 9% higher impulse for a push direction coming from the back of the robot, a 4% higher impulse for a push direction coming from the side and a 7% lower impulse for a push direction coming from the front. The improvement in maximum recoverable push is the best for a push coming from the back. Also, it is relatively easy to apply a push in the back on the real robot, compared to other parts of the robot. Valkyrie has a hard, flat back surface, which can handle larger impacts than other body parts. For those reasons, there is chosen to make a deeper evaluation of the specific push direction of 0 degrees, a back push, in simulation too.

Comparison of Equal Push

To evaluate the responses after a push in the back of the robot, pushes of equal magnitudes are applied on both control setups. In Figure 5-2, a phase plot is shown for the horizontal CoM

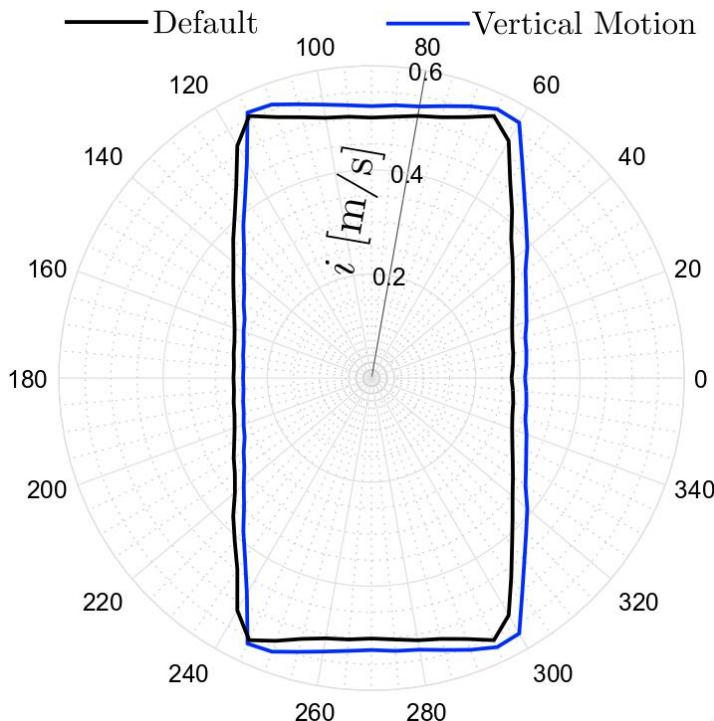


Figure 5-1: Polar plot of maximum recoverable pushes with an increment of 5 degrees. 0 degrees is a push from the back. The radius is the normalized impulse i .

state in the sagittal plane. One impulse is the maximum recoverable push for the default setup, the other for the vertical motion controller. The area covered by the phase plot is smaller for the vertical motion controller for a push of $i = 0.271 \text{ [m/s]}$. With the larger push of $i = 0.295 \text{ [m/s]}$, the default control setup does not manage to stabilize and diverges, while the vertical motion controlled setup recovers.

In Figure 5-3, time response plots are shown for a push of $i = 0.271 \text{ [m/s]}$. ‘Achieved’ is the value after the quadratic program (QP) found a solution. For the vertical linear momentum rate $\dot{\mathbf{l}}_z$, the achieved tracks the desired fairly well. There is a larger difference between the desired and achieved horizontal linear momentum rate $\dot{\mathbf{l}}_x$ for both control setups. If the achieved is compared between the two setups, the vertical motion controller achieves almost double the amount $\dot{\mathbf{l}}_x$ of what the default control setup achieves from 0.1 [s] until 0.25 [s]. There is a small difference in achieved angular momentum rate observable. However, if there is looked at the pelvis error $\theta_{pel,y}$ and the torso error $\theta_{torso,y}$ in the fifth and sixth row, the vertical motion controller has a little less rotation error than the default setup. This indicates no additional use of angular momentum strategies.

In the right column of the figure, from the ankle pitch torque $\tau_{ak,y}$, knee pitch torque $\tau_{kn,y}$, hip pitch torque $\tau_{hp,y}$ and back pitch torque $\tau_{bk,y}$, the difference in ankle and knee torque between both setups is the most noteworthy. The ankle torque of the vertical motion controller has a higher peak. Also, the ankle torque returns to the steady state value earlier than the default control setup. Similarly, the desired and measured CoP returns to the steady state value earlier. This may be an indication of an increase in robustness for the push, as is observed in the phase plot as well. The knee torque is on average lower for the vertical motion controller.

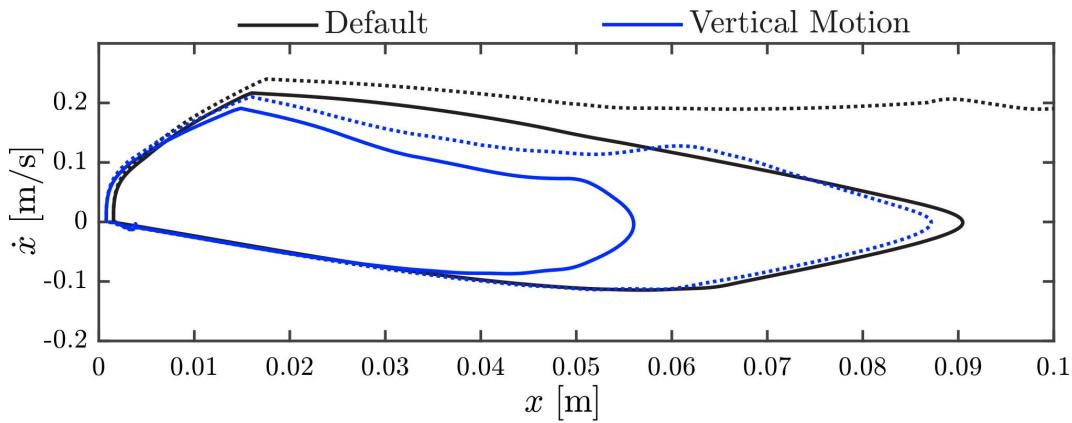


Figure 5-2: Phase plot of horizontal CoM motion in the sagittal plane for a push of $i = 0.271$ [m/s] (solid) and a push of $i = 0.295$ [m/s] (dotted).

On average, the ICP error $\xi_{e,x}$ is smaller, which indicates again an increase in robustness for the applied push.

5-2-2 Valkyrie Hardware

Hardware tests on Valkyrie are conducted by applying a push in the back at chest height on the physical robot. An iLoad Pro Digital load censor [43] is used to measure the impulse of the push. The load sensor is mounted to an aluminum stick. On the other side of the load cell, a steel plate with a rubber surface is mounted, which prevents the robot from being damaged. In Figure 5-4a, the end of the push stick with the load sensor is depicted. In Figure 5-4b, the test setup is shown, where the author is about to apply a push using the stick. A value of $\alpha_{\hat{z}_c} = 0.8$ is used for the prediction of the vertical dynamics. The remaining parameters are set to the same values as used in simulation.

Maximum Recoverable Push

To find the maximum recoverable push on hardware, a dozen test results are averaged where the CoM of the robot came closer than 5 [mm] from the polygon edge, but the robot still recovered. In Figure 5-5a, the average force profile with the standard deviation of the 12 pushes is depicted for the default setup. The average impulse is 35.3 [Ns], which equals a normalized impulse of $i = \frac{35.3}{127.3} = 0.277$ [m/s]. This is a slightly higher value than the impulse recoverable in simulation for this setup. In Figure 5-5b, the average force profile with the standard deviation is depicted for the vertical motion controller. The average impulse is 37.6 [Ns], which equals $i = 0.295$ [m/s], which is an increase in recoverable push of approximately 7% compared to the default setup.

Comparison of Equal Push

By selecting a force profile from each setup where the integrated force is approximately equal and the force profiles have a comparable shape, a comparison is made between both

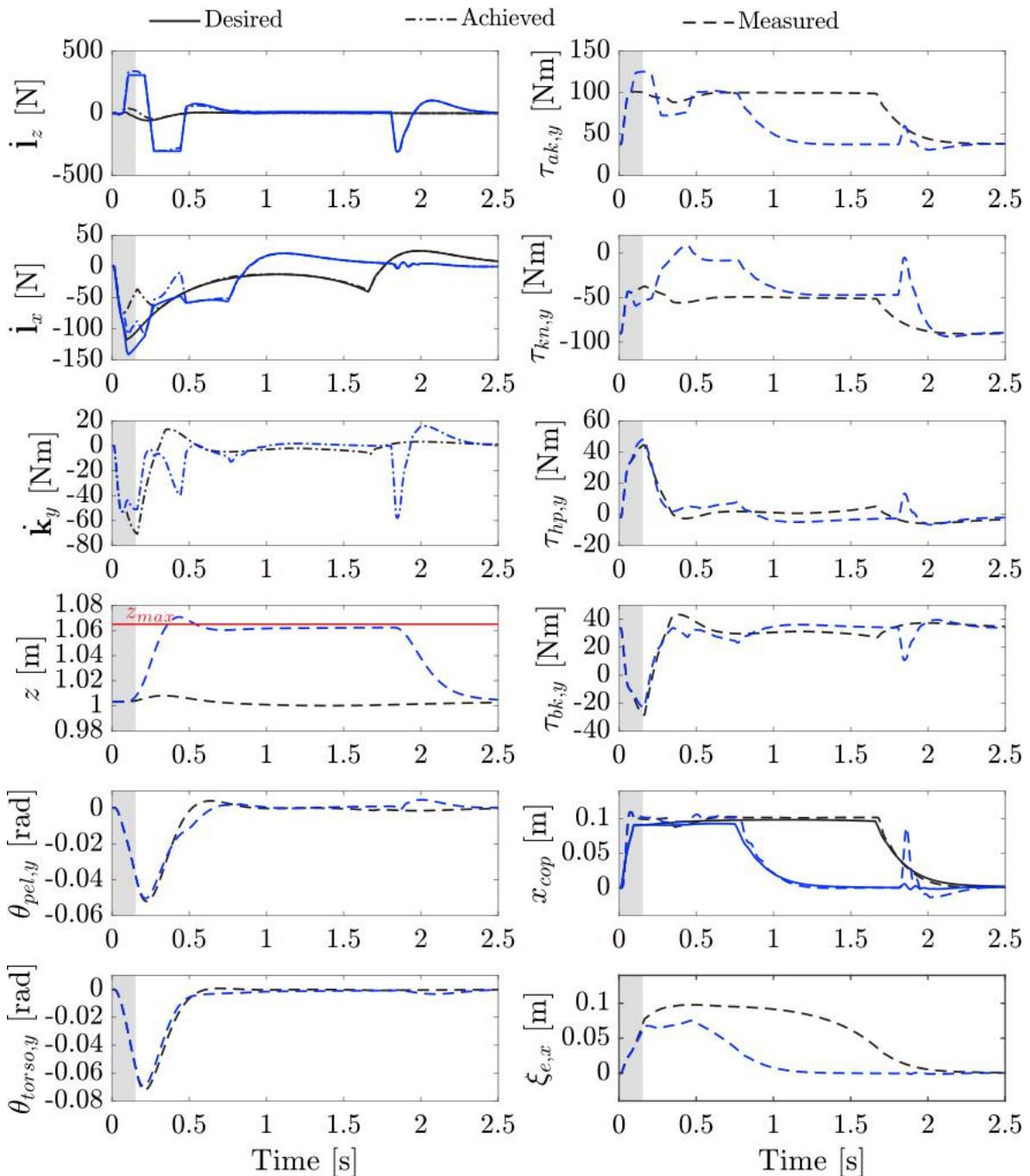


Figure 5-3: Time response plots for a push of $i = 0.271$ [m/s] for the default setup (black) and the vertical motion controller (blue). ‘Achieved’ is the value of the variable after the QP found a solution. The gray area is where the push is applied.

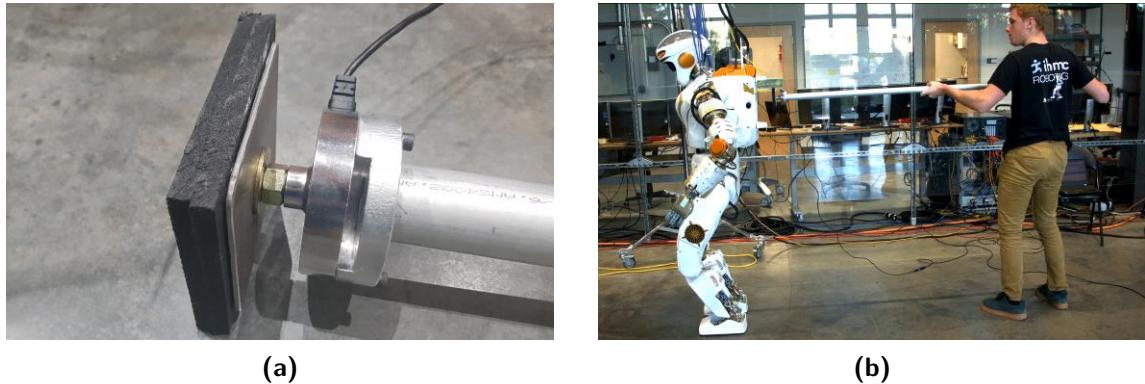


Figure 5-4: (a) Push head of push stick with a rubber surface, steel plate, load sensor and aluminum stick. (b) Tests setup, where the author applies a push using the push stick on Valkyrie.

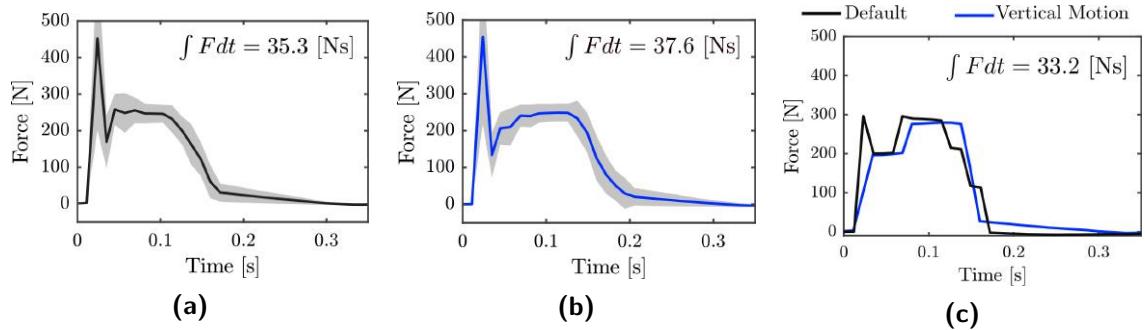


Figure 5-5: Average impulse of 12 pushes, where the CoM came closer than 5 [mm] from the polygon edge for the (a) default setup and (b) vertical motion controller. (c) Two picks of pushes, where the force profile and integrated force applied on both setups were similar.

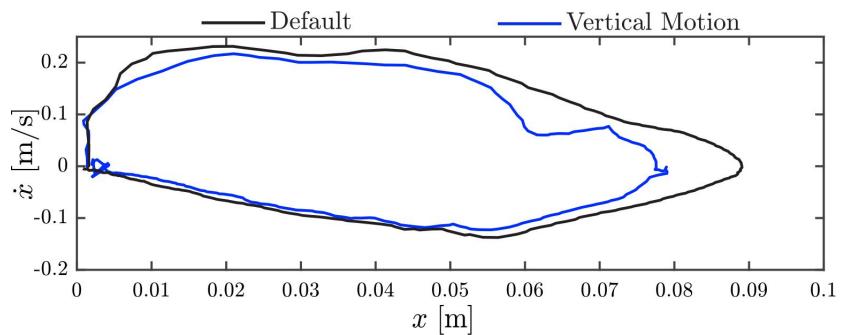


Figure 5-6: Phase plot of sagittal CoM motion for a push of $i = 0.261 \text{ [m/s]}$.

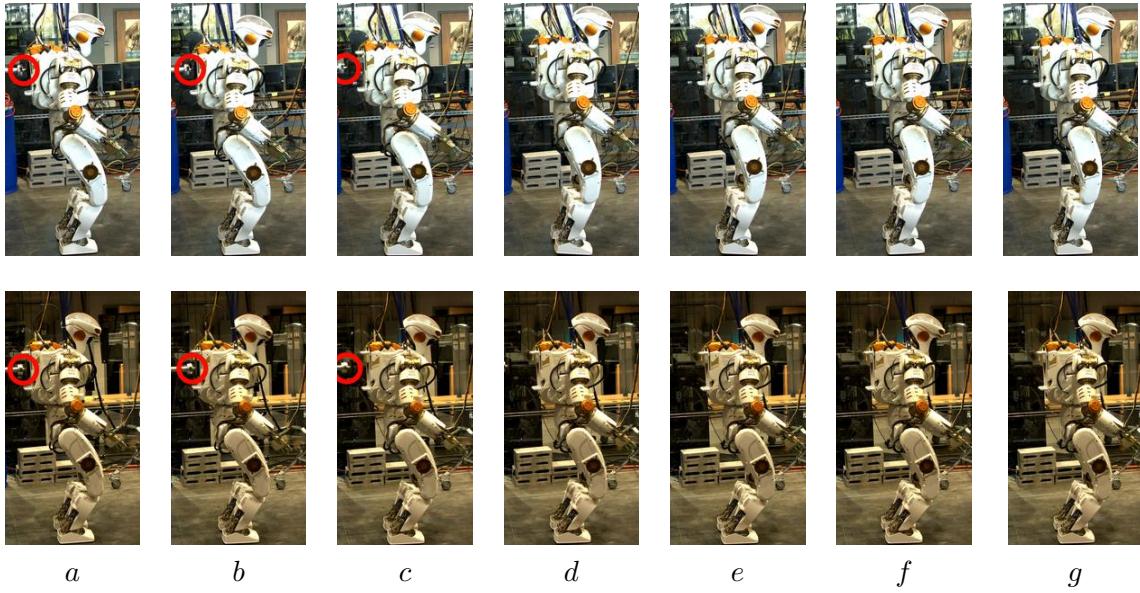


Figure 5-7: Time-lapse of Valkyrie recovering from a push using vertical motion (top row) and using the default controller setup (bottom row). The letters below the columns match with the letters next to the yellow lines in Figure 5-8. The push rod tip is encircled in red.

control setups under a similar disturbance. In Figure 5-5c, the two compared force profiles are shown. Both profiles have an integrated force of 33.2 [Ns], which is equal to $i = 0.261$ [m/s]. In Figure 5-6, a phase plot for the sagittal horizontal CoM motion is depicted. Like in simulation, the vertical motion covers a smaller area.

In Figure 5-7, a time-lapse is shown for both control setups after these pushes. The letters under each column match with the letters next to the vertical yellow lines in Figure 5-8. In Figure 5-8, time response plots are shown for the pushes of $i = 0.261$ [m/s]. Note how the vertical lines leave the gray area, when in Figure 5-7 the red encircled push head loses contact with the robot.

Also note how the timing of each ‘bang’, visible in the vertical linear momentum rate $\dot{\mathbf{i}}_z$, is different compared to the results observed in simulation. This might be a reason why a different value of $\alpha_{\dot{\mathbf{z}}_c}$ had to be chosen for tuning of the maximum CoM height. The $\dot{\mathbf{i}}_x$ plots are comparable with the simulation results. The achieved angular momentum rate $\dot{\mathbf{k}}_y$ of the vertical motion controller has a relatively larger overshoot than in simulation. However, the resulting pelvis and torso rotation errors are not larger for the vertical motion controller compared to the default setup, which indicates no additional use of angular momentum strategies, like in simulation.

In the right column of the figure, the torques have a desired and a measured value, as the torques of the robot are PD controlled with electrical current as input. Like in simulation, the differences in ankle torque and knee torque between the default setup and the vertical motion controller are the most noteworthy. For the vertical motion controller, the ankle torque has a higher peak, but returns quicker to steady-state, like in simulation. The CoP also returns earlier to steady state, which shows a slight increase in robustness for the applied push. Equivalently, the average ICP error is again smaller for the vertical motion controller.

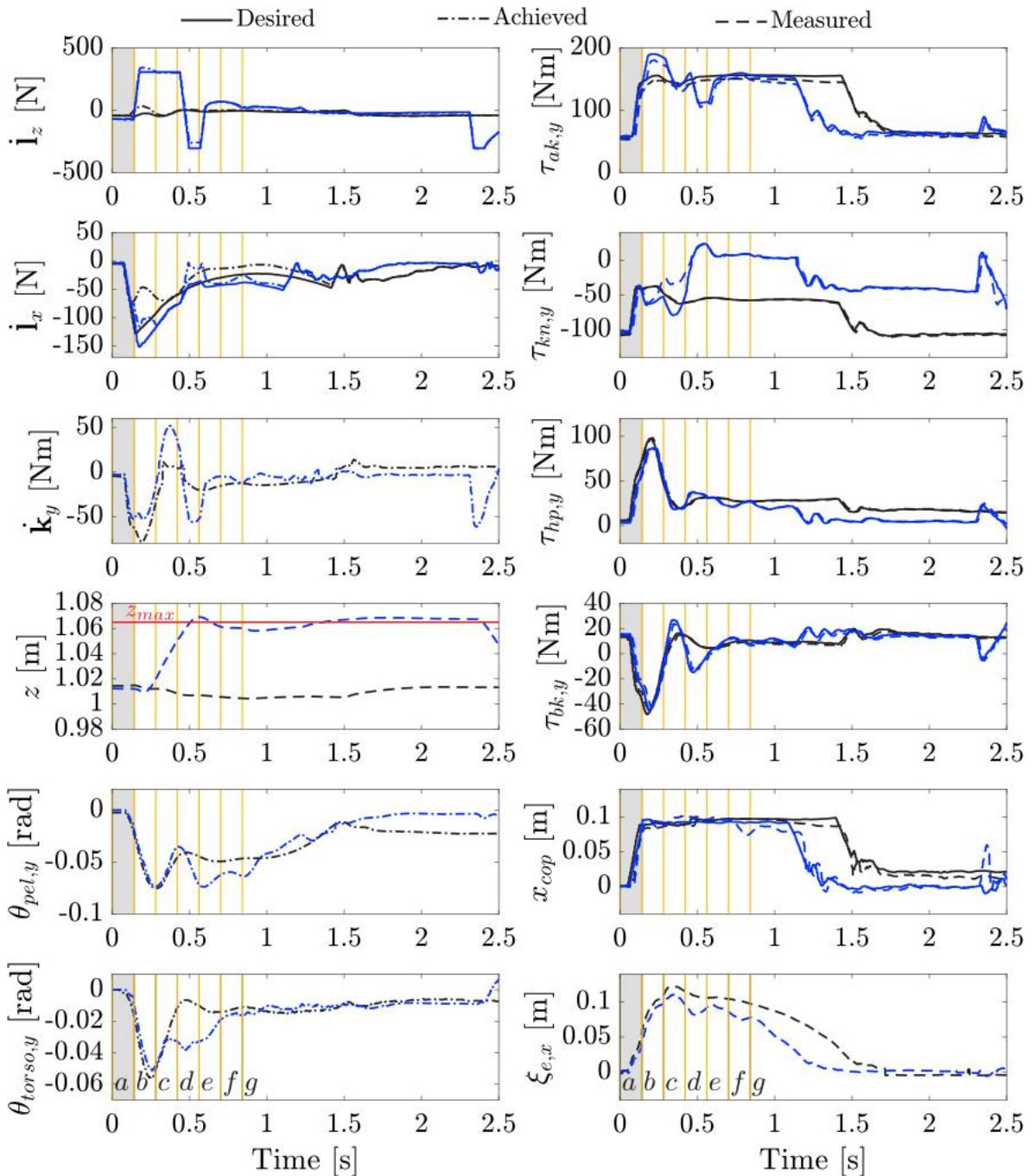


Figure 5-8: Time response plots for a push of $i = 0.261$ [m/s] for the default setup (black) and the vertical motion controller (blue). All joint torques, except for the back, are the average over left and right.

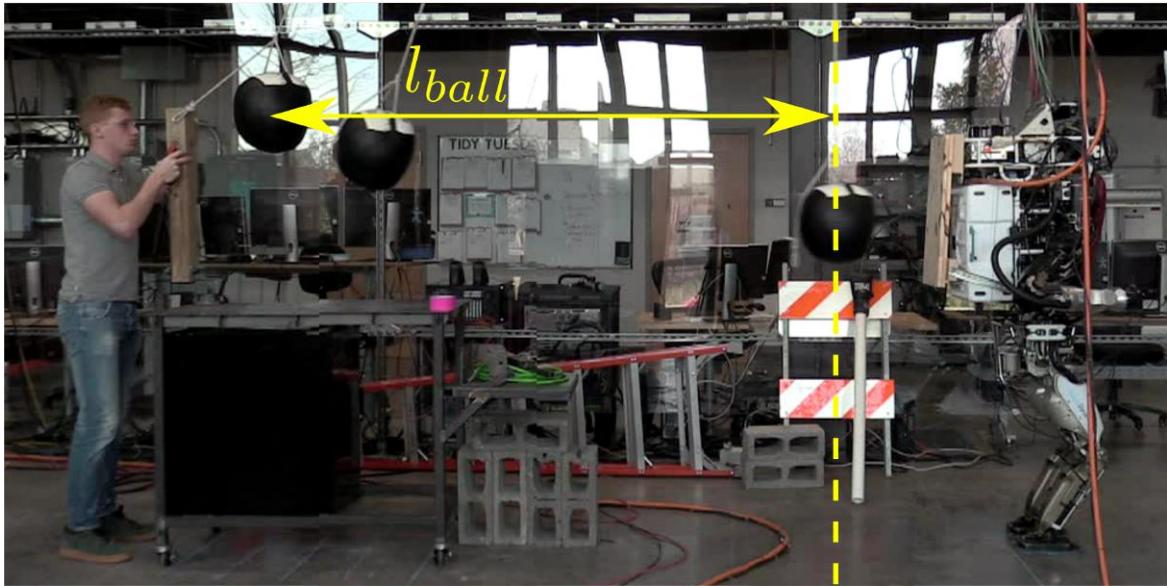


Figure 5-9: Experimental setup for hardware tests on Atlas.

5-2-3 Atlas Hardware

Next to the Valkyrie tests, tests are conducted on Boston Dynamics' Atlas humanoid robot ($m_{robot} = 155.9$ [kg]). An alternative experimental setup is used to test push recovery on hardware. A medicine ball with mass $m_{ball} = 13.6$ [kg] is hung from the ceiling at the robotics lab of the Institute for Human and Machine Cognition (IHMC). To push the robot, the ball is released from a certain distance. The impulse on the robot is assumed to be only depending on the difference in potential energy of the ball between release height and its lowest position. To protect the robot from the impact of the ball, a wooden plate is mounted on the frame of the robot. In Figure 5-9, the test setup is depicted. The length l_{ball} from the horizontal release location to the dead point of the pendulum is used to calculate the impulse applied on the robot. To release the ball, a simple mechanism is made in an attempt to get a more constant initial ball position. This mechanism is depicted in Figure 5-10. By pulling the aluminum pin, the loop on the rope will slide off the pin.

Maximum Recoverable Push

More than 60 experiments are conducted, but no clear improvement in recovery compared to the default controller could be observed. Different initial heights for the robot are tried, by gradually lowering the default initial height of $z_0 = 1.12$ [m] down to $z_0 = 1.05$ [m]. Lowering the initial height made the relative increase in recoverable push compared to the default setup larger in simulation, but recovery on hardware did not improve. The increase in recoverable push compared to the default controller for these initial heights in simulation are in the range from 6% to 10%. In another attempt to improve recovery, the foot polygon size for the whole-body QP was enlarged as well. The robot would recover from larger impulses, but the difference observed between the two control setups was still neglectable.

Additionally, in an attempt to improve recovery for the vertical motion controller, a gain



Figure 5-10: Release mechanism for the medicine ball tests.

for joint torque control is tuned. From the data collected from earlier tests, it was observed that the measured CoP had a relatively large tracking error with $\mathbf{r}_{cop,d}$, especially during the second ‘bang’ of the control law. The joint torques on Atlas are controlled using an input current I and are computed as follows [8]:

$$I = I_\tau + I_{\dot{q}}, \quad (5-8)$$

$$I_\tau = k_{ff_\tau} \tau_d + k_\tau (\tau_d - \tau), \quad (5-9)$$

$$I_{\dot{q}} = k_{ff_{\dot{q}}} \dot{q} + k_{\dot{q}} \left(\int \ddot{q}_d dt - \dot{q} \right), \quad (5-10)$$

where k_{ff_τ} and k_τ are a feedforward and feedback gain on tracking of desired joint torques respectively. The gain $k_{ff_{\dot{q}}}$ is a feedforward gain to compensate the oil flow in the cylinder as the actuator is moving. The gain $k_{\dot{q}}$ is a feedback gain on desired joint velocity, where the desired joint velocity is computed by integration of the desired joint acceleration \ddot{q}_d , an output of the whole-body QP. For the tests, CoP tracking improved by increasing the gain $k_{\dot{q}}$ from 15.0 to 40.0. In Figure 5-11, the difference in CoP tracking between the two gain values is depicted during the two ‘bangs’ of the control law using a vertical acceleration of $\ddot{z}_c = 3.2$ [m/s²]. Note how tracking improved during the second ‘bang’ after increasing the gain. Unfortunately, push recovery did not noticeably improve.

Comparison of Equal Push

For an initial height of $z_0 = 1.10$ [m], a maximum height of $z_{max} = 1.17$ [m] and a vertical acceleration of $\ddot{z}_c = 5.0$ [m/s²], a series of 20 tests for each control setup are conducted to investigate the responses after an equal push. The increase in recoverable push for these parameters in simulation is approximately 8%, which indicates that there could be a noticeable difference in performance between the two control setups for this configuration. For the hardware tests, an initial ball position of $l_{ball} = 2.54$ [m] is used. The medicine ball height

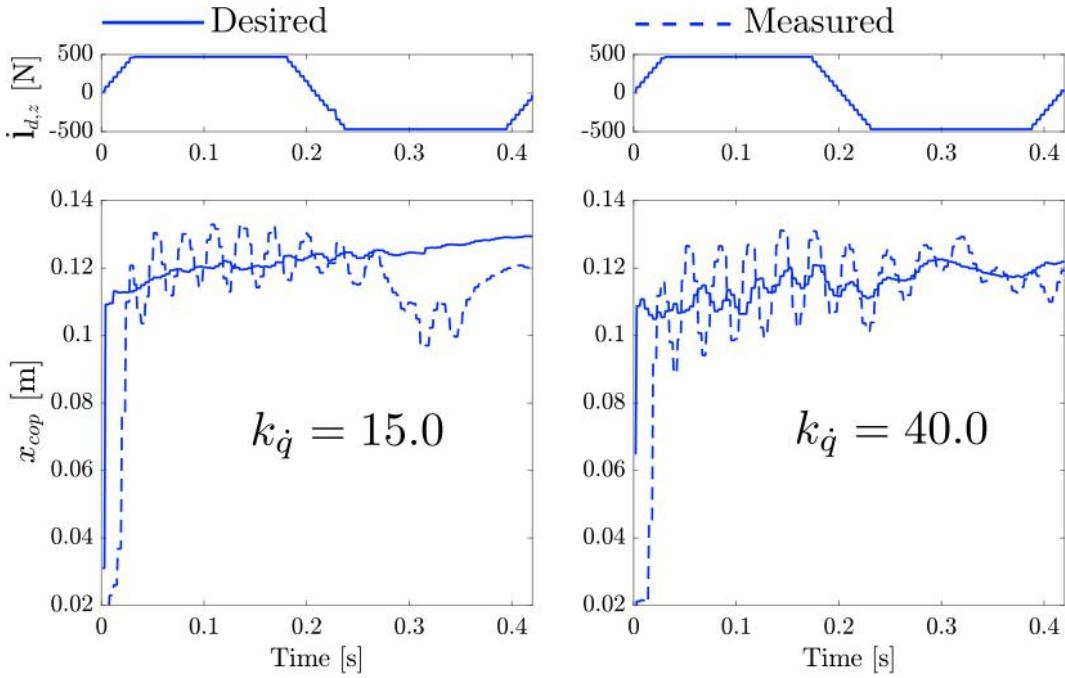


Figure 5-11: CoP tracking during the two ‘bangs’ of the vertical motion control law. In the left column, $k_{\dot{q}} = 15.0$. In the right column, $k_{\dot{q}} = 40.0$. For illustration of the ‘bangs’, $\dot{\mathbf{l}}_{d,z}$ is graphed.

difference δz_{ball} is computed as follows:

$$\delta z_{ball} = l_{rope} - \sqrt{l_{rope}^2 - l_{ball}^2} = 6.12 - \sqrt{6.12^2 - 2.54^2} = 0.543 \text{ [m]}, \quad (5-11)$$

where l_{rope} is the length of the rope between the ceiling attachment point and the ball. The impulse applied on the robot is calculated as:

$$m_{ball}\dot{x}_{ball} = m_{ball}\sqrt{2g\delta z_{ball}} = 13.6\sqrt{2 \cdot 9.81 \cdot 0.543} = 44.4 \text{ [Ns]}, \quad (5-12)$$

where \dot{x}_{ball} is the velocity of the ball. Normalizing the impulse, like with the Valkyrie tests, results in a value of $i = \frac{44.4}{155.9} = 0.285$ [m/s]. From the logging camera it is observed that the push duration is between 0.03 [s] and 0.05 [s].

In Figure 5-12, phase plots are shown for these tests. The thin transparent lines show the responses after each individual test. The bold plots show the averages of all results. The average phase plot for the vertical motion controller covers a slightly smaller area. However, the individual responses have a wider distribution, which shows a less predictable behavior. The wider distribution was also visible on the robot, as sometimes the robot would visibly almost fall over, and sometimes recover relatively quick. A reason for this could be that the robot moves when using height variation, which might cause larger sensing and actuation errors. Also, for the vertical motion controller, there is a slight increase in velocity observable after approximately $x = 0.06$ [m] in the upper half of the phase plot. Assuming no angular momentum change in the robot, an increase in velocity can be caused by the CoP still tracking poorly. A reason for this could be that, for the tests in the phase plot, a vertical acceleration of $\ddot{z}_c = 5.0$ [m/s²] was used, instead of the $\ddot{z}_c = 3.2$ [m/s²] used in the tuning of the gain $k_{\dot{q}}$, which might have made CoP tracking worse.

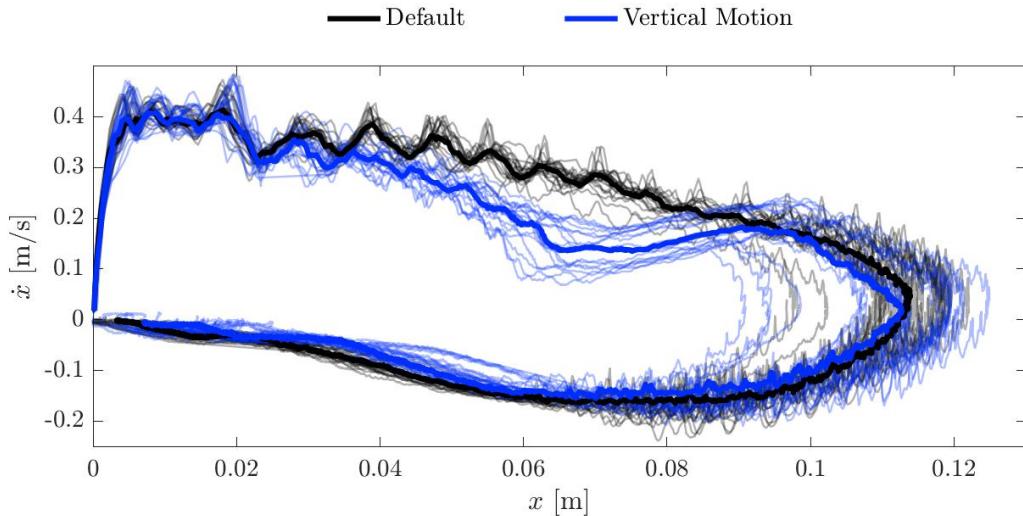


Figure 5-12: Phase plots of 20 Atlas hardware tests for each setup (thin transparent) with the averaged data for each setup (bold).

5-2-4 Comparison with Capture Regions

If it is assumed that the difference in recovery between the default setup and the vertical motion controller is equal to the difference between the LIP and the VHIP, a comparison can be made between the push recovery tests and the VHIP capture regions presented in Chapter 3.

Considering this assumption, the results on Atlas and Valkyrie are compared with the vertical force constrained capture region with the same \ddot{z}_c in the bang-bang control law. The following dimensionless vertical acceleration and maximum height are used for both robots:

- **Valkyrie:** $\ddot{z}_c = 2.4 \text{ [m/s}^2\text{]} \rightarrow \ddot{z}'_c = \frac{1}{4}$, $z'_{max} = \frac{1.065[\text{m}]}{1.0[\text{m}]} = 1.065$;
- **Atlas:** $\ddot{z}_c = 5 \text{ [m/s}^2\text{]} \rightarrow \ddot{z}'_c = \frac{1}{2}$, $z'_{max} = \frac{1.17[\text{m}]}{1.10[\text{m}]} = 1.064$.

In Figure 5-13, the increase in recoverable push for both robots are plotted in a zoomed-in version of the capture velocity plot of Figure 3-4. Unlike for Valkyrie, for Atlas in simulation only rear push recovery results are shown, as other directions are not discussed in this chapter. The difference in push recovery differs from the VHIP-LIP difference for both robots. For Atlas, the simulation result is located on the height constrained bound and the hardware result shows no increase in recoverable push.

Note how the simulation rear push recovery for Valkyrie even lies outside the height constrained bound in this plot. The average improvement on hardware for Valkyrie lies just inside the height constrained bound. The increase in recovery for the side push in simulation is about equal to the comparable force constrained capture position of $\ddot{z}'_c = \frac{1}{4}$. The frontal push recovers worse than the default control setup and is not comparable with the capture regions presented in Chapter 3.

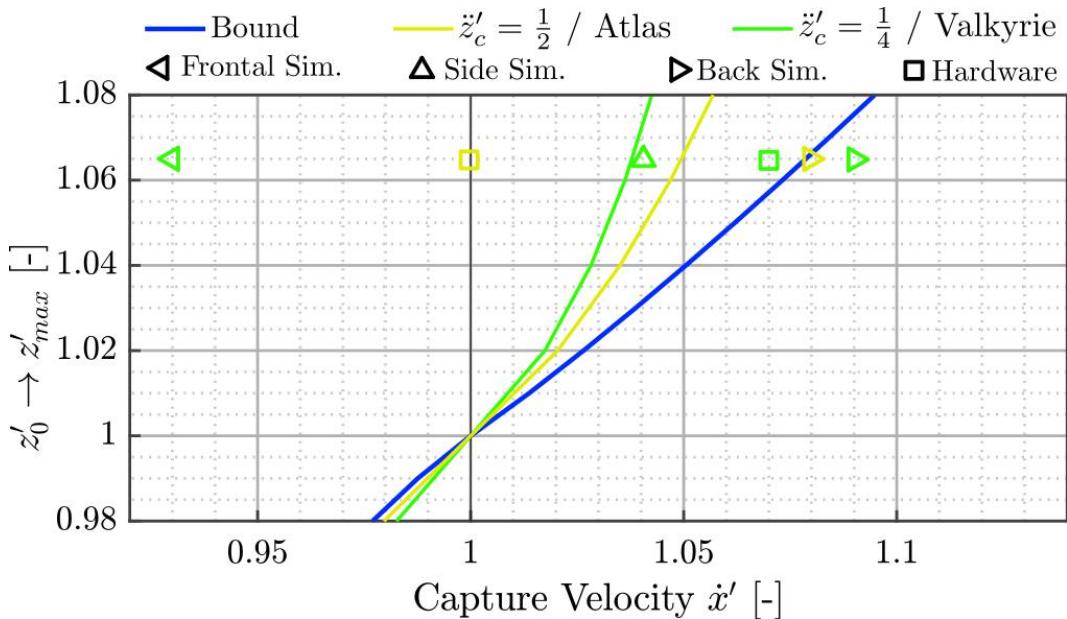


Figure 5-13: Difference in recovery of Valkyrie and Atlas in simulation and on hardware between the vertical motion controller and the default setup, plotted in the capture velocity plot in Figure 3-4.

5-3 Discussion

Chapter 5 presented a simple control law that activates a bang-bang action on vertical acceleration if a predefined threshold is met. Using the bang-bang controller, the vertical CoM dynamics are explicitly solvable and can be computed every tick. Also, hardware results were presented, which have not earlier been shown considering the research topic.

A difference between the related work in Section 2-3 is that most other existing control strategies use model predictive control (MPC). With MPC however, the problem formulation is very important for the performance of the controller. Caron & Mallein [14] present a method that generates capture trajectories based on the current CoM position and velocity, and the support polygon information. The objective is to minimize height variations, which results in the robot to only use vertical CoM motion if the desired zero moment point (ZMP) is placed on the polygon edge. Using this problem formulation though, the resulting capture trajectory is always under the assumption that the previously applied disturbance stopped disturbing. In the case of a push though, the increase in error is not an instant event and happens over time.

The control law presented in this chapter considers a different approach to the problem. Comparing with the method presented in [14], which has a constraint on minimum and maximum vertical acceleration in the MPC law, the bang-bang controller in this chapter can be seen as the extremum of what the MPC would output. Therefore, depending on the selection of the threshold when to activate the controller, the method presented in this chapter will in some cases use more, not necessarily needed, height variation than the control law of [14]. However, by using the additional height variation, the method might be more robust for longer push durations. The MPC law in [14] computes capture trajectories based on current

velocity differences and does not take expected future disturbances into account.

There were two push recovery test setups considered in this chapter for testing the controller on hardware while the robot is standing. An advantage of the tests with the stick is that the force data from the load cell could be measured with an accuracy of 0.25% at a record frequency of 100 [Hz] [43]. However, many tests had to be conducted to obtain the results of the maximum recoverable pushes, as the person pushing the robot was not always capable of applying the exact same force. With the medicine ball tests, the maximum recoverable push could be found more easily and fewer tests were needed. However, no measurement data was available of the applied push force. Also, the push duration could not be manipulated. Furthermore, stretch in the rope, a changing CoM position in the ball and energy loss of the impact of the ball can all influence the impulse applied on the robot.

Testing push recovery, Valkyrie and Atlas showed a similar behavior in simulation. In Simulation Construction Set (SCS), a ground stiffness and damping is modeled, where the robot is in contact with the ground with 4 ground contact points per foot. Also, a joint torque damping is simulated. Therefore, the differences in Atlas and Valkyrie in simulation are predominantly related to the differences in the multi-body structures of the robots.

On hardware however, larger differences were observed when comparing the responses of Valkyrie and Atlas. Valkyrie matched the simulation data with more accuracy, which could be a cause of the torque sensing of the series elastic actuators. However, while performing the tests, it is observed that Valkyrie is more sensitive to tuning of the maximum vertical acceleration, maximum jerk and foot polygon size used in the whole-body QP. With careless tuning, Valkyrie could stand on its toes when recovering from a push. Also, the actuators could turn in a state of over-excitation which resulted in the need to restart the robot. Using Atlas, less care had to be taken for tuning and the robot performed well in all tests in terms of internal stability. The less accurate sensing and the stiction in the hydraulic system on Atlas could be a cause that the differences observed between hardware and simulation were larger than on Valkyrie. For the future, it could be interesting to improve sensing and state estimation on Atlas, as this might improve push recovery of the vertical motion control law.

To analyze the responses on hardware, phase plots of the horizontal CoM state of the robots and different time responses were evaluated. The phase plots show the development of velocity over position and give insight in the ability of the robot to return to the desired steady state configuration. The time response plots show the behavior of momentum rates, joint pitch torques and CoP and ICP positions over time. However, no traditional system evaluations were conducted, such as a bode plot or other frequency analysis methods. A reason for this is that in the control method, the eigenvalue of the system is time-variant. For additional analysis, it may be interesting to measure the energy consumption of the robots using both control strategies, using the input current and voltage. The energy consumption can, for example, introduce an additional decision variable to choose between balance strategies.

On most tests, the improvement in recovery of the vertical motion control law versus the default control setup differed from the difference in the LIP capture point (LIPCP) and the vertical force constraint capture points. Reasons for this could include numerical integration in simulation, unmodeled dynamics such as ground contact, sensing errors, actuation limitations or a bug in the software.

Furthermore, the control strategy presented in this chapter is a two-dimensional space (2D) strategy. The third dimension is controlled with the $\mathbf{r}_{cop,d}$ orthogonal to the push direction,

already existing in the control framework. When the robot is walking however, this 2D strategy does not always improve balance, as the CoM is not always in the center of the support polygon if the push is applied. Therefore, the control strategy presented in this chapter is extended in the next chapter for the use while the robot is walking.

Chapter 6

Application: Walking

The control strategy for a static, standing case presented in the previous chapter is two-dimensional. If the robot is walking, the horizontal center of mass (CoM) position can be located outside the support polygon. The result of this is that the desired center of pressure (CoP) cannot always be placed in line with the instantaneous capture point (ICP) error. In this chapter, the control law presented in Chapter 5 is extended for the use while the robot is walking.

6-1 Experimental Setup

In this section, to determine when to activate a similar control action as in the previous chapter, tests are conducted preliminary to developing a controller. In these tests, a push is applied in the beginning of single support while the robot is walking. The stepping parameters used for the tests are given in Table 6-1, which are the default stepping parameters while testing in simulation. In Figure 6-1, the test setup in simulation is shown. The limited foothold options display that footstep location adjustment is not available as a balance strategy and that other balance strategies, such as CoM height variation, might be needed to recover. In Figure 6-2, the ICP reference trajectory, the centroidal moment pivot (CMP) reference trajectory and the measured CoM trajectory are made visible for the right foothold, where the push will be applied.

Table 6-1: Stepping parameters for the walking tests.

Parameter	Value	Unit
Step length	0.5	[m]
Step width	0.25	[m]
Single support time	0.6	[s]
Double support time	0.25	[s]



Figure 6-1: Test setup for push recovery during walking in simulation. The limited foothold options show that footstep adjustment is not available as a balance strategy.

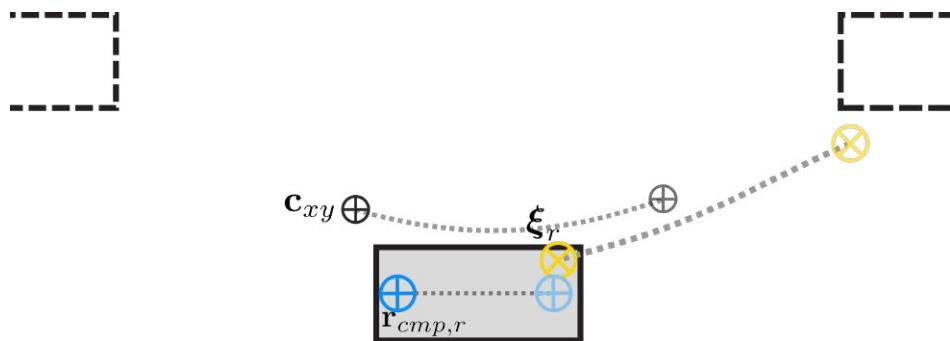


Figure 6-2: Trajectories during single support in the horizontal plane (gray dotted lines). The gray area is the current, right, footstep position where the push will be applied.

The following properties are observed, when applying pushes in different directions at the start of single support:

1. The direction of the ICP error stays often approximately the same until transition to double support;
2. If the ICP error is directed in the sagittal plane, the desired CMP often remains somewhat in the same location;
3. If the ICP error is directed in the coronal plane, the desired CMP slides from back to the front of the foot;
4. The configuration and velocity near transition to double support affect the robots ability to put its swing leg down at the desired time.

These properties are used as assumptions in the development of a control law.

6-2 Method

The control law of the previous chapter is first adjusted for desired CMP positions outside the support polygon to make a high-level comparison with angular momentum strategies. Second, decision variables are introduced for activating control actions. Third, control actions are presented, which are activated based on violation of thresholds of the presented control variables.

6-2-1 Avoiding Generating Additional Angular Momentum Rate

In the previous chapter, $\mathbf{r}_{cmp,d}$ is constrained to be inside the support polygon. Therefore, this point is assumed to coincide with $\mathbf{r}_{cop,d}$. In this section, the computation of the desired linear momentum rate is adjusted for $\mathbf{r}_{cmp,d}$ positions outside the support polygon. If $\mathbf{r}_{cmp,d}$ is outside the polygon, $\mathbf{r}_{cop,d}$ is obtained by projecting $\mathbf{r}_{cmp,d}$ on the polygon edge. For comparability with the default setup, the goal is to request little to no additional angular momentum rate from the robot to achieve the desired linear momentum rate. Therefore, the vertical motion controller generates an added desired linear momentum rate on top of the default control law if $\mathbf{r}_{cmp,d}$ is outside the support polygon. First, the desired horizontal linear momentum rate of change, as used in the previous chapter, is written in terms of $\mathbf{r}_{cmp,d}$. The assumption is made that the difference in body torque τ_c is directly related to the difference between $\mathbf{r}_{cmp,d}$ and $\mathbf{r}_{cop,d}$:

$$\dot{\mathbf{i}}_{d,xy} = \frac{\mathbf{c}_{xy} - \mathbf{r}_{cmp,d}}{z} (mg + \dot{\mathbf{i}}_{d,z}), \quad (6-1)$$

$$= \frac{\mathbf{c}_{xy} - \left(\mathbf{r}_{cop,d} - \frac{\tau_s}{(mg + \dot{\mathbf{i}}_{d,z})} \right)}{z} (mg + \dot{\mathbf{i}}_{d,z}), \quad (6-2)$$

$$= \frac{\mathbf{c}_{xy} - \mathbf{r}_{cop,d}}{z} (mg + \dot{\mathbf{i}}_{d,z}) + \frac{\tau_c}{z}. \quad (6-3)$$

Writing τ_c in $\mathbf{r}_{cmp,d}$ again, but only for the linear inverted pendulum (LIP) part of this equation (and assuming a constant height z_0 for this part), gives the following expression:

$$\dot{\mathbf{i}}_{d,xy} = \underbrace{\frac{\mathbf{c}_{xy} - \mathbf{r}_{cmp,d}}{z_0} mg}_{\dot{\mathbf{i}}_{d,xy,lip}} + \underbrace{\frac{\mathbf{c}_{xy} - \mathbf{r}_{cop,d}}{z} \dot{\mathbf{i}}_{d,z}}_{\dot{\mathbf{i}}_{d,xy,\dot{z}_d}}, \quad (6-4)$$

where $\dot{\mathbf{i}}_{d,xy,\dot{z}_d}$ is the additional desired horizontal linear momentum rate from the vertical motion controller and $\dot{\mathbf{i}}_{d,xy,lip}$ the desired horizontal linear momentum rate from the default control law.

In Figure 6-3, it is visually explained how this computation of $\dot{\mathbf{i}}_{d,xy}$ will not require additional angular momentum rate from the robot, as the scalar offset a in the figure is the same for both setups. If CoM height variation would be used based on the location of $\mathbf{r}_{cmp,d}$, a different angular momentum rate compared to the default setup would be needed to achieve the desired linear momentum rate. However, using this modified desired linear momentum rate with both $\mathbf{r}_{cmp,d}$ and $\mathbf{r}_{cop,d}$, the resulting CMP will be different than $\mathbf{r}_{cmp,d}$, as shown in the image.

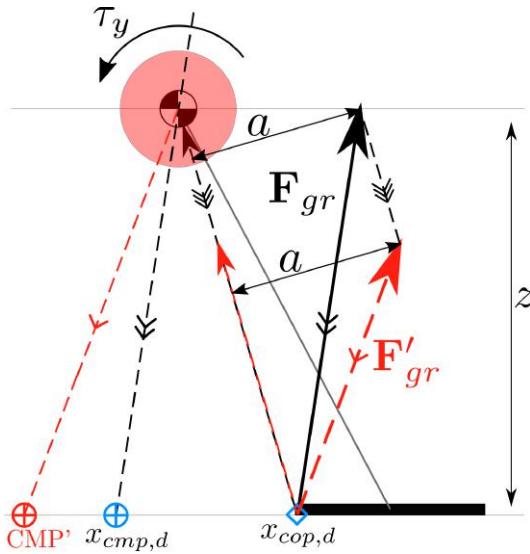


Figure 6-3: Explanatory drawing of the ground reaction force (GRF) resulting from the default desired momentum rate (black) versus the modified desired momentum rate for CoM height variation (red). The thin arrows show the part of the GRF that intersects with the CoM, which is used for height changes. The equal scalar offset a shows that the same angular momentum rate will be requested about the CoM.

6-2-2 Decision Variables

If the CoM is outside the support polygon, the local virtual leg between $\mathbf{r}_{cop,d}$ and the horizontal CoM location \mathbf{c}_{xy} may not be aligned with direction of the ICP error ξ_e . This results in the leg applying force in a different direction than is desired to cancel the error, which can result in additional error in another direction. Also, if \mathbf{c}_{xy} is close to the polygon edge, the distance with $\mathbf{r}_{cop,d}$ might be very small, such that height changes have little to no effect as the local variable height inverted pendulum (VHIP) is close to upright. To take these two aspects into account, the following variables are introduced, which will be used to determine when to use CoM height variation for balance:

- **Alignment angle ϕ :** the angle in the horizontal plane between the virtual leg $\mathbf{r}_{cop,d} - \mathbf{c}_{xy}$ and the ICP error ξ_e ;
- **Effective distance δ :** the distance in the horizontal plane between $\mathbf{r}_{cop,d}$ and \mathbf{c}_{xy} in the direction of the ICP error ξ_e .

In Figure 6-4a-b, the two variables are graphically explained using the right stance foot position and configuration of Figure 6-2. The desired CMP $\mathbf{r}_{cmp,d}$ is allowed to move a small distance outside the polygon. In Figure 6-4a, the angle ϕ is zero and the distance δ is relatively large. This is a relatively suitable error for height control, as the alignment angle is small and the local VHIP tip is relatively far from the base. In Figure 6-4b, the angle ϕ is 90 degrees and therefore the distance δ is zero. In this configuration, CoM height variation would not help drive the error back. Furthermore, an additional error would be caused, orthogonal to the current ξ_e , when using additional CoM height variation. Therefore, this error is not considered suitable for using CoM height variation for balance control.

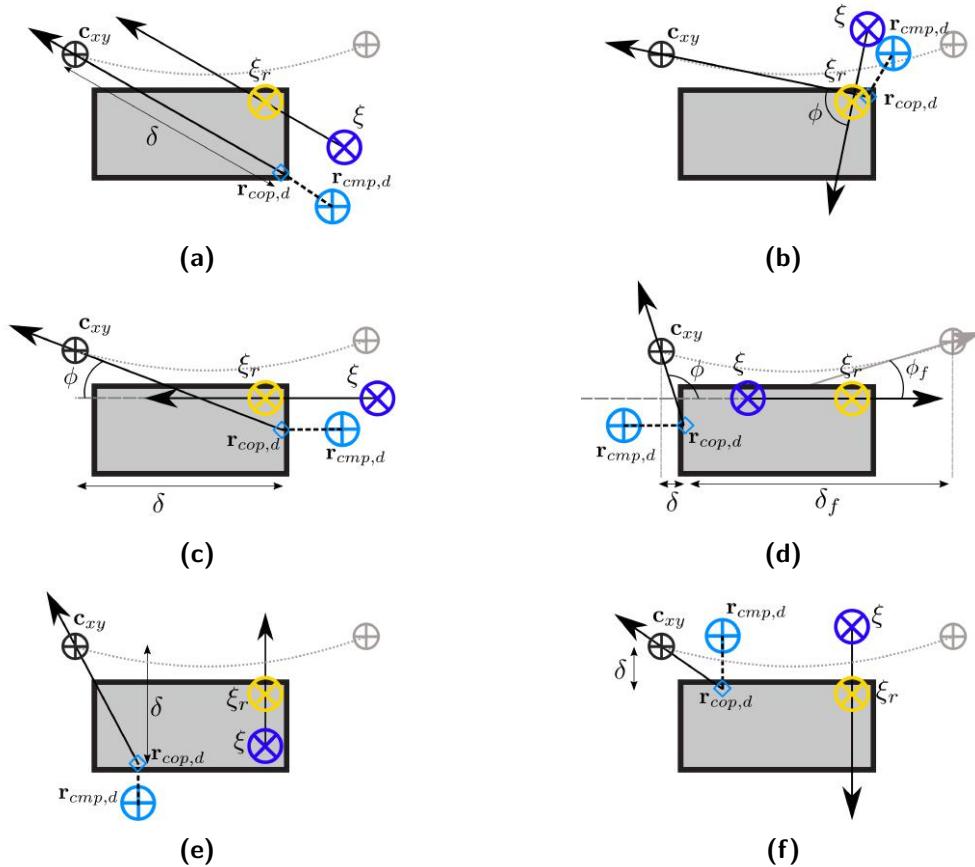


Figure 6-4: Explanatory visualizations of ϕ and δ for the configuration at start of single support with different ICP error directions.

6-2-3 Actions

The vertical motion control law is activated if a minimum threshold for ξ_e is met, like in the previous chapter. The alignment angle ϕ and the effective distance δ will be used to select a control action. From the assumptions of the preliminary observations 1 and 2, it is assumed that the angle ϕ will be dependent on the current ξ_e and $\mathbf{r}_{cop,d}$ throughout single support if ξ_e is directed in the sagittal plane. Therefore, the additional variables δ_f and ϕ_f are used, which are the expected alignment and distance in the end of single support, based on the planned CoM location coming from the ICP planner. Also, from assumption 3, ϕ is not used if a push is in the coronal plane, because throughout single support the foot length is available for future $\mathbf{r}_{cmp,d}$ placements that could correct any additional error. The sagittal or coronal plane error direction is determined based on the nearest polygon edge to $\mathbf{r}_{cop,d}$. Near the left or right edge shows a push in the coronal plane, near the front or back edge shows a push in the sagittal plane. Based on the discussed variables, the following three actions can be selected:

- **Positive alignment:** At the current control tick, ϕ is relatively small and δ relatively large for a push in the sagittal plane, or δ is relatively large for a push in the coronal plane. Also, ϕ has to be smaller than $\frac{1}{2}\pi$ [rad], as the virtual leg must be in direction

of the ICP error to make additional force effective. A bang-bang action is activated, similar to the control law of the previous chapter. The thresholds related to this action are $\phi_{min} < \phi < \phi_{max}$ (for a push in the sagittal direction) and $\delta > \delta_{min}$, where ϕ_{min} , ϕ_{max} and δ_{min} are parameters for the minimum and maximum ϕ and minimum δ .

- **Prepare:** At the current control tick, ϕ is relatively large or δ is relatively small, but ϕ_f and δ_f are at values suitable for the positive alignment action. The CoM height is gradually lowered to the minimum height, after which a positive ‘bang’ is used. The thresholds related to this action are $\phi_{min} < \phi_f < \phi_{max}$ and $\delta_f > \delta_{min}$.
- **Default:** All decisions variables ϕ , δ , ϕ_f and δ_f are at such values that vertical CoM motion does not improve recovery. The default height control law is used and no additional height changes are considered. The threshold related to this action is if the prepare and the positive alignment thresholds do not hold.

In Figure 6-4, six cases of ICP errors are shown to explain which actions will be used. In Figure 6-4a, the positive alignment action is used, as the δ is relatively large and ϕ relatively small. In Figure 6-4b, the default action is introduced, as δ is zero and ϕ is misaligned. In Figure 6-4c, the error is a result of a back push. A positive alignment action is introduced, as ϕ is relatively small and δ relatively large. In Figure 6-4d, a push is applied frontally on the robot. The prepare action is used, as δ_f and ϕ_f are more suitable for height control. In Figure 6-4e, the robot is pushed from the left. The positive alignment action is used, as the current δ is relatively large. In Figure 6-4f, the error is a result from a push from the right. The default action is activated, as δ is small throughout single support.

The bang-bang control law considered in the actions is similar to the control law presented in the previous chapter. However, the height constraints are slightly modified. For the maximum height constraint z_{max} , the same constant value as for the standing tests is used in the first half of single support. In the second half, the maximum height constraint is linearly interpolated between the maximum height constraint for standing and a maximum height constraint at the end of single support. For the minimum height constraint z_{min} , a constant value is considered throughout single support. The height constraints are visually explained in Figure 6-5.

Positive Alignment Action

Using z_{max} as specified above, a similar bang-bang control law as in the previous chapter is introduced with the positive alignment action. After the second ‘bang’, the height is not controlled to the maximum height z_{max} , but is given a feedforward downward acceleration computed based on a circle around the ankle. Using the circle acceleration, the stance leg singularity can be approximated and the downward velocity at touchdown will not be as high as after a free fall, which takes point 4 of the preliminary observations into account.

Consider the distance from the ankle of the robot to the sagittal CoM position x_{ankle} and the maximum leg length l_{max} . If it is assumed that the CoM height z is located at the hip, the horizontal position relates to the vertical position as:

$$z^2 = l_{max}^2 - x_{ankle}^2. \quad (6-5)$$

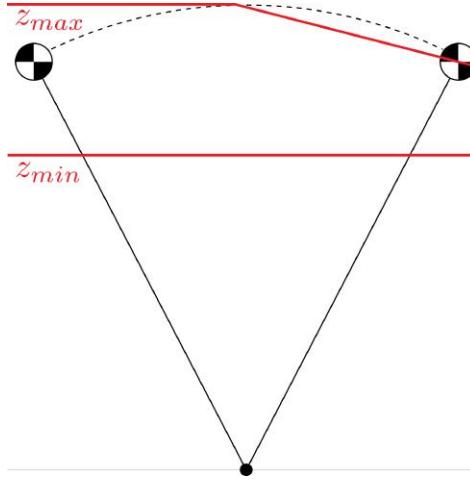


Figure 6-5: Height constraints throughout single support for the vertical motion controller while the robot is walking.

The vertical velocity resulting from this function reads as:

$$\dot{z} = -\frac{x_{ankle}\dot{x}}{\sqrt{l_{max}^2 - x_{ankle}^2}}. \quad (6-6)$$

The resulting vertical acceleration reads as:

$$\ddot{z} = \frac{\sqrt{l_{max}^2 - x_{ankle}^2}(-\dot{x}^2 - x_{ankle}\ddot{x}) - \frac{x_{ankle}^2\dot{x}^2}{\sqrt{l_{max}^2 - x_{ankle}^2}}}{l_{max}^2 - x_{ankle}^2}. \quad (6-7)$$

Assuming the sagittal acceleration \ddot{x} is zero, the desired vertical acceleration is computed as:

$$\ddot{z}_d = -\frac{\dot{x}^2}{\sqrt{l_{max}^2 - x_{ankle}^2}} - \frac{x_{ankle}^2\dot{x}^2}{(l_{max}^2 - x_{ankle}^2)^{1\frac{1}{2}}}. \quad (6-8)$$

Prepare Action

When the future errors are more suitable for using vertical CoM motion for balance, the CoM height is prepared for applying more force later by being lowered. The control action uses the time it takes to accelerate from the minimum height constraint to the maximum height constraint. This time uses the kinetic and potential energy:

$$z_{min} + \frac{1}{2}\ddot{z}_c t_{z_{min} \rightarrow z_{max}}^2 + \frac{1}{2} \frac{(\ddot{z}_c t_{z_{min} \rightarrow z_{max}})^2}{\alpha_{\ddot{z}_c} \ddot{z}_c} = z_{max}, \quad (6-9)$$

where $t_{z_{min} \rightarrow z_{max}}$ is the time from the minimum height constraint to the maximum height constraint, considering a zero initial vertical velocity. The solution for this time reads as:

$$t_{z_{min} \rightarrow z_{max}} = \sqrt{\frac{2(z_{max} - z_{min})}{\ddot{z}_c + \frac{\dot{z}_c}{\alpha_{\ddot{z}_c}}}}. \quad (6-10)$$

This time is used to determine when the first ‘bang’ should be activated, after the CoM height is lowered. The known remaining time in single support t_r (the total single support time minus the time already spent in single support) is shortened by $t_{z_{min} \rightarrow z_{max}}$:

$$t_{z \rightarrow z_{min}} = t_r - t_{z_{min} \rightarrow z_{max}}, \quad (6-11)$$

where $t_{z \rightarrow z_{min}}$ is the time available to move from the current height z to the minimum height z_{min} . Using this time, at every control tick the desired acceleration \ddot{z}_d is computed by using the equation:

$$z + \dot{z}t_{z \rightarrow z_{min}} + \frac{1}{2}\ddot{z}_d t_{z \rightarrow z_{min}}^2 - \frac{1}{2} \frac{(\ddot{z}_d t_{z \rightarrow z_{min}} + \dot{z})^2}{\alpha_{\hat{z}_c} \ddot{z}_c} = z_{min}, \quad (6-12)$$

$$\underbrace{-\frac{1}{2} \frac{t_{z \rightarrow z_{min}}^2}{\alpha_{\hat{z}_c} \ddot{z}_c} \ddot{z}_d^2}_{a} + \underbrace{\left(\frac{1}{2} t_{z \rightarrow z_{min}}^2 - \frac{t_{z \rightarrow z_{min}} \dot{z}}{\alpha_{\hat{z}_c} \ddot{z}_c} \right) \ddot{z}_d}_{b} + \underbrace{z - z_{min} + \dot{z}t_{z \rightarrow z_{min}} - \frac{1}{2} \frac{\dot{z}^2}{\alpha_{\hat{z}_c} \ddot{z}_d}}_{c} = 0, \quad (6-13)$$

which has the negative solution:

$$\ddot{z}_d = \frac{-b + \sqrt{b^2 - 4ac}}{2a}. \quad (6-14)$$

This value for \ddot{z}_d is used until $t_{z \rightarrow z_{min}} < 0$, after which the positive ‘bang’ is activated.

6-3 Results

To test the presented control actions used by the vertical motion controller, push recovery is tested on Valkyrie in simulation. Additionally, the positive alignment action is tested on Atlas on hardware.

6-3-1 Valkyrie Simulation

The results on Valkyrie in simulation are obtained by pushing the robot in single support when the right foot is the support foot, as in the previous sections. Initially, the robot is pushed at entrance of single support. To test if the robot recovered, there are four more steps taken after the current step, and checked if the robot did not fall over. Additionally, pushes are applied in different moments in single support, using the notation $\Delta_{push,SS}$ for the fraction of the swing time when the push is applied. To have a more instant change in error, a push duration of $t_{push} = 0.03$ is chosen. The following parameters are used for the control law:

- $\xi_{e,min} = 0.03$ [m], the same value as is used for the standing tests;
- $\phi_{min} = -0.85$ [rad], $\phi_{max} = 0.85$ [rad] and $\delta_{min} = 0.04$ [m], which are selected based on results obtained after searching for maximum recoverable pushes at the beginning of single support using different values for the parameters;
- $\ddot{z}_c = 5.0$ [m/s²] and $\ddot{z}_{max} = 200$ [m/s³], to have a more reactive response than on the standing tests.

Maximum Recoverable Push

Like in the previous chapter, the maximum recoverable pushes are searched for every 5 degrees and the normalized impulse $i = \frac{\int F_{push} dt_{push}}{m_{robot}}$ is used. The results for both control setups are shown in Figure 6-6, with the radial lines showing the push directions and actions used by the vertical motion controller. The right foot is the support foot. For the push directions, 0 degrees is a rear push, 90 degrees is a push from the right, 180 degrees is a frontal push and 270 degrees is a push from the left. For these results, $\mathbf{r}_{cmp,d}$ is constrained to be inside the support polygon and is assumed to coincide with $\mathbf{r}_{cop,d}$. It can be observed that the actions have less effect when the push is applied later in single support, and often result in worse recovery than the default setup at $\Delta_{push,SS} = 0.5$. The recovery for a push around 200 degrees is worse when the vertical motion control law is enabled when $\Delta_{push,SS} > 0.0$.

In Figure 6-7, the maximum recoverable pushes are shown when $\mathbf{r}_{cmp,d}$ is constrained to be inside the polygon, like in the previous figure, and when $\mathbf{r}_{cmp,d}$ is allowed to leave the polygon 0.05 [m]. This distance outside the polygon is commonly used at the Institute for Human and Machine Cognition (IHMC) [42]. Allowing $\mathbf{r}_{cmp,d}$ to leave the polygon requests angular momentum rate from the robot to achieve the desired linear momentum rate. However, how much of the desired linear momentum rate is achieved is highly depending on the weights used in the whole-body quadratic program (QP), which are presented in Appendix A. Note how the plots with larger possible $\mathbf{r}_{cmp,d}$ locations seem to match shape with the plots where $\mathbf{r}_{cmp,d}$ is constrained to be inside the polygon. Also note how for push directions coming from the back, the recovery is similar for the default controller with larger $\mathbf{r}_{cmp,d}$ locations compared with the vertical motion controller with $\mathbf{r}_{cmp,d}$ constrained to be inside the polygon for push moments until $\Delta_{push,SS} = 0.3$.

Comparison of Equal Push

A deeper evaluation is again made for the responses after an equal push for certain push directions. A rear and frontal push at the start of single support are chosen to make a deeper evaluation of. The default setup has a recoverable impulse of $i = 0.156$ [m/s] and 0.315 [m/s] respectively for these push directions. In Figure 6-8, time responses are shown for these applied impulses for both control setups. For the vertical motion controller, the positive alignment action is used for the rear push and the prepare action is used for the frontal push, which can also be observed in Figure 6-6.

For the rear push, the desired vertical linear momentum rate $\dot{\mathbf{i}}_{d,z}$ from a circle is clearly visible after the second ‘bang’ of the vertical motion controller. The maximum height violates the maximum allowed height for the controller slightly halfway single support. In the sagittal reference and estimated ICP plots, it is visible that the ICP error of the vertical motion controller remains smaller after the push is applied, compared to the default setup.

For the frontal push, the positive ‘bang’ near the end of single support is clearly visible in $\dot{\mathbf{i}}_{d,z}$ when enabling the vertical motion controller. Note that after the frontal push, $\dot{\mathbf{i}}_{d,z}$ has fairly low negative values near the end of single support with the default setup. The minimum height of the trajectory stays above the minimum height constraint, when using the vertical motion control law. From the reference and measured ICP in the sagittal direction, it can again be observed that the final ICP error is smaller for the vertical motion controller.

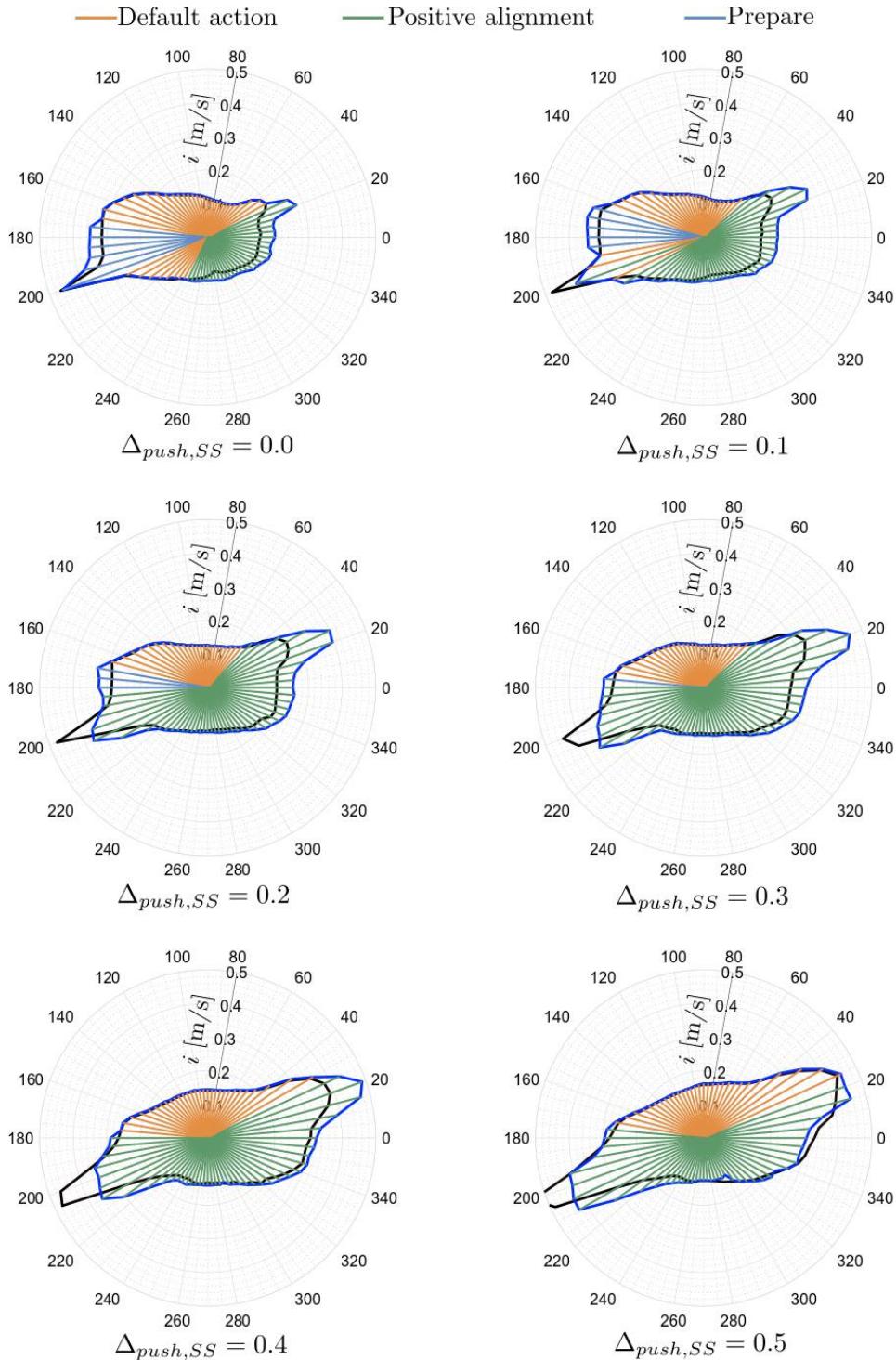


Figure 6-6: Polar plots of the maximum recoverable impulses i (radius) for the default controller (black) and the vertical motion controller (blue) for pushes applied at different moments in single support. The right foot is the support foot and 0 degrees corresponds with a push from the back. The colors of the radial lines show the actions used by the vertical motion control law.

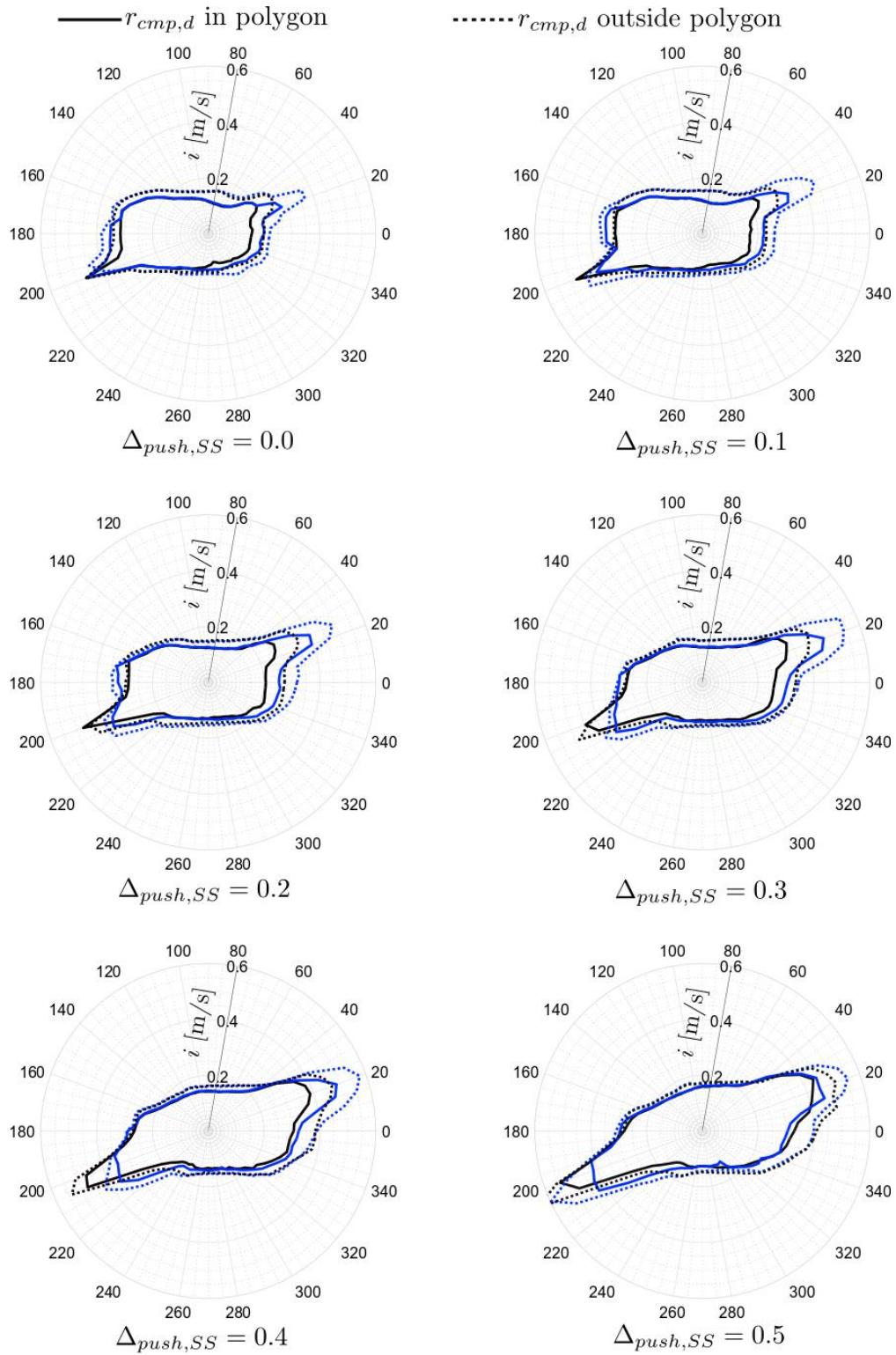


Figure 6-7: Polar plots of the maximum recoverable impulses i (radius) for the default controller (black) and vertical motion controller (blue). The right foot is the support foot and 0 degrees corresponds with a rear push. A comparison is shown for when the desired CMP is constrained to be inside the polygon, and when the desired CMP is allowed to leave the polygon 0.05 [m].

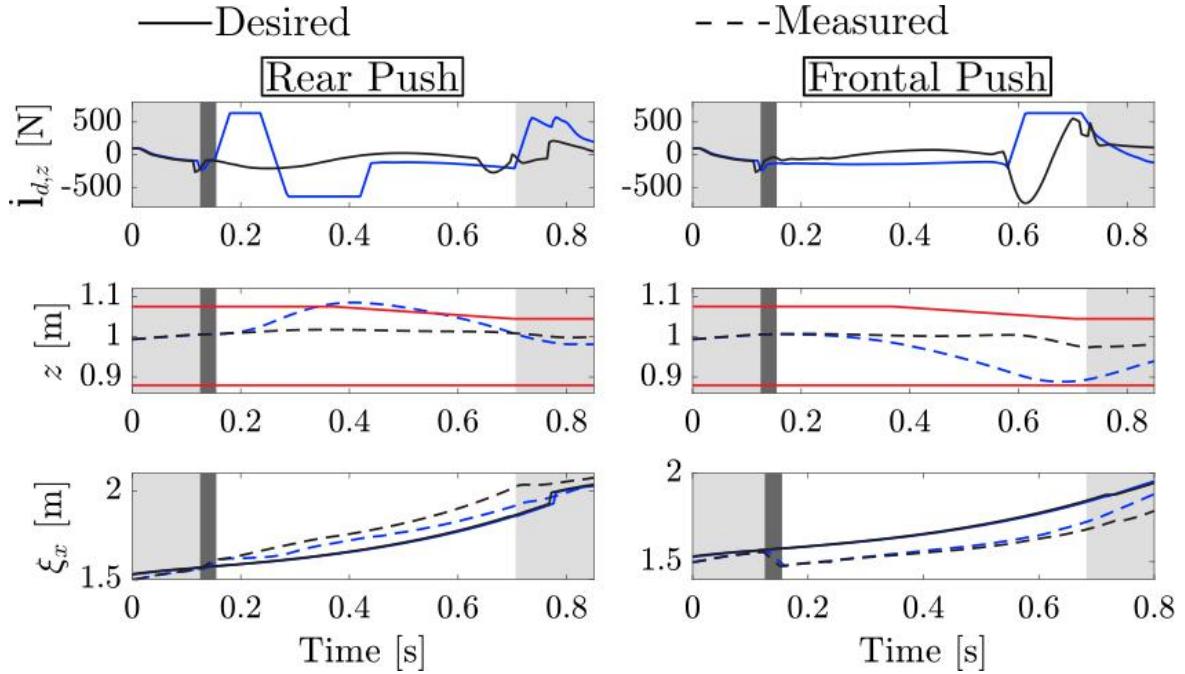


Figure 6-8: Time responses after a rear push of $i = 0.156$ [m/s] and a frontal push of 0.315 [m/s] are applied on Valkyrie for the default setup (black) and the vertical motion controller (blue). The push is applied in the dark gray area and the light gray area shows when the robot is in double support.

6-3-2 Atlas Hardware

Additionally, some hardware tests on Atlas are conducted during walking, triggering the positive alignment action. These tests are conducted on Atlas, as Atlas is able to walk faster than Valkyrie due to its more powerful actuation. Walking faster allows for CoM positions at further horizontal distance from the CoP positions, in which CoM height variations can have more effect on balance control. Unlike in the Valkyrie simulations, a step length of 0.4 [m] is used. The same single support and double support times as in the Valkyrie simulations are used, which are about the maximum what the robot could do. Longer step lengths or shorter step times resulted in the hydraulic pump to fail in generating the desired fluid pressure, which resulted in the robot to fall.

In Figure 6-9, a three-dimensional space (3D) plot of the walking pattern and a time response plot of $\dot{z}_{d,z}$ are depicted for one test result. Intermediate local pendulums between $\mathbf{r}_{cmp,d}$ and the CoM are depicted in the walking pattern after the push is applied with a constant time interval of 0.2 [s]. The yellow lines show the projections on the xz -plane of the pendulums. The letters above the yellow lines match with letters below the images in Figure 6-10; the pendulums are the $\mathbf{r}_{cmp,d} - \mathbf{c}$ configurations for the images. At a the push is applied and at b released. The height change and the bang-bang control law of the positive alignment action are clearly visible. Instead of using the acceleration from a circle after the second ‘bang’, a constant feedforward value is used on desired acceleration, because this was found to be simpler for tuning on the robot.

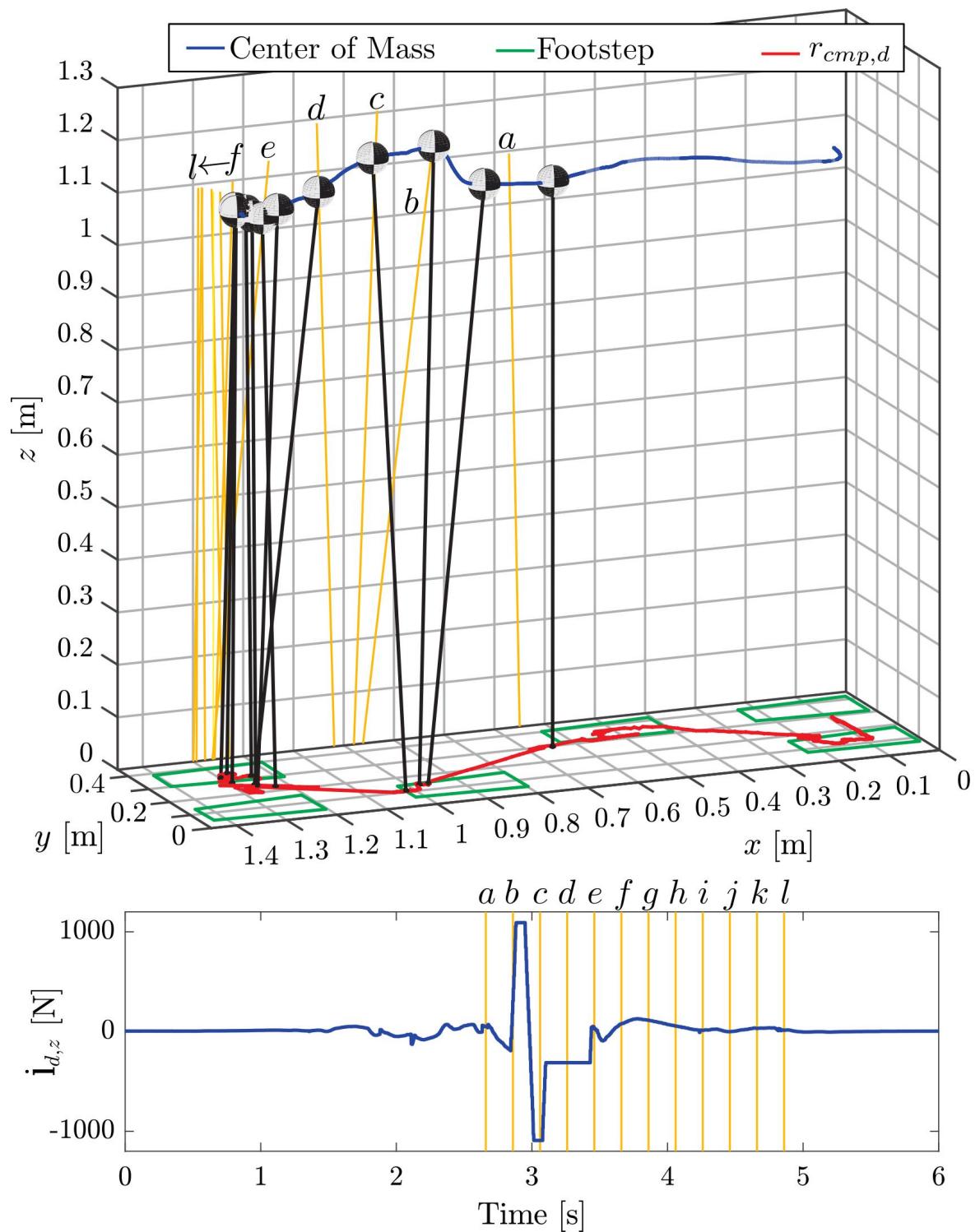


Figure 6-9: Visualization in 3D of the CoM and $r_{cmp,d}$ trajectory during the walking push recovery test (top). The yellow lines are the pendulums projected on the xz -plane. The letters above the yellow lines correspond with the letters below the images in Figure 6-10 and show the $r_{cmp,a-c}$ configurations. At a , the push is applied and at b the push is released. Also, $i_{d,z}$ over time is shown (bottom).

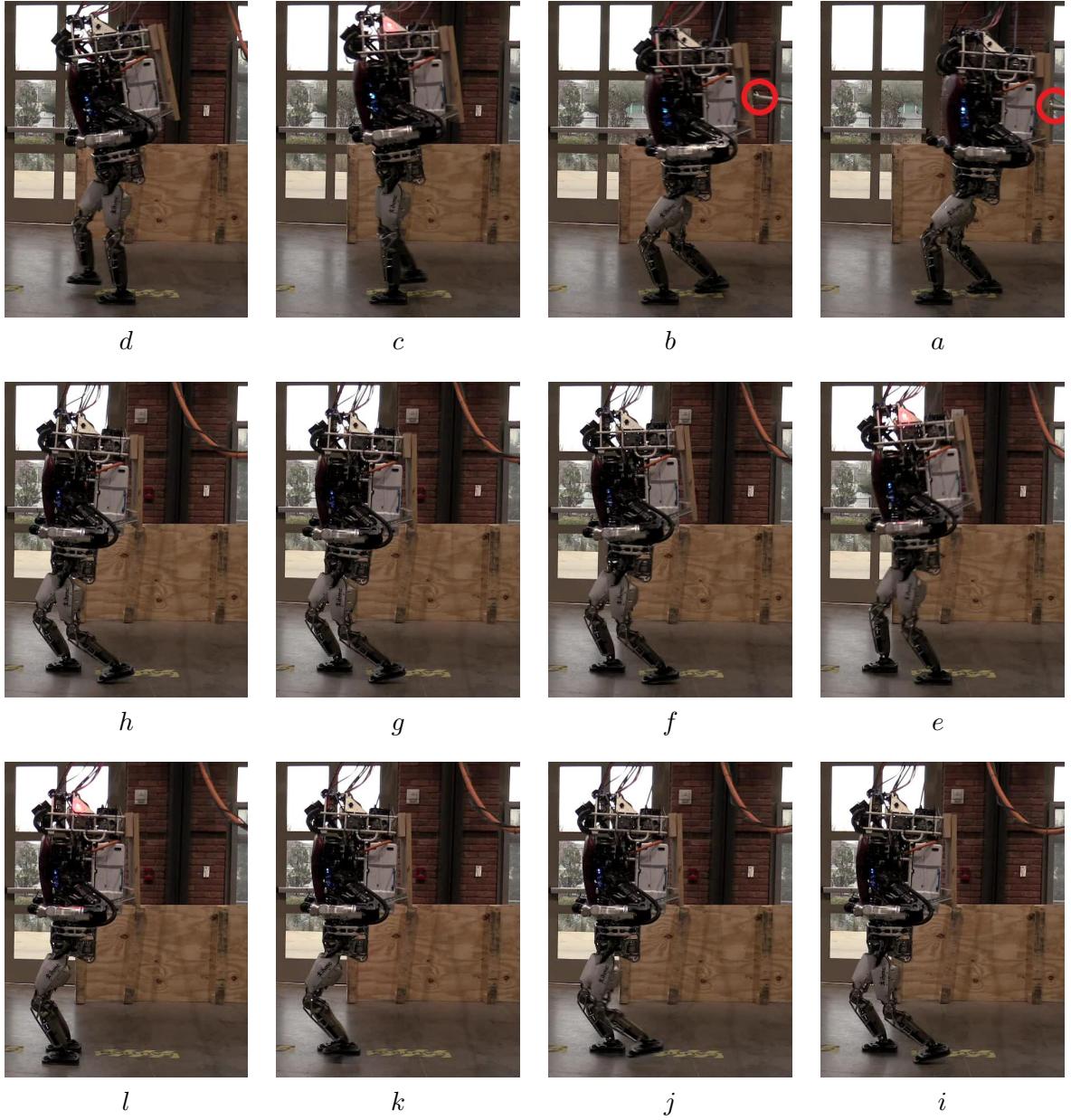


Figure 6-10: Time-lapse of Atlas recovering from a push using the positive alignment action. The letters below the images match with the corresponding letters above the yellow lines in Figure 6-10. The push rod tip is encircled in red.

6-4 Discussion

Chapter 6 presented a control law for the use during walking, extending the method introduced in Chapter 5. This control law uses predefined strategies, that can be heuristically be selected. These strategies rely on the predefined step time and step location used in the control framework. Like in the previous chapter, control actions are chosen based on the violation of certain thresholds.

The effective distance δ and the alignment angle ϕ allow for making a heuristic decision for the use of CoM height variation for balance control in different configurations. These variables are depending on the already existing ICP control law. Based on these variables, a control action is selected. Any misalignment with the error would result in additional error when using CoM height variation for balance. This error is controlled with future $\mathbf{r}_{cmp,d}$ positions. For the future, it could be interesting to already adjust for these additional errors by modifying $\mathbf{r}_{cmp,d}$ in advance.

From the results obtained for the step time and length considered, it seemed that the proposed control actions in the beginning of single support resulted in the most improvement compared to strategies that do not use CoM height variation for balance. The vertical motion controller performs worse than the default setup around the 200 degree push direction for all push moments in swing considered. This may be a results of the chosen control actions based on the thresholds ϕ_{min} , ϕ_{max} and δ_{min} , which are constant during the swing phase.

For the Atlas tests on hardware, the control law was difficult to compare with the default setup, and therefore the choice was made to only show one example of a result using the vertical motion controller. For the standing tests in Chapter 5, it was assumed that the direction orthogonal to the push is constrained and the recovery problem is in two-dimensional space (2D). For the walking tests, the CoM is located outside the support polygon if the push is applied and can also move orthogonal to the push direction. Furthermore, the robot is moving while the push is applied during walking. Any small difference in the moment when the push is applied, the push location or the push direction can make a large difference in how the robot recovers. Therefore, a reliable comparison with the default setup on hardware for the walking tests could not be made and might be interesting to consider in the future.

Chapter 7

Conclusion

The objective of this work was to improve balance control of a humanoid robot using center of mass (CoM) height variation. In this work, novel capture regions for the variable height inverted pendulum (VHIP) model were proposed and compared with the linear inverted pendulum (LIP) model, which addressed the theoretical part of the research objective. Furthermore, control actions that use CoM height variation for balance were presented and results were shown on hardware on humanoid robots in comparison with predefined CoM height approaches. For Valkyrie, it was observed that balance was improved using CoM height variation, which addressed the applied part of the research objective.

For the VHIP model, capture regions were proposed in Chapter 3 considering a unilateral contact constraint, after which height constraints and force constraints were added. It was observed that the capture region becomes smaller after addition of constraints. Also, a comparison with the LIP and LIP plus flywheel capture regions was made, which gives a high-level measure of the potential effects of CoM height variation. The presented capture regions were derived under the assumption that kinematic limits and joint torque limits of the robot can be approximated with a constraint on minimum and maximum height and vertical acceleration respectively.

Because the vertical force constrained capture regions are computed numerically, there was experimented with another control law, a model predictive control (MPC) law, in Chapter 4. Based on the shape of the control input of the MPC however, which is constrained to be a polynomial function, there was chosen to not use this control law in applications in Chapter 5 and Chapter 6.

Similar to the control law used to compute the vertical force constrained capture regions, a bang-bang control law was designed for implementation in a momentum-based whole-body control framework in Chapter 5. This control law is activated when a predefined threshold is met. With this control law, push recovery tests were conducted on NASA's Valkyrie and Boston Dynamics' Atlas, while the robots were standing. The results for Valkyrie in simulation showed that push recovery improved 9% when pushing the robot in the back and 4% when pushing from the side when the bang-bang control law on vertical CoM motion

was used. Remarkably, push recovery was 7% worse after a frontal push when enabling the bang-bang controller. The rear push tests were also conducted on hardware, using a push stick and a load sensor, where an average of 7% increase in maximum recoverable push was observed. The vertical force constrained capture position for the same height change and vertical acceleration differed approximately 4% from the LIP capture point (LIPCP), so differences were observed between the VHIP model and the results on Valkyrie. Additional hardware tests were conducted on Atlas using a medicine ball on a rope. However, recovery did not improve noticeably. Different initial heights for the robot were tried, as well as a tuning of a joint torque control gain, which had no noticeable effect. In simulation however, recovery did improve for Atlas when enabling the bang-bang control action.

Similar to the bang-bang control action, three actions were proposed for the use during walking in Chapter 6. Using the two presented variables, the alignment angle and the effective distance, a control action was chosen heuristically based on outputs of instantaneous capture point (ICP) control. Compared to a constant height control approach, recovery improved the most when pushing the robot in the back or from the front in the first part of the swing phase. On hardware, evaluation of the proposed control law was difficult, because of the additional uncertainties compared to the standing tests. Therefore, only an example was shown for a control action on hardware on Atlas while the robot is walking.

7-1 Recommendations

The results presented in this work have demonstrated that CoM height variations can improve balance control. There are however shortcomings, both on the theoretical as well as the applied side of the proposed research. In the following sections, recommendations for future work are presented. First, recommendations for extension of the proposed approaches are presented, after which a broader outlook on future work is briefly presented.

7-1-1 Extending the Proposed Approach

In this section, opportunities for improvement of the presented theory, tests and results are presented.

Extension of Capture Regions

The unilateral contact and height constrained capture regions give bounds on the capture region. However, these cannot directly be used in a control law, as impacts are considered in the computation. The vertical force constrained capture points can be used in control, but use numerical integration to find future state information. It would be interesting to explore closed-form solutions for a force constrained capture problem, without overly constraining the VHIP like in [10] and [13]. With a closed-form solution for example, the control law used in application in this thesis could be predictive, as a vertical force constrained capture position could be computed on every time instance.

Improving Push Recovery Tests

Balance control of the robots was tested in this work by testing push recovery. The push stick with load sensor [43] measures push force accurately, but a person pushing the robot is in general not able to apply a desired force precisely. With the tests with the medicine ball, the ball location could be put relatively precise. However, the push duration on the robot using these tests cannot be adjusted and is quite short, which resulted in high impacts on the robot. Furthermore, stretch in the rope and in the ball can change the CoM height of the ball. Also, the transfer of the energy of the ball to the robot depends on the damping properties of the ball and the robot. For future push recovery tests, it could be interesting to use a device that can accurately apply force according to a desired profile over time.

Improving State Estimation and Center of Pressure Control

Applying the methods presented in this thesis requires good state estimation and control of the center of pressure (CoP). In some experiments on Atlas it was found that performance did not improve when applying the presented control law, even though performance could improve according to the VHIP model and the obtained simulation results. This lack of performance could be related to a CoP error, which could be caused by the additional movement of the robot when the presented method was used. By improving state estimation and CoP control, the theoretically predicted improvements could potentially be better achieved in practice.

Standing Tests for Lowering Center of Mass Height

The standing push recovery tests presented in this work all use an increase in CoM height for balance control, as the initial CoM position of the robot was inside the support polygon at the moment the push was applied. With the walking tests, the CoM height was lowered. However, the walking tests were difficult to test on hardware, because of the increased number of uncertainties. It could be interesting to find a test setup, where the robot should lower the CoM height to balance to a standing configuration. The robot can be given an initial velocity when the CoM is outside the support polygon, that lowering the CoM height would be needed to balance. This test setup would be comparable with a human landing after a long jump.

Analysis of Results

In this work, the data obtained from the robots was predominantly analyzed based on time response and phase plots. It could be interesting to perform additional analysis, as for example analyzing the energy consumption on the robot when using the different control strategies. The energy consumption could be determined by measuring the electric current and voltage going to the robot. Energy consumption can, for example, be an additional decision variable for choosing between balancing strategies.

7-1-2 Outlook

This work contributes merely a small part to balancing strategies for humanoid robots. For the future, it could be interesting to analyze CoM height variations in legged systems further. It would be interesting to investigate when humans use height variation, and why the strategy is chosen instead of the ‘hip strategy’ in such scenarios for example. Furthermore, it would be interesting to see different balancing strategies combined with height variation on humanoid robots. The decision making in balancing strategies, under the constraints of kinematic limits, force limits, disturbances and terrain remains a broad research area to explore.

Appendix A

Test Parameters

In Table A-1, control framework parameters are shown which are used on the Valkyrie tests. If the weights are denoted as a vector, the entries are the xyz -components. The wrench cone basis vector multiplier ρ is used to compute the desired ground reaction force (GRF), as explained in Section 2-2.

Table A-1: Control framework parameters for the Valkyrie tests.

Task group	Task	Gains	Weights
ICP control	-	$k_\xi = 2$	-
Height control	-	$k_p = 50, k_d = 14$	-
Centroidal momentum rate	Linear	-	[0.05, 0.05, 0.01]
Motion angular	Pelvis	$k_p = 100, k_d = 16$	[5.0, 5.0, 5.0]
Motion angular	Chest	$k_p = 100, k_d = 16$	[15.0, 10.0, 5.0]
Motion angular	Foot stance	$k_p = 200, k_d = 28$	[2.0 2.0 2.0]
Motion linear	Foot stance	$k_p = 0, k_d = 0$	[80.0 80.0 80.0]
Motion angular	Foot swing	$k_p = 200, k_d = 20$	[2.0 2.0 2.0]
Motion linear	Foot swing	$k_p = 150, k_d = 17$	[80.0 80.0 80.0]
Motion joint	Spine	$k_p = 50, k_d = 11$	10.0
Regularization	Joint acceleration	-	$5 \cdot 10^{-3}$
Regularization	Joint jerk	-	$1.6 \cdot 10^{-6}$
Regularization	ρ	-	$1.0 \cdot 10^{-5}$
Regularization	ρ rate	-	$5.0 \cdot 10^{-8}$

Appendix B

2D Analysis from Model to Robot

Contributing to this thesis, the paper included in the following pages is submitted for publication in *IEEE Robotics and Automation Letters (RA-L)* with the *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* option on January 6, 2019. The contributions in the paper are closely related to the work presented in Chapter 3 and Chapter 5.

Balancing using Vertical Center of Mass Motion: A 2D Analysis from Model to Robot

Boris J. van Hofslot^{1,2}, Robert Griffin¹, Sylvain Bertrand¹, and Jerry Pratt¹

Abstract—Balancing strategies for humanoid robots often include center of pressure control (‘ankle’ strategies), change of body angular momentum (e.g., ‘hip’ strategies) and taking a step. In this work, we propose using vertical center of mass motion as an additional input for balance control. We walk through the process of analyzing simple 2D models, after which we analyze the effects of those models after application on a real robot. First, we specify analytic, theoretical capture regions under unilateral contact and height constraints only. Second, we add a vertical acceleration constraint and come to a simple control law for implementation. Third, we implement the control law in our momentum-based whole-body control framework. We test push recovery while standing on NASA’s Valkyrie humanoid robot and compare with a constant height controller, and show that recovery can be improved using vertical motion. Furthermore, we discuss the differences that can be observed after application of a simple model on a robot.

I. INTRODUCTION

Keeping balance is a fundamental problem in humanoid robotics. Throughout the years, many conditions and expressions have emerged for analyzing the ability of the robot to stabilize. Examples are the capture point and capture region [1], [2], stability regions [3], the divergent component of motion [4] and the boundedness condition [5], which all link to the energetics of the pendulum-based model and its ability to converge.

These conditions commonly rely on a linear inverted pendulum (LIP) model, with optionally a mass with inertia to model the robots angular momentum. The LIP model provides fast, closed-form solutions when integrating over time. This results in the center of mass (CoM) height usually to be fixed in the dynamic planning problem. Vertical center of mass motions are considered as pre-defined and deviations from the dynamic model are considered as disturbances. Those disturbances are commonly controlled with ‘ankle’ strategies, i.e., moving the center of pressure (CoP) or pendulum base, and to a lesser extent, with ‘hip’ strategies: change of body angular momentum. These strategies can be generated by using, e.g., a momentum-based whole-body control framework [6], [7], [8], which determines center of

This work was supported by NASA’s Johnson Space Center under Grant 80NNSC18M0071.

¹The author is with the Florida Institute for Human and Machine Cognition, 40 S Alcaniz St, 32502 Pensacola FL, United States {bvanhofslot, rgriffin, sbertrand, jpratt}@ihmc.us

²The author is with the Department of Cognitive Robotics, Delft University of Technology, Mekelweg 2, 2628 CD Delft, Netherlands b.j.vanhofslot@student.tudelft.nl

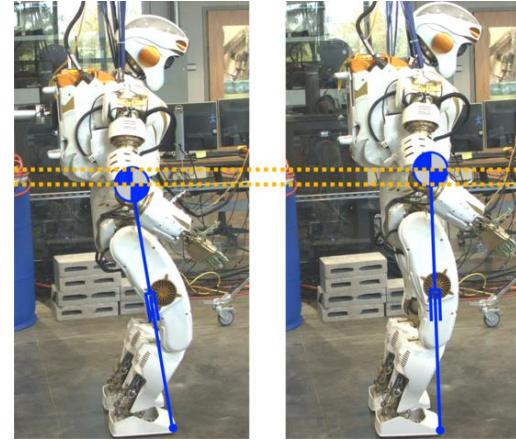


Fig. 1. Visualization of the idea in this paper.

pressure position and angular momentum rate by optimizing between desired momenta and motion objectives.

Although the use of a predefined height trajectory has advantages, it can be overly constraining. In cases where traditional balancing strategies are saturated and the robot is in risk of falling, it is interesting to explore additional methods for the robot to recover. Vertical CoM motion can be used to generate additional horizontal force on the CoM, which can improve balance.

Recently, efforts have been made to use vertical CoM motions for balance control. In [9], an analytic model predictive controller is derived in 2D from the *orbital energy* proposed in [10]. Regions of attraction for this controller are investigated, as well as limitations on the region in which recovery is possible for a variable height inverted pendulum model under the constraint of unilateral contact only. In [11], different 2D strategies are proposed for multi-step recovery using vertical CoM motion. In [12], a model predictive control law for 3D capture trajectories is proposed using a nonlinear solver. The authors manage to solve the nonlinear control problem online using a variable height inverted pendulum model. However, applied results on hardware are not shown yet.

In this paper, we analyze capture regions of a simple 2D model. Subsequently, we implement a control law based on this model on NASA’s Valkyrie and test push recovery (see Fig. 1). We propose capture regions by adding constraints to a base model. We add a contact unilaterality constraint, followed by height constraints, from which we derive ana-

lytic capture regions. We add a vertical force constraint and formulate a bang-bang control law on vertical acceleration, after which the solution to a capture region needs to be computed numerically. Furthermore, we analyze the differences between the capture regions. Finally, we implement the bang-bang control law in our momentum-based control framework [8]. We test push-recovery on NASA's Valkyrie [13] and compare with a setup that has a constant height objective and only uses CoP. Furthermore, we discuss the differences that we observe when going from a simple model to application on a robot.

The remainder of the paper is structured as follows. In Section II, we give a short overview of balancing strategies and *capture*. In Section III, we derive capture regions for a pendulum with variable height. We test balancing on NASA's Valkyrie by applying pushes in Section IV. Finally, in Section V, we conclude and give our outlook on balance control for humanoids.

II. MODEL & CAPTURE

Throughout the paper, we disregard stepping and focus on comparing ‘0-step’ capture [2]. Our goal is to explore the effects of height variation in balancing tasks, and how these compare to constant height control approaches. To consider vertical motions, we use a variable height inverted pendulum model, which has the following dynamics:

$$\ddot{x} = \frac{x}{z} u, \quad (1)$$

where $u = g + \dot{z}$ is the normalized vertical force, x is the position of the point-mass relative to the CoP and z the height of the mass.

In this paper, we use the term capture region [1] to describe the set of CoP locations where balance can be achieved. Also, the capture point as introduced by Pratt *et al.* is considered, only this is for comparison denoted as:

$$x_{cp,lip} = \sqrt{\frac{z_0}{g}} \dot{x}_0, \quad (2)$$

where z_0 is the initial height and \dot{x}_0 the initial horizontal velocity. To avoid confusion, we use the term *capture position* to describe a point where the current state and the resulting trajectory will lead to convergence of the pendulum-based model. We denote a capture position as a positive value:

$$x_{cp} = |x_0|, \quad \text{if } x_f = 0 \quad \text{and } \dot{x}_f = 0, \quad (3)$$

where x_{cp} is the capture position, x_f the final horizontal position and \dot{x}_f the final horizontal velocity. We use an initial horizontal velocity of greater than zero and that $x\dot{x} < 0$ for any capture trajectory [9].

III. CAPTURE REGIONS

This section proposes bounds on the capture position (3). The dynamics of (1) are considered. For simplicity and comparison with the LIP capture point (2), we take the initial vertical velocity $\dot{z}_0 = 0$. In each subsection, we add constraints to come to a more realistic model.

A. Unilateral Contact Constraint

Considering the constraint of contact unilaterality only, the capture region is bounded by the current position and the ballistic touchdown point:

$$x_{cp,unilateral} \in (0, x_{bal}], \quad \forall u \geq 0, \quad (4)$$

where x_{bal} is the ballistic touchdown point. This is the location where the point-mass would intersect with the ground plane after a free fall. $x_{cp,unilateral}$ is the capture position under unilateral contact constraint only. The proof for this region is given in [9]. For the zero initial vertical velocity, the ballistic touchdown point reads as:

$$x_{bal} = t\dot{x}_0 = \sqrt{\frac{2z_0}{g}} \dot{x}_0 = \sqrt{2} x_{cp,lip}. \quad (5)$$

The region can be interpreted as follows. At an infinitesimally small distance on the side of CoM in the direction of its horizontal velocity, there exists an infinite impact of the leg that stops the horizontal motion of the CoM. On the other side of the capture region, the leg, without constraints on height, can apply an impact when the mass is at ground height that stops the motion of the mass.

B. Addition of Height Constraints

To take kinematic limits of the robot into account, we derive capture positions under a *minimum* and *maximum* height constraint respectively. We consider a combination of impacts of the leg, the LIP capture trajectory and the ballistic trajectory, such that analytic capture positions can be found.

Preliminary, we temporally set $\dot{z}_0 \neq 0$ to calculate the influence of an impact on $x_{cp,lip}$. We can use this in the next paragraph to derive a capture position under a minimum height constraint. Considering an initial negative vertical velocity $\dot{z}_0 < 0$ that is driven to zero by a vertical impact, the influence on the LIP capture point is:

$$x_{cp,I} = \sqrt{\frac{z_0}{g}} \left(\dot{x}_0 + \frac{x_{cp,I}}{z_0} \dot{z}_0 \right), \quad (6)$$

$$= \frac{z_0}{\sqrt{g z_0 - \dot{z}_0}} \dot{x}_0, \quad (7)$$

where $x_{cp,I}$ is the impact influenced capture point from an initial impact that results in $\dot{z} = 0$.

1) *Minimum height*: Under a minimum height constraint z_{min} , we can find a capture position from which the trajectory ‘just’ touches the constraint. We first let the mass follow the ballistic trajectory, after which it is vertically stopped by the impact influenced capture point:

$$x_{cp,z_{min}} = x_{bal}(\delta z_{min}) + x_{cp,I}(z_{min}, \dot{z}_{z_{min}}), \quad (8)$$

where $x_{cp,z_{min}}$ is the capture position over the minimum height constraint, $x_{bal}(\delta z_{min})$ the horizontal position after the ballistic fall $\delta z_{min} = z_0 - z_{min}$ and $x_{cp,impact}(z_{min}, \dot{z}_{z_{min}})$ is $x_{cp,impact}$ after the ballistic fall. The velocity at the moment the ballistic trajectory hits the constraint is:

$$\dot{z}_{z_{min}} = -\sqrt{2g\delta z_{min}}, \quad (9)$$

where $\dot{z}_{z_{min}}$ is the vertical velocity at z_{min} . Using (5), (7), (8) and (9), the capture position over the minimum height constraint becomes:

$$x_{cp,z_{min}} = \left(\sqrt{\frac{2\delta_{z_{min}}}{g}} + \frac{z_{min}}{\sqrt{gz_{min}} + \sqrt{2g\delta_{z_{min}}}} \right) \dot{x}_0. \quad (10)$$

2) *Maximum height:* Also under a maximum height constraint z_{max} , an analytic capture position can be found. We consider a vertical impact by the leg at the initial position $x = x_0$. This impact is of such magnitude, that the mass is exactly at the maximum height constraint, if it is at its apex. After the vertical velocity of the mass is driven to zero by gravity, we apply $x_{cp,lip}$. This point reads as:

$$x_{cp,z_{max}} = \left(t_{\dot{z}>0} + \sqrt{\frac{z_{max}}{g}} \right) \dot{x}_{0,I}, \quad (11)$$

where $x_{cp,z_{max}}$ is the capture position following the maximum height constraint, $t_{\dot{z}>0}$ is the time $\dot{z} > 0$ and $\dot{x}_{0,I}$ is the initial velocity influenced by the impact of the leg. The vertical velocity resulting from the impact that lets the mass just touch z_{max} is:

$$\dot{z}_I = \sqrt{2g\delta_{z_{max}}}, \quad (12)$$

where $\delta_{z_{max}} = z_{max} - z_0$. Noting that $t_{\dot{z}>0} = \frac{\dot{z}_I}{g}$ and filling in (11) gives:

$$x_{cp,z_{max}} = \left(\frac{\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}} \right) \left(\dot{x}_0 - \frac{x_{cp,z_{max}}}{z_0} \dot{z}_I \right). \quad (13)$$

Bringing $x_{cp,z_{max}}$ to the left-hand side and filling in (12) gives:

$$x_{cp,z_{max}} = \frac{\frac{\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}}}{1 + \left(\frac{\dot{z}_I}{g} + \sqrt{\frac{z_{max}}{g}} \right) \frac{\dot{z}_I}{z_0}} \dot{x}_0, \quad (14)$$

$$= \frac{z_0(\sqrt{2\delta_{z_{max}}} + \sqrt{z_{max}})}{\sqrt{g}(z_0 + 2\delta_{z_{max}} + \sqrt{2z_{max}\delta_{z_{max}}})} \dot{x}_0. \quad (15)$$

3) *Bounds:* We will show that the capture positions $x_{cp,z_{min}}$ and $x_{cp,z_{max}}$ are also the outer bounds on the capture region.

Lemma 1: Considering the dynamics of (1), $\dot{z}_0 = 0$, minimum height constraint z_{min} and maximum height constraint z_{max} , $x_{cp,z_{min}}$ and $x_{cp,z_{max}}$ are the outer bounds on the capture region.

Proof: For any capture position x_{cp} , $\dot{x} < 0$ [9] and $0 > x_0 \geq -x_{bal}$ (4). We use that $x \leq 0, \forall t$ and $x \rightarrow 0$ along any trajectory. From (1), and $z > 0$, it follows that any input u will slow \dot{x} down. Showing that $\frac{x}{z} \rightarrow 0, \forall t$ will prove that $u = 0$ for the longest possible time t will lead to the farthest x_{cp} , and a maximum u at the earliest possible t will lead to the closest x_{cp} .

For $u = g$, z remains constant and $\frac{x}{z} \rightarrow 0$. For $u > g$, z will grow and $\frac{x}{z} \rightarrow 0$. If $u < g$, we can show with the derivative of $\frac{x}{z}$ that this is always increasing:

$$\frac{d\frac{x}{z}}{dt} = \frac{z\dot{x} - x\dot{z}}{z^2}, \quad (16)$$

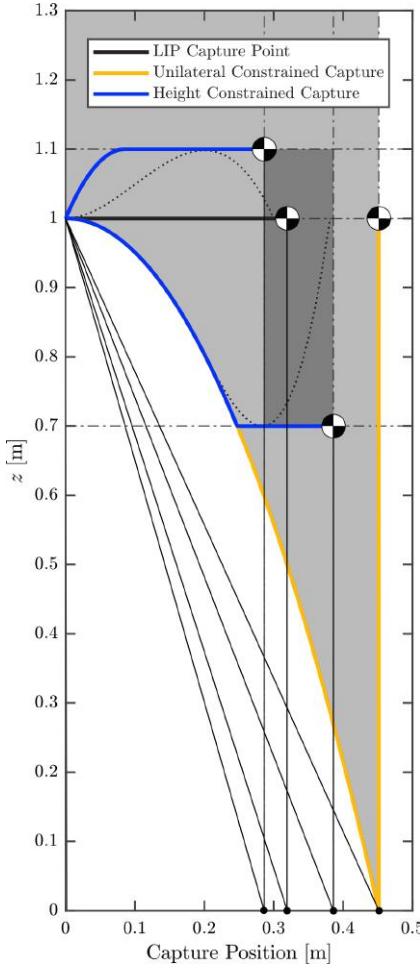


Fig. 2. Visualization of the analytic capture regions for $\dot{x}_0 = 1$ [m/s] and $\dot{z}_0 = 0$ [m/s]. The light gray area shows the unilateral contact constrained capture region (4). The dark gray area shows the height constrained capture region (Lemma 1) for $0.7 < z < 1.1$ [m]. The dotted plots are made with the *orbital energy controller* of [9] and show that the final points are inside the height constrained region.

where $x \leq 0$ and $z\dot{x} \geq 0$. Taking the extreme case $u = 0$ leads to:

$$z\dot{x} - x\dot{z} = (z_0 - \frac{1}{2}gt^2)\dot{x}_0 + (x_0 + \dot{x}_0 t)gt, \quad (17)$$

$$= (z_0 + \frac{1}{2}gt^2)\dot{x}_0 + x_0 gt. \quad (18)$$

Noting that all terms are positive except for x_0 , which has the largest negative value for $x_0 = -x_{bal}$:

$$(z_0 + \frac{1}{2}gt^2)\dot{x}_0 - \sqrt{\frac{2z_0}{g}}\dot{x}_0 gt = \dot{x}_0 \left(\sqrt{\frac{1}{2}gt} - \sqrt{z_0} \right)^2, \quad (19)$$

which is always greater than or equal to zero for all t . ■

In Fig. 2, the discussed capture regions are visualized. The LIP capture point lies inside the height constrained region, which lies inside the unilateral contact constrained region.

C. Addition of Vertical Force Constraints

We assume that the robot specific limitations on joint torques can be approximated with a minimum and maximum vertical force on the CoM. In doing so, we add constraints on the minimum and maximum vertical acceleration to the dynamics (1). From Lemma 1, any vertical acceleration extremum at the earliest convenience will lead to staying closer to a height constrained bound. By inserting a constraint on vertical acceleration, an analytic solution for a capture position is not available anymore and needs to be solved numerically¹.

In [1], [3], [2], a bang-bang control law is used to regulate the angular momentum in the body of model. Instead, we use a bang-bang control law on the input u (1) to regulate the vertical dynamics:

$$u = g + \ddot{z}_{c,1}H(t) - (\ddot{z}_{c,1} - \ddot{z}_{c,2})H(t - t_1) - \ddot{z}_{c,2}H(t - t_2), \quad (20)$$

where $[\ddot{z}_{c,1}, \ddot{z}_{c,2}]$ are the first and second constant control inputs and have opposite signs. $H(\cdot)$ is the Heaviside step function and

$$t_1 = \sqrt{\frac{2(z_{const} - z_0)}{\ddot{z}_{c,1} - \frac{\ddot{z}_{c,1}^2}{\ddot{z}_{c,2}}}}, \quad (21)$$

which is the solution of:

$$z_0 + \frac{1}{2}\ddot{z}_{c,1}t_1^2 - \frac{1}{2}\frac{(\ddot{z}_{c,1}t_1)^2}{\ddot{z}_{c,2}} = z_{const}, \quad (22)$$

where $z_{const} = z_{min}$ if $\ddot{z}_{c,1} < 0$ and $z_{const} = z_{max}$ otherwise. The time $t_2 = (1 - \frac{\ddot{z}_{c,1}}{\ddot{z}_{c,2}})t_1$, as the second ‘bang’ needs to drive the vertical velocity resulting from the first bang to zero.

We use a binary search to find the capture positions with this control law. In Fig. 3, simulation results are shown in perspective with the height constrained bounds. Note that when the bang-bang control inputs are larger, both trajectory and capture position come closer to the height constrained bounds.

D. Comparison

We make a high-level comparison with the LIP, the height constrained bounds and the force constrained capture positions. We use a dimensional analysis as in [1], [3] and [2]. The following parameters are used for dimensionless position and height:

$$x' = \frac{x}{z_0}, \quad z' = \frac{z}{z_0}, \quad (23)$$

and horizontal velocity:

$$\dot{x}' = \frac{1}{\sqrt{g z_0}} \dot{x}. \quad (24)$$

In this comparison, we take $\ddot{z}_c = |\ddot{z}_{c,1}| = |\ddot{z}_{c,2}|$ for the vertical force constraint.

¹The authors of [11] give analytic solutions using vertical acceleration, but consider a constant height in the model. For comparison later in this paper, we do not consider this constant height assumption.

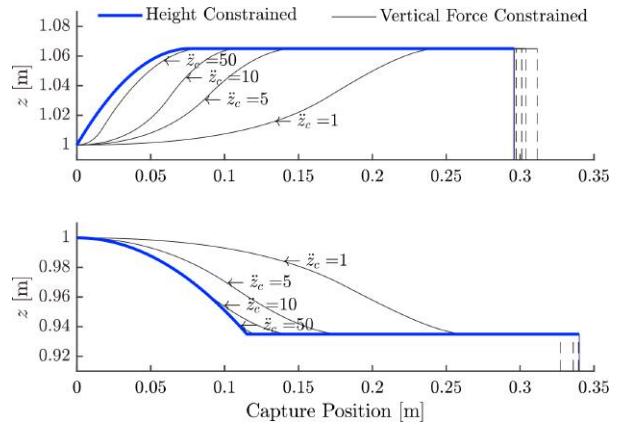


Fig. 3. Simulation results for the vertical force constrained capture positions for $\dot{x}_0 = 1$ [m/s], $\dot{z}_0 = 0$ [m/s] and $\delta z_{max} = \delta z_{min} = 0.065$ [m]. The constant acceleration $\ddot{z}_c = |\ddot{z}_{c,1}| = |\ddot{z}_{c,2}|$ if $\ddot{z}_c \leq g$ and otherwise the constant with negative sign is set to $-g$; units are in [m/s²]. The dashed vertical lines mark the capture positions. Closer to the height constrained bound means a higher value of \ddot{z}_c .

For comparison, we make a rough estimate of realistic values of vertical forces that are achievable on both human and robot.

First, we would like to see what would be achievable for a human being. A human jumping vertically with maximum effort generates approximately $2mg$ ground reaction force [14]. If we assume this value can also be used in recovery, we can take $\ddot{z}_c = g = 9.8$ [m/s²] for a human. Second, we want to see what is possible on the robot. We found on hardware experiments on NASA’s Valkyrie in Section IV-D that $\ddot{z}_c = 2.4$ [m/s²] was a well working value. Larger accelerations would result in the robot to shake and did not improve recovery. In Fig. 4, the height constrained bounds are shown, together with our approximations of what is realistic for vertical acceleration constraints on a human and on the robot. Note how the capture positions relate differently under a minimum height constraint than under a maximum height constraint. Also note how the capture position linking to our approximation for a robot, seems to approach a minimum and maximum value quite soon after changing height.

IV. PUSH RECOVERY ON NASA’S VALKYRIE

In this section we apply a simple controller that uses vertical motion for balance on Valkyrie while standing. The motivation in control design is, instead of using a model predictive controller, to develop a controller that applies the maximum acceleration possible in a worst-case scenario to avoid falling. We compare with CoP control with constant height.

A. Control Law

Our default control law is based on instantaneous capture point (ICP) [2] control:

$$x_{cop,d} = x_{cop,r} + k_\xi \xi_{x,e}, \quad (25)$$

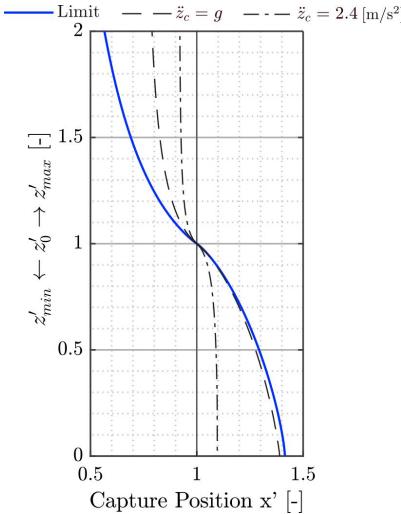


Fig. 4. Plot of reachable dimensionless capture positions for $\dot{x}'_0 = 1$.

where $x_{cop,d}$ is the desired CoP, $x_{cop,r}$ the reference CoP, k_ξ the ICP control gain and $\xi_{x,e}$ is the ICP error between the reference ICP and the current ICP. For this particular test case, we assume a constant $x_{cop,r}$, in the center of the support polygon. Also, $x_{cop,d}$ is constrained to be inside the polygon, such that we can make the assumption that no angular momentum is used in recovery.

The robot is controlled with a momentum-based control framework [8]. Our framework makes use of *centroidal momentum* [15], the angular and linear momentum about the CoM of the robot. A desired centroidal momentum rate, along with motion objectives, is sent to a quadratic program, which optimizes over desired joint accelerations and desired ground reaction forces. Desired joint torques are finally computed using an inverse-dynamics algorithm. We typically only select the linear part of the desired momentum rate for control, allowing the controller to use angular momentum rate as needed. The desired horizontal linear momentum rate is computed as:

$$\dot{\mathbf{i}}_{d,x} = \frac{x - x_{cop,d}}{z} (mg + \dot{\mathbf{i}}_{d,z}), \quad (26)$$

where $\dot{\mathbf{i}}_d \in \mathbb{R}^3$ is the desired linear momentum rate of change. Note that with little vertical motion, $\dot{\mathbf{i}}_{d,z}$ is small.

Normally while standing, the height is controlled to a default constant reference height. In this experiment, we use a similar control law for vertical acceleration as the bang-bang controller in Section III-C. The following parameters are used for the controller in addition to the already discussed constraints:

- \ddot{z}_{max} : maximum allowed vertical CoM jerk;
- $\alpha_{\ddot{z}_c}$: parameter to scale down expected \ddot{z}_c for the second ‘bang’, due to jerk limits.

The control sequence we use for the bang-bang controller reads as follows. The controller is activated when $x_{cop,d}$ touches the polygon edge, an event that determines the worst-case scenario. The controller turns off if $\xi_{x,e}$ is at a small

value, a measure for stability. For the first ‘bang’: the desired acceleration $\ddot{z}_d = \ddot{z}_c$. The transition from the first ‘bang’ to the second is if:

$$z + sign(\dot{z}) \frac{1}{2} \frac{\dot{z}^2}{\alpha_{\ddot{z}_c} \ddot{z}_c} > z_{max}, \quad (27)$$

in the case of approaching a maximum height. This results in $\ddot{z}_d = -\ddot{z}_c$ until $\dot{z} < 0$, after which the height is controlled to z_{max} until the controller turns off:

$$\ddot{z}_d = k_p(z_r - z) - k_d \dot{z}, \quad (28)$$

where $[k_p, k_d] = [50.0, 14.0]$ are the PD-control gains and reference height $z_r = z_{max}$. If the controller is turned off, the height is controlled to the default height and $z_r = z_0$. Finally, the rate of \ddot{z}_d is limited with the maximum allowed jerk and $\dot{\mathbf{i}}_{d,z} = m\ddot{z}_d$.

B. Experimental Setup

We test push recovery on Valkyrie ($m = 127.3$ [kg]) while the robot is standing, by applying a push from the back at chest height. Note that with this setup the resulting motion is always upward, as $x_{cop,d}$ can be placed on the other side of the CoM compared to the direction of $\dot{\mathbf{i}}_{d,x}$. The following parameter values are chosen to work with:

- $z_0 = 1.0$ [m], our default reference CoM height while the robot is standing;
- $z_{max} = 1.065$ [m], the maximum CoM height while standing, such that the legs are not in singular configuration and the feet are still in contact with the ground;
- $\ddot{z}_{max} = 80.0$ [m/s³];
- $\ddot{z}_c = 2.4$ [m/s²], a value that we found to work ‘well’ on hardware. E.g., higher values would result in the robot to shake.

Additionally, whole-body controller parameters relevant to the test are given in Table I. The term basis vector multiplier in the table refers to the optimization variable to calculate the ground reaction forces. This variable multiplies the four basis vectors of the friction cone for each ground contact point. For the angular motion objectives, the desired is generated with PD-control about a constant reference with $[k_p, k_d] = [100.0, 16.0]$. The quadratic program uses an active-set solver [16].

TABLE I
RELEVANT WHOLE-BODY CONTROL PARAMETERS

Task group	Task	Weight
Momentum rate linear	X	$5 \cdot 10^{-2}$
Momentum rate linear	Z	$1 \cdot 10^{-2}$
Motion angular	Chest Y	$1.5 \cdot 10^1$
Motion angular	Pelvis Y	$5 \cdot 10^0$
Motion angular	Support foot Y	$5 \cdot 10^0$
Regularization	Basis vector multiplier	$1 \cdot 10^{-5}$
Regularization	Basis vector multiplier rate	$5 \cdot 10^{-8}$
Regularization	Joint acceleration	$5 \cdot 10^{-3}$
Regularization	Joint jerk	$1.6 \cdot 10^{-6}$

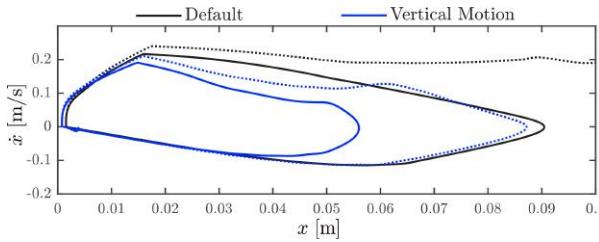


Fig. 5. Phase plot of a push of 34.5 [Ns] (solid) and a push of 37.6 [Ns] (dotted).

C. Simulation Results

For simulation tests, we used a value of $\alpha_{\ddot{z}_c} = 0.4$ for the influence of jerk limitations. We used a push duration of 0.15 [s], as we were able to apply approximately the same push duration on hardware. We compare our default control setup with the controller that uses vertical motion.

1) *Analysis:* The maximum recoverable push for the default control setup is 34.5 [Ns] for the given push duration. The vertical motion controller still recovered after a push of 37.6 [Ns]. In Fig. 5, a phase plot is shown for the two push magnitudes for both control setups. The default setup loses stability after the larger push. With the smaller push, the vertical motion controller encircles a considerably smaller area than the default control setup.

We analyzed the differences in resulting joint torques and noticed that the difference in ankle torque is the largest amount. Furthermore, we found it interesting to compare the maximum rotation error of the pelvis and torso. Angular momentum strategies commonly result in rotation of the upper body. We want to include this rotation in our analysis, as not rotating the body can be one of the advantages of vertical motion compared to angular momentum strategies.

In Fig. 6, centroidal momentum rate, CoM height and ankle torque plots over time are shown. The achieved vertical linear momentum rates have a little overshoot for both controllers. This may also be a reason for the overshoot in height in the fourth graph in the figure. Conversely, the achieved horizontal linear momentum rate is lower than the desired for both controllers. In the time frame 0.10 – 0.25 [s], the vertical motion controller achieves almost double the horizontal momentum rate compared to the default controller. The differences in achieved angular momentum rate are relatively small. We measured a maximum rotation error of $[-0.052, -0.072]$ radians for pelvis and torso respectively for the default setup and $[-0.051, -0.069]$ radians for the vertical motion controller. The resulting rotation errors in the upper body are a little less for the vertical motion controller. The ankle torque has a higher peak with the vertical motion controller, but returns to steady state earlier than the default setup.

2) *Comparison with Capture Regions:* The average recoverable push is about 9% higher for the vertical motion controller compared to the default control setup. Comparing this with the capture regions: the force constrained capture position for the same \ddot{z}_c and z_{max} is only about 4% closer

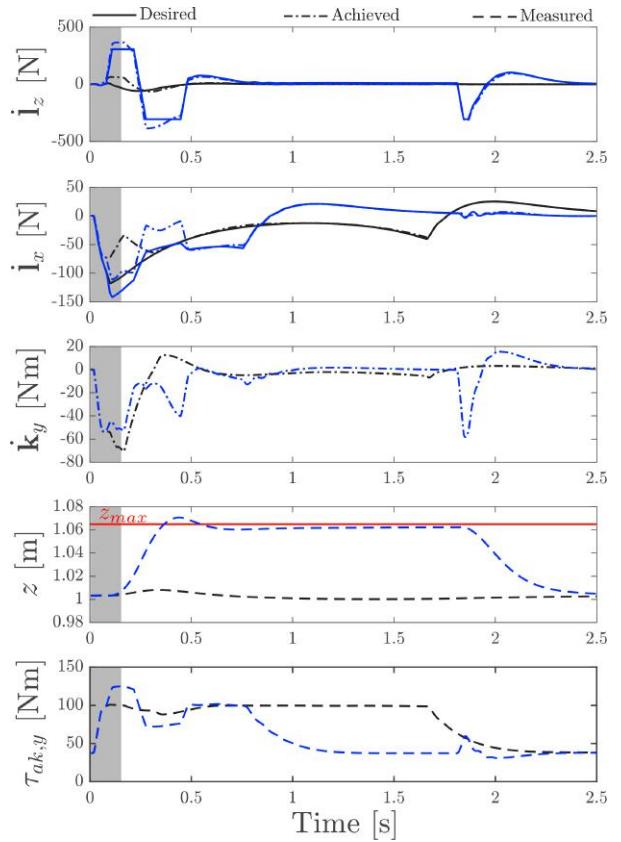


Fig. 6. Comparison of push recovery between the default setup (black) versus the vertical motion controller (blue) for a push of 34.5 [Ns]. The gray area is where the push is applied. ‘Achieved’ is the value after the quadratic program found a solution. \dot{k}_y is the angular momentum rate.

than the LIP capture point, as can be seen in Fig. 4.

From the results obtained, we assume that a difference in angular momentum between the two Valkyrie tests is not a reason for this difference in capturability between model and robot. Also, we noticed that joint angle limits and joint acceleration limits were not violated during the tests. However, the difference in the achieved horizontal momentum rate is large. Note that this can be a result of the momentum-based control framework. Generation of horizontal linear momentum rate may conflict with other objectives, such as keeping the upper body straight, and maintaining a certain height. We also tried commanding the $\dot{i}_{d,x}$ of the vertical motion controller, while using $\dot{i}_{d,z}$ of the default controller, to see if the difference in recovery is just a result of additional desired horizontal linear momentum rate. However, the maximum recoverable push with this setup was the same as with the default setup.

The difference between the increase of the capture region of the model and the increase of recovery of the robot shows that a model-based expectation can differ a lot from results observed on the robot.

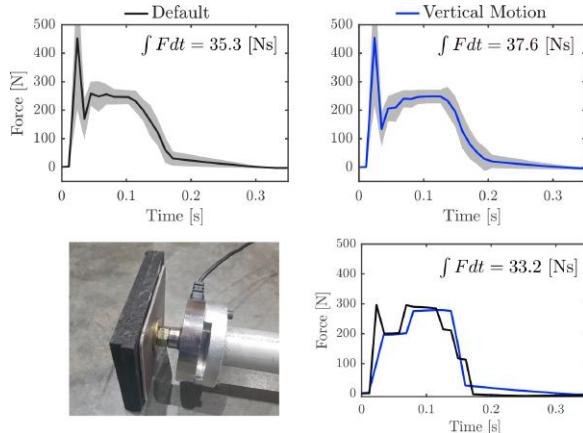


Fig. 7. Average push force profiles of 12 pushes where the CoM went closer than 5.0 [mm] from the polygon edge, the gray area is the standard deviation above and below the graph (top). Load sensor on stick with rubber surface (bottom left). Two pushes with approximately the same integrated force (bottom right).

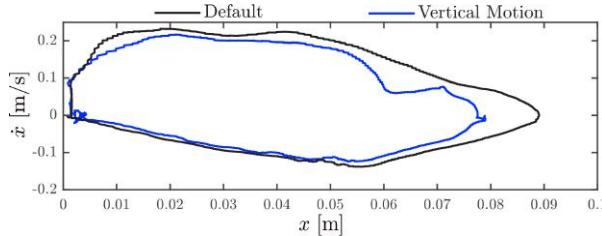


Fig. 8. Phase plot of push recovery on hardware. This is a pick of our data, where both pushes were of magnitude 33.2 [Ns].

D. Hardware Results

We tested the same two control setups on hardware as in simulation. We used a value of $\alpha_{\dot{z}_c} = 0.8$, which appeared to be a better ‘estimate’ for reaching the maximum height for the vertical motion controller. We compared two times a dozen test samples that the robot ‘just’ recovered from the applied push, which we define as the CoM coming closer than 5.0 [mm] from the polygon edge. We measured the push force with an iLoad Pro Digital load sensor at its maximum record frequency of 100 [Hz], see Fig. 7 (bottom left). In Fig. 7 (top), the average force profiles with standard deviation for both control setups are made visible, as well as an image of the load cell. The default setup still recovered with an average push of 35.3 [Ns] and the vertical motion controller with 37.6 [Ns], showing a slight robustness increase. The values for the integrated push force are very similar to the simulation results. However, the measured force profiles on hardware are different from the profile of the constant force applied in simulation.

We take an example case for comparison where the integrated push force on both setups was 33.2 [Ns]. In Fig. 7 (bottom right), the profiles for these two pushes are graphed. Note that this is a rough approximation of a similar disturbance, as other aspects like the force profile, record frequency and measurement noise of the load sensor also

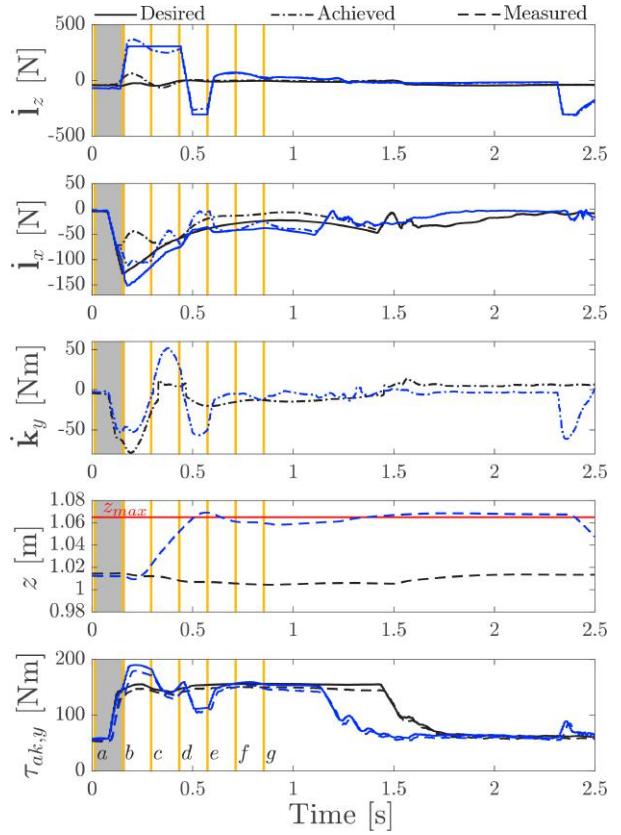


Fig. 9. Time plot of push recovery on hardware. This is a pick of our data, where both pushes were of magnitude 33.2 [Ns]. $\tau_{ank,y}$ is the average of the left and right ankle pitch torque. The letters next to the yellow lines match with the columns in Fig. 10.

play a role in differences observed on the robot. In Fig 8, a phase plot is shown for this push on both setups. A slight increase in robustness for the applied push can be observed.

In Fig. 9, the same variables over time are shown as in the previous section. We found the differences in the length of each ‘bang’ interesting, compared to simulation. This likely also is a reason why a higher value of $\alpha_{\dot{z}_c}$ was possible on hardware. Also notice the difference in resulting angular momentum rate. For this push on hardware, we measured $[-0.055, -0.074]$ radians of maximum pelvis and torso rotation error on the default setup and $[-0.045, -0.058]$ radians on the vertical motion controller, which is again less resulting body rotation with the vertical motion controller. The averaged ankle pitch torque over left and right has a higher peak for the vertical motion controller, as expected. Also, the ankle torque of the vertical motion controller returns to steady state earlier.

In Fig. 10, a time-lapse image is shown of Valkyrie recovering from a push using the vertical motion controller, and using the default controller setup. The letters below the columns correspond with the numbers next to the yellow lines in Fig. 9. Note how there is no contact of the push head, when the yellow lines are outside the gray area.

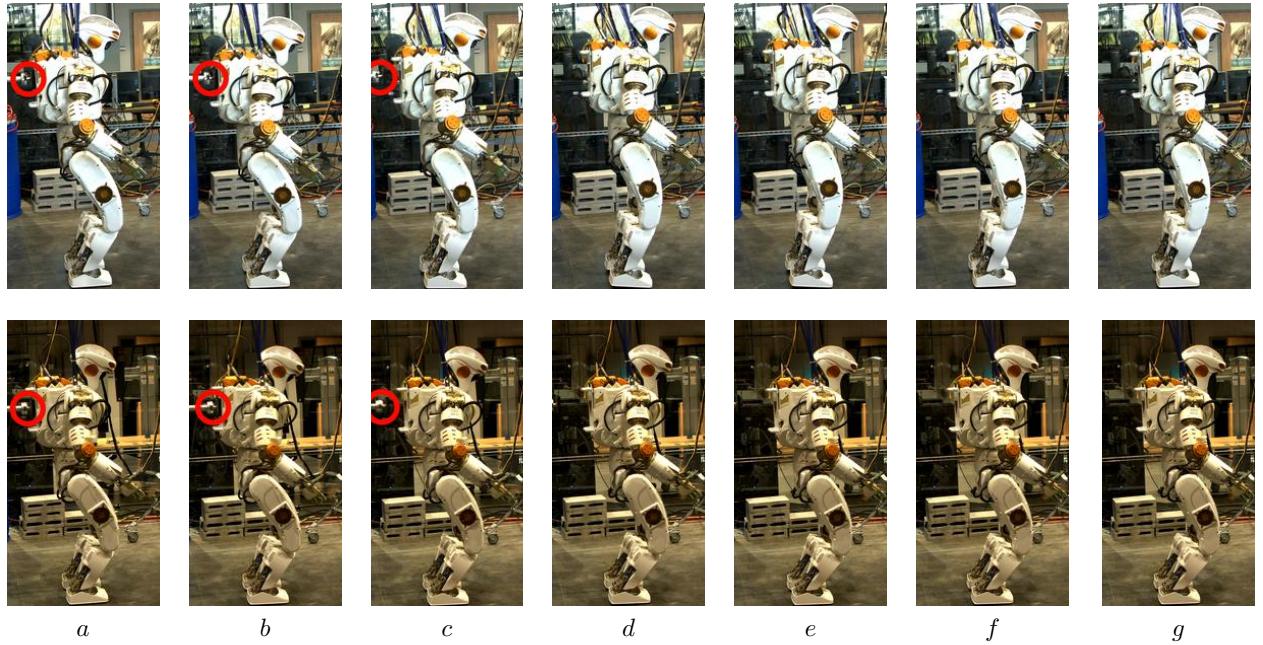


Fig. 10. Time-lapse of Valkyrie recovering from a push using vertical motion (top row) and using the default controller setup (bottom row). The letters below the columns match with the letters next to the yellow lines in Fig. 9. The push rod tip is encircled in red.

V. CONCLUSION

To increase the reliability, it is important that humanoid robots improve their balancing behavior. In this paper, we studied the effectiveness of vertical CoM motion in balance control in 2D. We derived capture regions for varying CoM height on a commonly used, simple model. We showed on Valkyrie in simulation and on hardware that balance can be improved using vertical CoM motions. Using this model-to-robot analysis, we showed differences that can be observed when going from a model-based expectation to the real result.

For the future, we are interested in 3D and multi-step strategies for the robot to balance using CoM height variation. Also, we are interested in the coupled effects of, e.g., combining vertical CoM motion with angular momentum strategies. We believe in building a portfolio of balancing strategies, as in [17], which can be used by the robot depending on the situation.

REFERENCES

- [1] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 200–207.
- [2] T. Koolen, T. De Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part I: Theory and application to three simple gait models," *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.
- [3] B. Stephens, "Humanoid push recovery," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2007, pp. 589–595.
- [4] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot-1 st report: Walking gait pattern generation," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 1084–1091.
- [5] L. Lanari, S. Hutchinson, and L. Marchionni, "Boundedness issues in planning of locomotion trajectories for biped robots," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 951–958.
- [6] S. Kajita, F. Kaneko, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Resolved momentum control: Humanoid motion planning based on the linear and angular momentum," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 2, 2003, pp. 1644–1650.
- [7] S.-H. Lee and A. Goswami, "A momentum-based balance controller for humanoid robots on non-level and non-stationary ground," *Autonomous Robots*, vol. 33, no. 4, pp. 399–414, 2012.
- [8] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.
- [9] T. Koolen, M. Posa, and R. Tedrake, "Balance control using center of mass height variation: limitations imposed by unilateral contact," in *IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 8–15.
- [10] J. E. Pratt and S. V. Drakunov, "Derivation and application of a conserved orbital energy for the inverted pendulum bipedal walking model," in *IEEE Int. Conf. on Robot. Autom.*, 2007, pp. 4653–4660.
- [11] W. Gao, Z. Jia, and C. Fu, "Increase the feasible step region of biped robots through active vertical flexion and extension motions," *Robotica*, vol. 35, no. 7, pp. 1541–1561, 2017.
- [12] S. Caron and B. Mallein, "Balance control using both zmp and com height variations: A convex boundedness approach," in *IEEE Int. Conf. Robot. Autom.*, 2018, pp. 1779–1784.
- [13] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdeyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgewater, et al., "Valkyrie: Nasa's first bipedal humanoid robot," *Journal of Field Robotics*, vol. 32, no. 3, pp. 397–419, 2015.
- [14] N. P. Linthorne, "Analysis of standing vertical jumps using a force platform," *American Journal of Physics*, vol. 69, no. 11, pp. 1198–1204, 2001.
- [15] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, vol. 35, no. 2-3, pp. 161–176, 2013.
- [16] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *IEEE Int. Conf. Robot. Autom.*, 2014, pp. 2589–2594.
- [17] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt, "Walking stabilization using step timing and location adjustment on the humanoid robot, atlas," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 667–673.

Bibliography

- [1] S. Behnke, “Humanoid robots-from fiction to reality?,” *Kuenstliche Intelligenz (KI)*, vol. 22, no. 4, pp. 5–9, 2008.
- [2] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, “The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, pp. 239–246, 2001.
- [3] S. Kajita, T. Yamaura, and A. Kobayashi, “Dynamic walking control of a biped robot along a potential energy conserving orbit,” *Transactions on Robotics and Automation*, vol. 8, no. 4, pp. 431–438, 1992.
- [4] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, “Capture point: A step toward humanoid push recovery,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 200–207, 2006.
- [5] T. Koolen, T. De Boer, J. Rebula, A. Goswami, and J. Pratt, “Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models,” *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.
- [6] A. L. Hof, “The ‘extrapolated center of mass’ concept suggests a simple control of balance in walking,” *Human Movement Science*, vol. 27, no. 1, pp. 112–125, 2008.
- [7] T. Takenaka, T. Matsumoto, and T. Yoshiike, “Real time motion generation and control for biped robot-1 st report: Walking gait pattern generation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1084–1091, 2009.
- [8] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, “Design of a momentum-based control framework and application to the humanoid robot atlas,” *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.

- [9] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, “Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot,” *Autonomous Robots*, vol. 40, no. 3, pp. 429–455, 2016.
- [10] J. E. Pratt and S. V. Drakunov, “Derivation and application of a conserved orbital energy for the inverted pendulum bipedal walking model,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4653–4660, 2007.
- [11] J. Englsberger, C. Ott, and A. Albu-Schäffer, “Three-dimensional bipedal walking control using divergent component of motion,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2600–2607, 2013.
- [12] M. A. Hopkins, D. W. Hong, and A. Leonessa, “Humanoid locomotion on uneven terrain using the time-varying divergent component of motion,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 266–272, 2014.
- [13] T. Koolen, M. Posa, and R. Tedrake, “Balance control using center of mass height variation: limitations imposed by unilateral contact,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 8–15, 2016.
- [14] S. Caron and B. Mallein, “Balance control using both zmp and com height variations: A convex boundedness approach,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1779–1784, 2018.
- [15] S. M. Hasson, *Clinical exercise physiology*. Mosby Inc, 1994.
- [16] A. D. Kuo, J. M. Donelan, and A. Ruina, “Energetic consequences of walking like an inverted pendulum: step-to-step transitions,” *Exercise and Sport Sciences Reviews*, vol. 33, no. 2, pp. 88–97, 2005.
- [17] P. Sardain and G. Bessonnet, “Forces acting on a biped robot. center of pressure-zero moment point,” *Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 34, no. 5, pp. 630–637, 2004.
- [18] M. Vukobratovic and D. Juricic, “Contribution to the synthesis of biped gait,” *Transactions on Biomedical Engineering*, no. 1, pp. 1–6, 1969.
- [19] M. Vukobratović and B. Borovac, “Zero-moment point—thirty five years of its life,” *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 157–173, 2004.
- [20] M. B. Popovic, A. Goswami, and H. Herr, “Ground reference points in legged locomotion: Definitions, biological trajectories and control implications,” *The International Journal of Robotics Research*, vol. 24, no. 12, pp. 1013–1032, 2005.
- [21] B. Stephens, “Humanoid push recovery,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 589–595, 2007.
- [22] Y. Liu, P. M. Wensing, D. E. Orin, and Y. F. Zheng, “Trajectory generation for dynamic walking in a humanoid over uneven terrain using a 3d-actuated dual-slip model,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 374–380, 2015.

- [23] W. J. Schwind, *Spring loaded inverted pendulum running: A plant model*. PhD thesis, University of Michigan, 1998.
- [24] T. Seyde, A. Shrivastava, J. Englsberger, S. Bertrand, J. Pratt, and R. J. Griffin, “Inclusion of angular momentum during planning for capture point based walking,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1791–1798, 2018.
- [25] J. Englsberger and C. Ott, “Integration of vertical com motion and angular momentum in an extended capture point tracking controller for bipedal walking,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 183–189, 2012.
- [26] “IHMC’s open-source software.” <https://github.com/ihmcrobotics>. Accessed: 19/02/2019.
- [27] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdelyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgwater, *et al.*, “Valkyrie: Nasa’s first bipedal humanoid robot,” *Journal of Field Robotics*, vol. 32, no. 3, pp. 397–419, 2015.
- [28] “Image of the Atlas humanoid robot.” <http://robots.ihmc.us/>. Accessed: 27/01/2019.
- [29] “Image of the Valkyrie humanoid robot.” <https://spectrum.ieee.org/automaton/robotics/humanoids/video-friday-ihmc-valkyrie-harvard-arthropods-flying-wheeled-robot>. Accessed: 27/01/2019.
- [30] J. Englsberger, T. Koolen, S. Bertrand, J. Pratt, C. Ott, and A. Albu-Schäffer, “Trajectory generation for continuous leg forces during double support and heel-to-toe shift based on divergent component of motion,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4022–4029, 2014.
- [31] D. E. Orin, A. Goswami, and S.-H. Lee, “Centroidal dynamics of a humanoid robot,” *Autonomous Robots*, vol. 35, no. 2-3, pp. 161–176, 2013.
- [32] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Resolved momentum control: Humanoid motion planning based on the linear and angular momentum,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 2, pp. 1644–1650, 2003.
- [33] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt, “Straight-leg walking through underconstrained whole-body control,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–5, 2018.
- [34] J. M. Wang, S. R. Hamner, S. L. Delp, and V. Koltun, “Optimizing locomotion controllers using biologically-based actuators and objectives,” *ACM transactions on graphics*, vol. 31, no. 4, 2012.
- [35] C. Brasseur, A. Sherikov, C. Collette, D. Dimitrov, and P.-B. Wieber, “A robust linear mpc approach to online generation of 3d biped walking motion,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 595–601, 2015.

- [36] S. Kajita, M. Benallegue, R. Cisneros, T. Sakaguchi, S. Nakaoka, M. Morisawa, K. Kaneko, and F. Kanehiro, “Biped walking pattern generation based on spatially quantized dynamics,” in *IEEE-RAS International Conference on Humanoid Robotics (Humanoids)*, pp. 599–605, 2017.
- [37] W. Gao, Z. Jia, and C. Fu, “Increase the feasible step region of biped robots through active vertical flexion and extension motions,” *Robotica*, vol. 35, no. 7, pp. 1541–1561, 2017.
- [38] L. Lanari, S. Hutchinson, and L. Marchionni, “Boundedness issues in planning of locomotion trajectories for biped robots,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 951–958, 2014.
- [39] T. A. McMahon and J. T. Bonner, *On size and life*. Scientific American Library, 1983.
- [40] N. P. Linthorne, “Analysis of standing vertical jumps using a force platform,” *American Journal of Physics*, vol. 69, no. 11, pp. 1198–1204, 2001.
- [41] S. A. Gard, S. C. Miff, and A. D. Kuo, “Comparison of kinematic and kinetic methods for computing the vertical motion of the body center of mass during walking,” *Human Movement Science*, vol. 22, no. 6, pp. 597–610, 2004.
- [42] R. J. Griffin, *Natural, Efficient Walking for Compliant Humanoid Robots*. PhD thesis, Virginia Tech, 2017.
- [43] “iLoad Pro Digital Load Cell 500 lb specifications.” <https://www.loadstarsensors.com/iload-pro-industrial-grade-digital-load-cell.html>. Accessed: 27/01/2019.

List of Acronyms

IHMC	the Institute for Human and Machine Cognition
ICP	instantaneous capture point
LIPCP	linear inverted pendulum (LIP) capture point
DCM	divergent component of motion
ZMP	zero moment point
CoP	center of pressure
CoM	center of mass
CMP	centroidal moment pivot
LIP	linear inverted pendulum
VHIP	variable height inverted pendulum
2D	two-dimensional space
3D	three-dimensional space
MPC	model predictive control
SCS	Simulation Construction Set
SLIP	spring-loaded inverted pendulum
GRF	ground reaction force
QP	quadratic program

