

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 80NSSC18M0071.

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

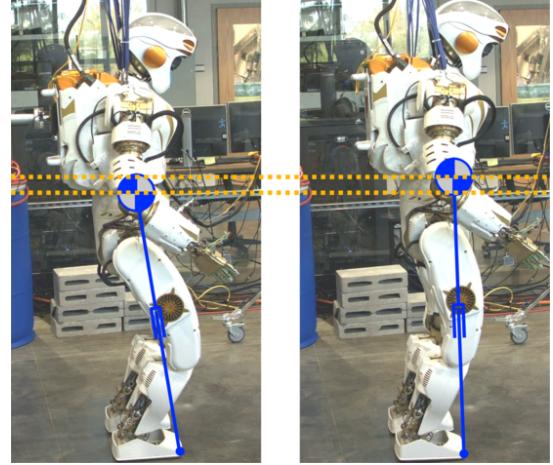


Fig. 1. Visualization of the idea in this paper.

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:

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

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

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

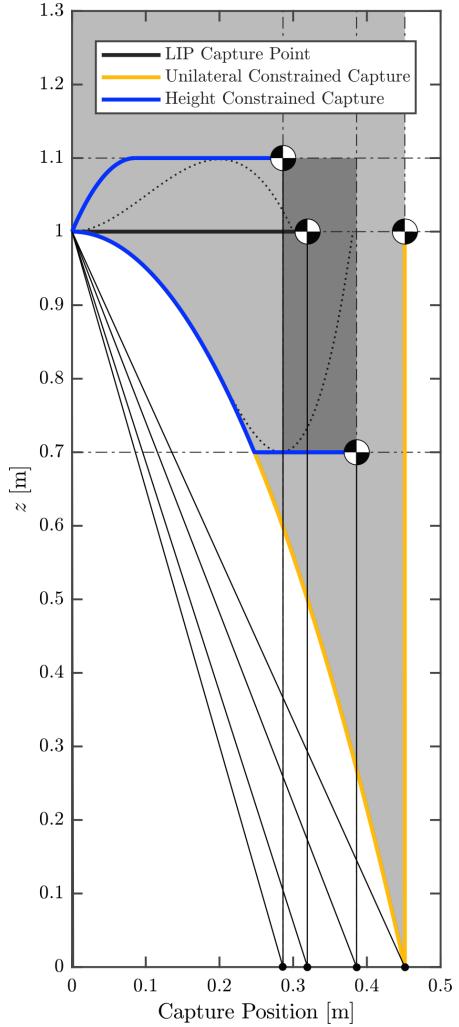


Fig. 2. Visualization of the analytic capture regions for $\dot{x}_0 = 1$ [m/s] and $\dot{z}_0 = 0$ [m/s]. The light gray area shows the unilateral contact constrained capture region (4). The dark gray area shows the height constrained capture region (Lemma 1) for $0.7 < z < 1.1$ [m]. The dotted plots 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

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

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.

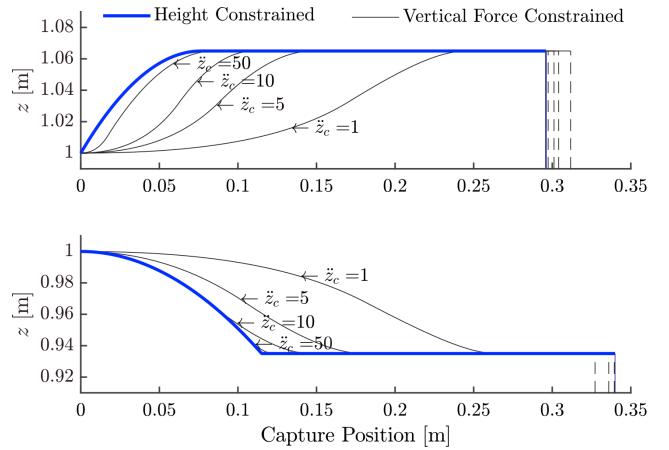


Fig. 3. Simulation results for the vertical force constrained capture positions for $\dot{x}_0 = 1$ [m/s], $\dot{z}_0 = 0$ [m/s] and $\delta z_{max} = \delta z_{min} = 0.065$ [m]. The constant acceleration $\ddot{z}_c = |\ddot{z}_{c,1}| = |\ddot{z}_{c,2}|$ if $\ddot{z}_c \leq g$ and otherwise the constant 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:

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

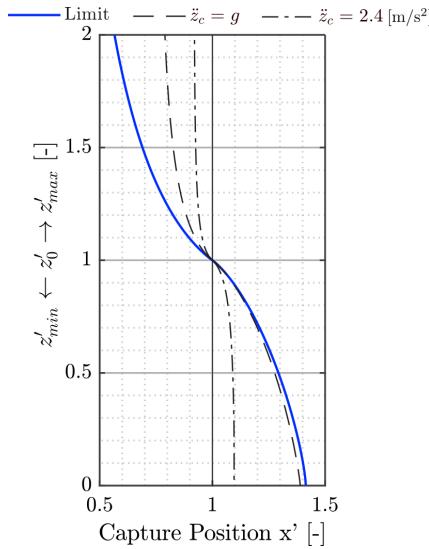


Fig. 4. Plot of reachable dimensionless capture positions for $\dot{x}'_0 = 1$.

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 $\dot{\mathbf{i}}_{d,z} = m\ddot{\mathbf{i}}_d$.

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;
- $\dot{z}_{max} = 80.0$ [m/s²];
- $\ddot{z}_c = 2.4$ [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	$1.6 \cdot 10^{-6}$

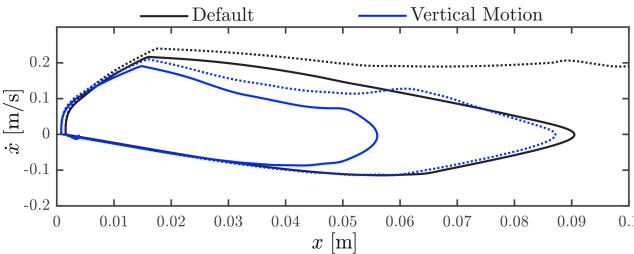


Fig. 5. Phase plot of a push of 34.5 [Ns] (solid) and a push of 37.6 [Ns] (dotted).

C. Simulation Results

For simulation tests, we used a value of $\alpha_{\dot{z}_c} = 0.4$ 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 \dot{z}_c and z_{max} is only about 4% closer

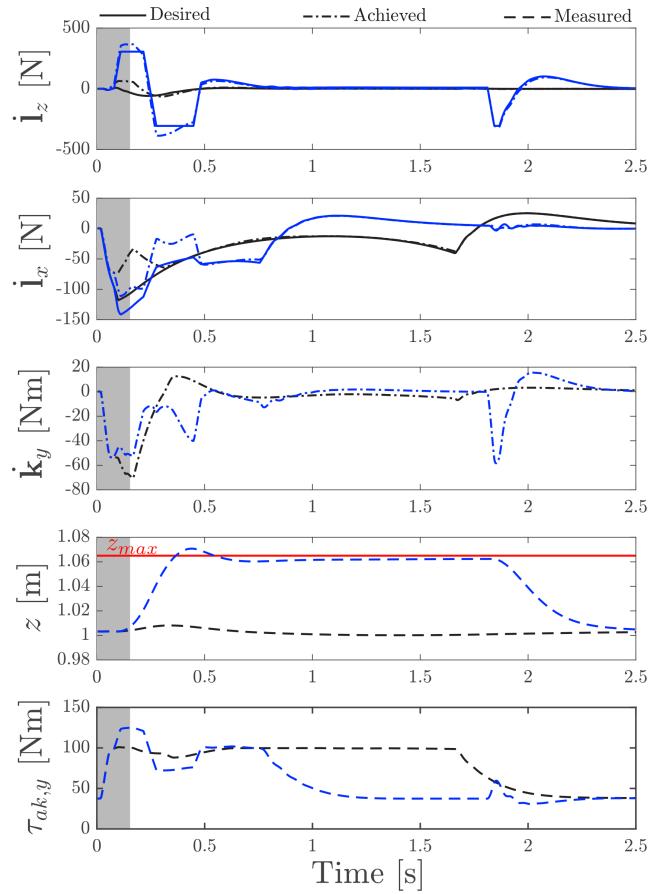


Fig. 6. Comparison of push recovery between the default setup (black) versus the vertical motion controller (blue) for a push of 34.5 [Ns]. The gray area is where the push is applied. ‘Achieved’ is the value after the quadratic program found a solution. k_y is the angular momentum rate.

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.

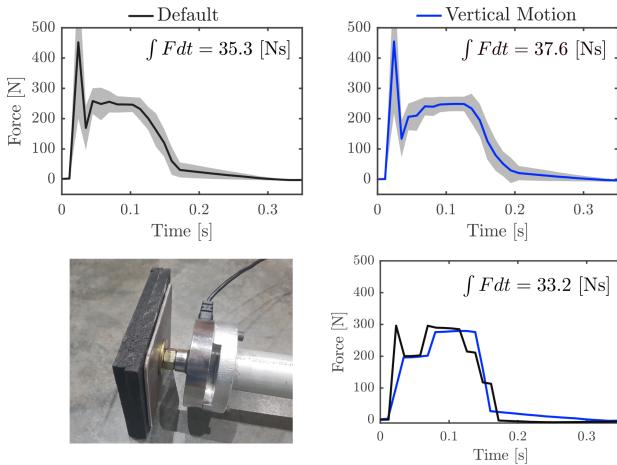


Fig. 7. Average push force profiles of 12 pushes where the CoM went closer than 5.0 [mm] from the polygon edge, the gray area is the standard deviation above and below the graph (top). Load sensor on stick with rubber surface (bottom left). Two pushes with approximately the same integrated force (bottom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

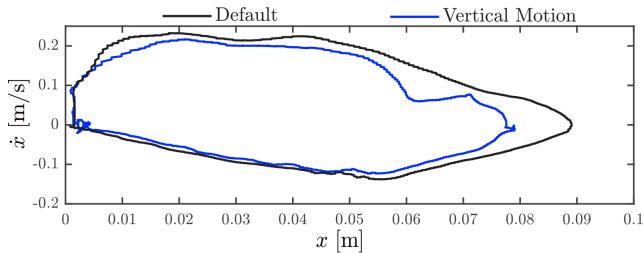


Fig. 8. Phase plot of push recovery on hardware. This is a pick of our data, where both pushes were of magnitude 33.2 [Ns].

D. Hardware Results

We tested the same two control setups on hardware as in simulation. We used a value of $\alpha_{\hat{z}_c} = 0.8$, 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

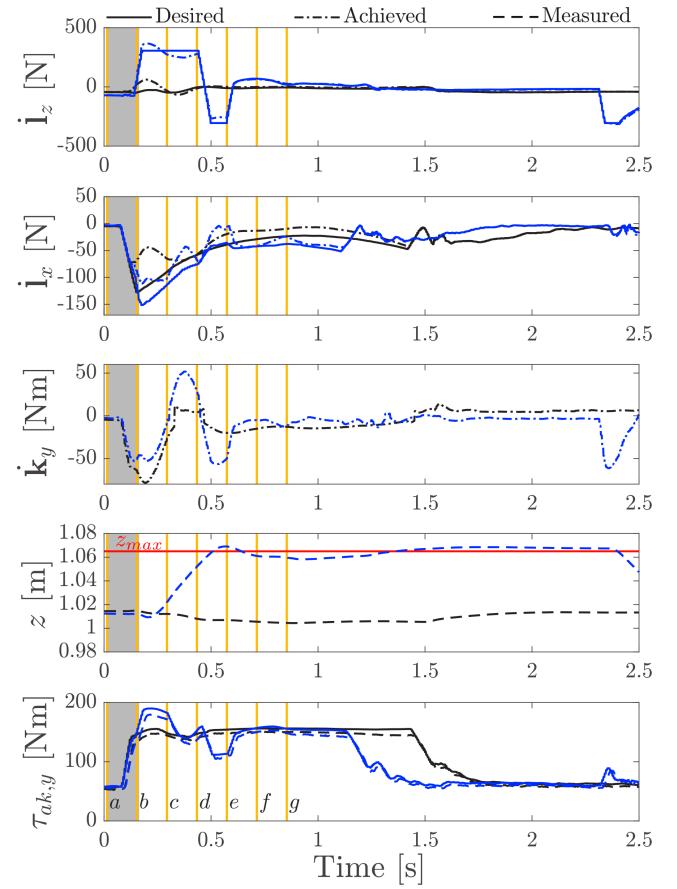


Fig. 9. Time plot of push recovery on hardware. This is a pick of our data, where both pushes were of magnitude 33.2 [Ns]. $\tau_{ak,y}$ is the average of the left and right ankle pitch torque. The letters next to the yellow lines match with the columns in Fig. 10.

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 $\alpha_{\hat{z}_c}$ 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.

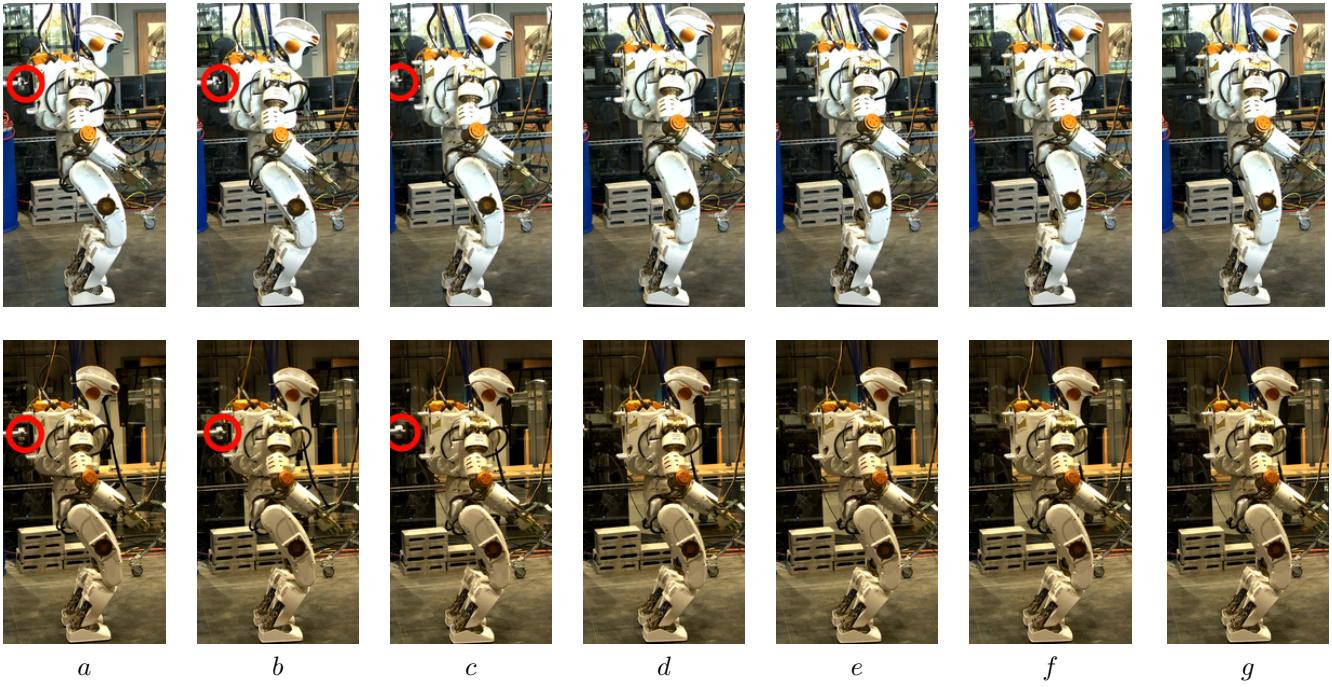


Fig. 10. Time-lapse of Valkyrie recovering from a push using vertical motion (top row) and using the default controller setup (bottom row). The letters below the columns match with the letters next to the yellow lines in Fig. 9. The push rod tip is encircled in red.

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 1: 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. 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 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.