Force control for a humanoid robot using momentum and instantaneous capture point dynamics

Twan Koolen
Institute for Human and Machine Cognition
Pensacola, Florida, USA
Email: tkoolen@ihmc.us

***Abstract*—This paper presents a novel controller that exploits the properties of whole-body momentum to control a high-degree-of-freedom force-controlled humanoid robot. The controller computes desired centers of pressure and ground reaction forces at each foot, which can be mapped directly to a desired rate of change of whole-body momentum. Balance is achieved by modulating both the locations of the centers of pressure as well as the directions of the ground reaction forces, based on the dynamics of the instantaneous capture point. This results in a combined hip- and ankle balance strategy. Simulation results are presented that show a 26-degree-of-freedom robot climbing a set of stairs that is sufficiently steep that only the toes can be in contact with the steps.**

***Index Terms*—humanoid robots, force control, momentum, capture point**

I. Introduction

Balance control for bipedal robots remains an active field of research, as the abilities of humans have not yet been matched by current robots. A number of recent studies have proposed balance controllers based on the properties of the linear and angular momentum of the robot as a whole [1], [2]. The direct relationships between the rate of change of momentum, joint accelerations, and ground reaction forces exerted on the robot makes this approach very attractive.

Lee and Goswami have recently introduced a momentum-based force control scheme that formed the inspiration for this study [2]. Like their work, we determine a desired rate of change of linear and angular momentum, and subsequently compute compatible joint torques. However, whereas Lee and Goswami used a simple PD control law to determine the desired rate of change of momentum, we have developed a model-based approach using the dynamics of the *instantaneous capture point* (ICP).

Recent studies have shown that the simple form of the dynamics of the ICP can be used to design controllers for a bipedal robot [3], [4], [5], [6]. We showed that for a simple gait model, the instantaneous capture point diverges away from the Centroidal Moment Pivot (CMP) [7]. The CMP is a ground reaction point, similar to the Center of Pressure (CoP), that encodes both the application point and the direction of the total ground reaction force. It too is closely related to the rate of change of momentum. In this paper, we investigate the links between momentum, the instantaneous capture point, and the CMP, and generalize results obtained in [5]. This analysis leads to the design of a controller that allows a simulated 26-degree-of-freedom robot to quickly climb a set of stairs that are so steep that only the toes can be placed on the steps.

The paper is structured as follows. Section II introduces the basic theory and notation used throughout the paper. Section III applies this theory to a bipedal robot and derives equations that govern the rate of change of momentum, as well as the instantaneous capture point dynamics. Section IV describes the proposed control framework. Section V presents the state machine that was built on top of the control framework to obtain a stair climbing behavior. Simulation results are presented in section VI, and a discussion is provided in section VII. Section VIII concludes the paper.

II. Theoretical introduction and notation

This section gives a brief introduction to the theory of homogeneous transformations and twists, which will be used in this paper to model robot kinematics and dynamics. For a more in-depth introduction to this topic, refer to [8], [9], [10]. Our notation is adapted from [9].

The configuration of a rigid body with attached coordinate frame Φ_j with respect to a second coordinate frame Φ_i can be described numerically by a *homogeneous transformation matrix*

$$\mathbf{H}_j^i := \begin{pmatrix} \mathbf{R}_j^i & \mathbf{p}_j^i \\ \mathbf{0}_{1\times 3} & 1 \end{pmatrix} \in \mathbb{R}^{4\times 4}$$

where $\mathbf{R}_j^i \in \mathbb{R}^{3\times 3}$ is the rotation matrix that rotates vectors from Φ_j to Φ_i, and $\mathbf{p}_j^i$ is the position of the origin of Φ_j expressed in Φ_i.

The velocity of Φ_j with respect to Φ_i (both angular and linear) can be expressed in a third frame Φ_k as

$$\hat{\mathbf{T}}_j^{k,i} := \mathbf{H}_i^k \dot{\mathbf{H}}_j^i \mathbf{H}_k^j = \begin{pmatrix} \hat{\boldsymbol{\omega}}_j^{k,i} & \mathbf{v}_j^{k,i} \\ \mathbf{0}_{1\times 3} & 0 \end{pmatrix} \in \mathbb{R}^{4\times 4}$$

where the operator $\hat{}$ maps a vector in $\mathbb{R}^3$ to the associated skew-symmetric matrix in $\mathbb{R}^{3\times 3}$ such that $\hat{x}y = x \times y$ for all $x, y \in \mathbb{R}^3$. The pseudovector $\boldsymbol{\omega}_j^{k,i} \in \mathbb{R}^3$ is the angular velocity of Φ_j with respect to Φ_i, rotated to frame Φ_k, and

$\mathbf{v}_j^{k,i} \in \mathbb{R}^3$ can be interpreted as the 'stream velocity' of points that are instantaneously coincident with the origin of Φ_i but fixed in Φ_j, rotated to frame Φ_k. This description of velocity as a 4×4 matrix is highly redundant, but it can easily be transformed to a non-redundant form,

$$\mathbf{T}_j^{k,i} := \begin{pmatrix} \boldsymbol{\omega}_j^{k,i} \\ \mathbf{v}_j^{k,i} \end{pmatrix} \in \mathbb{R}^6$$

called the *twist* of Φ_j with respect to Φ_i, expressed in Φ_k. A twist can be expressed in a different frame Φ_m using

$$\mathbf{T}_j^{m,i} = \mathrm{Ad}_{\mathbf{H}_k^m} \mathbf{T}_j^{k,i}, \text{ with } \mathrm{Ad}_{\mathbf{H}_k^m} := \begin{pmatrix} \mathbf{R}_k^m & \mathbf{0}_{3\times3} \\ \hat{\mathbf{p}}_k^m \mathbf{R}_k^m & \mathbf{R}_k^m \end{pmatrix}$$

Two other useful properties are inversion, and addition of twists that are expressed in the same frame:

$$\mathbf{T}_i^{k,j} = -\mathbf{T}_j^{k,i} \tag{1}$$

$$\mathbf{T}_m^{k,i} = \mathbf{T}_j^{k,i} + \mathbf{T}_m^{k,j} \tag{2}$$

The relative acceleration between Φ_j and Φ_i can be described using a *spatial acceleration vector* $\dot{\mathbf{T}}_j^{k,i}$, which is simply the derivative of the twist with respect to time.

The inertia of the rigid body is expressed in Φ_j as

$$\mathbf{I}^{j,j} := \begin{pmatrix} \mathbf{J}^{j,j} & m^j \hat{\mathbf{c}}^{j,j} \\ -m^j \hat{\mathbf{c}}^{j,j} & m^j \mathbf{I}_3 \end{pmatrix} \in \mathbb{R}^{6\times6}$$

where $m^j > 0$ is the mass of the rigid body, $\mathbf{J}^{j,j} \in \mathbb{R}^{3\times3}$ is the mass moment of inertia expressed in Φ_j, and $\mathbf{c}^{j,j} \in \mathbb{R}^3$ is the center of mass location expressed in Φ_j.

The momentum of the rigid body, expressed in body frame, is then defined as

$$\mathbf{h}^{j,j} = \begin{pmatrix} \mathbf{k}^{j,j} \\ \mathbf{l}^{j,j} \end{pmatrix} := \mathbf{I}^{j,j} \mathbf{T}_j^{j,\mathrm{in}}$$

where the twist is computed with respect to an inertial reference frame Φ_{in}, and $\mathbf{k}^{j,j}$ and $\mathbf{l}^{j,j}$ are angular and linear momentum respectively. A momentum vector can be expressed in a different frame using

$$\mathbf{h}^{k,j} = \mathrm{Ad}_{\mathbf{H}_k^j}^T \mathbf{h}^{j,j}$$

If momentum of a rigid body is expressed in an inertial frame, its rate of change obeys Euler's laws of motion,

$$\dot{\mathbf{h}}^{\mathrm{in},j} = \mathbf{W}^{\mathrm{in},j} := \begin{pmatrix} \boldsymbol{\tau}^{\mathrm{in},j} \\ \mathbf{f}^{\mathrm{in},j} \end{pmatrix}$$

where $\mathbf{W}^{\mathrm{in},j}$ is the net *wrench* exerted on the body, expressed in Φ_{in}, with torque $\boldsymbol{\tau}^{\mathrm{in},j}$ and force $\mathbf{f}^{\mathrm{in},j}$.

We now consider a mechanism that consists of multiple rigid bodies. The total momentum of the mechanism, expressed in the inertial frame, $\mathbf{h}^{\mathrm{in}}$, is obtained by simply summing all of the individual momenta. Hence,

$$\dot{\mathbf{h}}^{\mathrm{in}} = \begin{pmatrix} \dot{\mathbf{k}}^{\mathrm{in}} \\ \dot{\mathbf{l}}^{\mathrm{in}} \end{pmatrix} := \sum_i \dot{\mathbf{h}}^{\mathrm{in},i} = \sum_i \mathbf{W}^{\mathrm{in},i} = \mathbf{W}_{\mathrm{ext}}^{\mathrm{in}} \tag{3}$$

where $\mathbf{W}_{\mathrm{ext}}^{\mathrm{in}}$ is the net external wrench exerted on the robot. The last equality follows from Newton's third law, which implies that wrenches internal to the robot cancel out in the summation, and only external wrenches remain.

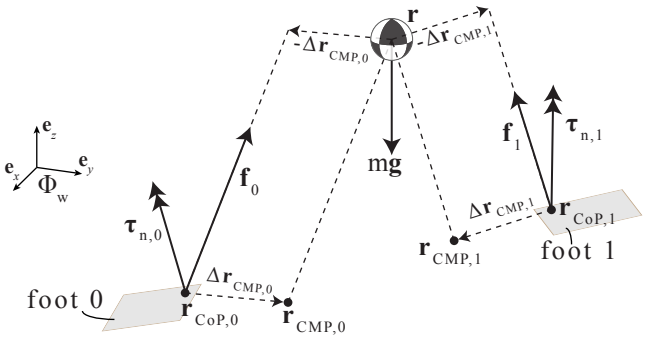

Figure 1. External forces and torques acting on a bipedal robot.

In the next section, we will use this theoretical framework in the specific case of a bipedal robot.

III. Bipedal robot dynamics

This section examines the wrenches exerted on a fairly general bipedal robot (III-A), after which the translational (III-B) and rotational dynamics (III-C) are derived.

It is convenient to express the dynamics in a centroidal frame Φ_c [11], i.e. an inertial frame with the same orientation as a ground-fixed world frame Φ_w, but with its origin instantaneously coincident with the CoM position, $\mathbf{r}$.

A. Wrenches exerted on a bipedal robot

Fig. 1 shows the external wrenches acting on a bipedal robot. We assume that the feet are flat, and that the only external wrenches are the wrench due to gravity, $\mathbf{W}_{\mathrm{g}}^c$, and the ground reaction wrenches $\mathbf{W}_{\mathrm{GR}}^{c,i}$ exerted on foot number $i \in \{0, 1\}$:

$$\mathbf{W}_{\mathrm{ext}}^c = \begin{pmatrix} \boldsymbol{\tau}_{\mathrm{ext}}^c \\ \mathbf{f}_{\mathrm{ext}}^c \end{pmatrix} = \mathbf{W}_{\mathrm{g}}^c + \mathbf{W}_{\mathrm{GR}}^{c,0} + \mathbf{W}_{\mathrm{GR}}^{c,1} \tag{4}$$

The wrench due to gravity is

$$\mathbf{W}_{\mathrm{g}}^c = \begin{pmatrix} \mathbf{0}_{3\times1} \\ m\mathbf{g} \end{pmatrix} \tag{5}$$

where m is the total mass and $\mathbf{g} = \begin{pmatrix} 0 & 0 & -g_z \end{pmatrix}^T$.

Any system of forces exerted on foot number i can be expressed in terms of the foot's CoP $\mathbf{r}_{\mathrm{CoP},i}$ [12], located on the sole, a resultant force $\mathbf{f}_i$ exerted at the CoP, and a torque $\boldsymbol{\tau}_{\mathrm{n},i}$ normal to the sole, as visualized in Fig. 1. The ground reaction wrench on foot i is then expressed as

$$\mathbf{W}_{\mathrm{GR}}^{c,i} = \begin{pmatrix} (\mathbf{r}_{\mathrm{CoP},i} - \mathbf{r}) \times \mathbf{f}_i + \boldsymbol{\tau}_{\mathrm{n},i} \\ \mathbf{f}_i \end{pmatrix} \tag{6}$$

where the ground reaction force can be written as

$$\begin{aligned} \mathbf{f}_i &= k_i \left(\mathbf{r} - \mathbf{r}_{\mathrm{CoP},i} - \Delta\mathbf{r}_{\mathrm{CMP},i} \right) \\ &= k_i \left(\mathbf{r} - \mathbf{r}_{\mathrm{CMP},i} \right) \end{aligned} \tag{7}$$

with some scalar $k_i > 0$ representing the intensity of the force. Here, $\mathbf{r}_{\text{CMP},i}$ is the Centroidal Moment Pivot (CMP) for foot i, and $\Delta\mathbf{r}_{\text{CMP},i}$ is what we call the CMP offset, which has no vertical component; $\Delta z_{\text{CMP}} = 0$. The CMP offset allows the direction of the ground reaction force to be changed without changing the CoP, but with an effect on the rate of change of centroidal angular momentum.

The CMP was defined as a ground reference point for an entire robot in [7]. Here we slightly abuse this concept by defining a CMP for each foot, which is at the same height as the CoP and may hence not be at ground height. The main idea of having the difference between the CoP and the CMP represent a way to change horizontal CoM motion by changing angular momentum remains the same.

B. Translational dynamics

Substituting (5), (6), and (7) into (4) and selecting the last three rows (i.e. the translational part), we obtain

$$\mathbf{f}^c_{\text{ext}} = (k_0 + k_1)\,\mathbf{r} - k_0\mathbf{r}_{\text{CMP},0} - k_1\mathbf{r}_{\text{CMP},1} + m\mathbf{g} \tag{8}$$

Since the whole-body linear momentum is $\dot{\mathbf{l}}^c = m\ddot{\mathbf{r}}$, the last three rows of (3) can be written as

$$m\ddot{\mathbf{r}} = \mathbf{f}^c_{\text{ext}} \tag{9}$$

Substituting (8) into (9) and using the projection matrix $\mathbf{P} = \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{smallmatrix}\right)$ to select the horizontal part, we obtain

$$\mathbf{P}\ddot{\mathbf{r}} = \omega_0^2\mathbf{P}\left(\mathbf{r} - \mathbf{r}_{\text{PCMP}}\right) \tag{10}$$

where $\omega_0 := \sqrt{\frac{k_0+k_1}{m}}$ is the reciprocal of the system's time constant, and $\mathbf{r}_{\text{PCMP}}$ is what we call the pseudo-Centroidal Moment Pivot (pseudo-CMP), defined as

$$\mathbf{r}_{\text{PCMP}} := w\mathbf{r}_{\text{CMP},0} + (1-w)\,\mathbf{r}_{\text{CMP},1} \tag{11}$$

with weighting factor

$$w := \frac{k_0}{k_0 + k_1} \tag{12}$$

In [5], we showed that the unstable part of the horizontal CoM dynamics is described by the motion of the instantaneous capture point (ICP), defined as[1]

$$\mathbf{r}_{\text{ic}} := \mathbf{P}\left(\mathbf{r} + \frac{\dot{\mathbf{r}}}{\omega_0}\right). \tag{13}$$

Differentiating this definition and comparing with (10) shows that the ICP dynamics are described by

$$\dot{\mathbf{r}}_{\text{ic}} = \omega_0\left(\mathbf{r}_{\text{ic}} - \mathbf{P}\mathbf{r}_{\text{PCMP}}\right). \tag{14}$$

This is a generalization of the results of [5], where we obtained a similar equation for a constant height, lumped inertia model. Here we have made no such assumptions.

[1] In [5], we used the definition $\mathbf{r}_{\text{ic}} := \mathbf{P}\mathbf{r} + \frac{\dot{\mathbf{r}}}{\omega_0}$ for models for which $\dot{z} = 0$. The definition in (13) could also have been used in that paper and seems to fit better here.

Equation (14) shows that the ICP diverges away from the pseudo-CMP. For the special case that both ω_0 and the pseudo-CMP are constant, we can solve (14) to obtain the time trajectory of the ICP,

$$\mathbf{r}_{\text{ic}}(t) = \left(\mathbf{r}_{\text{ic}}(0) - \mathbf{P}\mathbf{r}_{\text{PCMP}}\right)e^{\omega_0 t} + \mathbf{P}\mathbf{r}_{\text{PCMP}} \tag{15}$$

C. Rotational dynamics

Using (5), (6), and (7), the first three rows of (3) can be written as

$$\begin{aligned}\dot{\mathbf{k}}^c = \boldsymbol{\tau}^c_{\text{ext}} &= m\omega_0^2 w\left(\mathbf{r} - \mathbf{r}_{\text{CoP},0}\right) \times \Delta\mathbf{r}_{\text{CMP},0} \\ &\quad + m\omega_0^2(1-w)\left(\mathbf{r} - \mathbf{r}_{\text{CoP},1}\right) \times \Delta\mathbf{r}_{\text{CMP},1} + \boldsymbol{\tau}_n\end{aligned} \tag{16}$$

with w as defined in (12), and $\boldsymbol{\tau}_n := \boldsymbol{\tau}_{n,0} + \boldsymbol{\tau}_{n,1}$. This equation describes the relative contribution of CoP placements, CMP offsets, and normal torque on the rate of change of angular momentum and will be used as the basis of an angular momentum control law.

IV. Control framework

This section describes the controller design, which is based on the derivations in the previous section and can be summarized as follows:

1) Compute desired ground reaction wrench (IV-A)
2) Distribute ground reaction wrench over feet (IV-B)
3) Compute desired joint accelerations that are compatible with the ground reaction wrench (IV-C)
4) Compute joint torques using inverse dynamics, based on desired joint accelerations and distributed ground reaction wrenches (IV-D)

Fig. 2 shows the high level flow of information in terms of the mathematical symbols used later in this section.

A. Total ground reaction wrench

In this section we compute the desired total ground reaction wrench $\mathbf{W}^c_{\text{GR}} := \mathbf{W}^{c,0}_{\text{GR}} + \mathbf{W}^{c,1}_{\text{GR}}$ to be exerted on the robot. We make two assumptions:

1) $\Delta\mathbf{r}_{\text{CMP},0} = \Delta\mathbf{r}_{\text{CMP},1} = \Delta\mathbf{r}_{\text{CMP}}$, i.e. we choose the force vectors $\mathbf{f}_0$ and $\mathbf{f}_1$ to both point in the direction of $\mathbf{r} + \Delta\mathbf{r}_{\text{CMP}}$
2) $\boldsymbol{\tau}_n$ acts in the vertical direction, i.e. the ground surface normal points directly upward everywhere, although the feet may be at different heights

With these assumptions and the use of (10) and (16), the total ground reaction wrench can be written as

$$\mathbf{W}^c_{\text{GR}} = \begin{pmatrix} m\omega_0^2\left(\mathbf{r} - \mathbf{r}_{\text{PCoP}}\right) \times \Delta\mathbf{r}_{\text{CMP}} + \boldsymbol{\tau}_n \\ m\omega_0^2\left(\mathbf{r} - \mathbf{r}_{\text{PCoP}} - \Delta\mathbf{r}_{\text{CMP}}\right) \end{pmatrix} \tag{17}$$

where we define the pseudo-CoP [13] as

$$\mathbf{r}_{\text{PCoP}} := w\mathbf{r}_{\text{CoP},0} + (1-w)\,\mathbf{r}_{\text{CoP},1} \tag{18}$$

Comparing this to (11) shows that

$$\mathbf{r}_{\text{PCMP}} = \mathbf{r}_{\text{PCoP}} + \Delta\mathbf{r}_{\text{CMP}}. \tag{19}$$

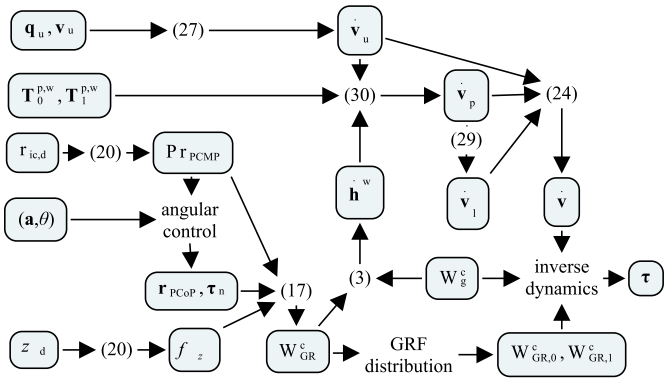

Figure 2. Controller overview

We will now proceed to derive control laws for $\mathbf{r}_{\mathrm{PCoP}}$, $\Delta\mathbf{r}_{\mathrm{CMP}}$, $\boldsymbol{\tau}_n$, and f_z, the total vertical force. This vertical force, together with the wrench distribution over the feet (section IV-B) will specify ω_0.

1) ICP control: We use (14) as the basis for an ICP control law:

$$\mathbf{P}\mathbf{r}_{\mathrm{PCMP}} = \mathbf{r}_{\mathrm{ic}} - \frac{1}{\omega_0}\dot{\mathbf{r}}_{\mathrm{ic,d}} + k_{\mathrm{ic}}\left(\mathbf{r}_{\mathrm{ic}} - \mathbf{r}_{\mathrm{ic,d}}\right) \tag{20}$$

where $\mathbf{r}_{\mathrm{PCMP}}$ is the desired location of the pseudo-CMP and $k_{\mathrm{ic}} > 0$ is a proportional gain. This control law determines the horizontal components of the desired pseudo-CMP and will be used to track a desired ICP trajectory, described by the desired ICP position $\mathbf{r}_{\mathrm{ic,d}}$ and velocity $\dot{\mathbf{r}}_{\mathrm{ic,d}}$, as provided by the state machine (see section V). The feedforward term involving $\dot{\mathbf{r}}_{\mathrm{ic,d}}$ greatly improves tracking performance compared to the control law presented in [6].

2) Angular control: Angular control is based on (16) and (17). We compute a tentative desired CMP offset using

$$\Delta\bar{\mathbf{r}}_{\mathrm{CMP}} = \mathrm{clip}\left(-\left(b_{\mathrm{r}}\mathbf{k}^c + k_r\mathbf{a}\theta\right) \times \mathbf{e}_z, \gamma\right)$$

where $(\mathbf{a}, \theta)$ forms an angle-axis representation of the relative orientation between the rigid body and a desired orientation, $b_r > 0$ and $k_r > 0$ are gains and the function $\mathrm{clip}\left(\cdot,\cdot\right)$ limits the magnitude of the CMP offset, and is defined as

$$\mathrm{clip}\left(\mathbf{x}, \gamma\right) = \begin{cases} \frac{\gamma}{\|\mathbf{x}\|}\mathbf{x} & \|\mathbf{x}\| > \gamma \\ \mathbf{x} & \text{otherwise} \end{cases}$$

With the pseudo-CMP from (20), we can now use (19) to compute a tentative desired pseudo-CoP, $\bar{\mathbf{r}}_{\mathrm{PCoP}}$. Since the pseudo-CoP, (18), is a weighted average of the individual foot CoPs, it must lie inside the convex hull spanned by the two foot polygons. If $\bar{\mathbf{r}}_{\mathrm{PCoP}}$ lies within the base of support, then we simply take $\Delta\mathbf{r}_{\mathrm{CMP}} = \Delta\bar{\mathbf{r}}_{\mathrm{CMP}}$ and $\mathbf{r}_{\mathrm{PCoP}} = \bar{\mathbf{r}}_{\mathrm{PCoP}}$. However, if $\bar{\mathbf{r}}_{\mathrm{PCoP}}$ lies outside the base of support, then we project it onto the base of support to obtain $\mathbf{r}_{\mathrm{PCoP}}$, and $\Delta\mathbf{r}_{\mathrm{CMP}}$ is found by applying (19). This means that we sacrifice orientation control for balance control, i.e. to obtain the desired pseudo-CMP computed by the balance control law (20).

We use $\boldsymbol{\tau}_n$ to dampen angular momentum in the z-direction:

$$\boldsymbol{\tau}_n := \mathbf{e}_z\left(-b_n\mathbf{e}_z \cdot \mathbf{k}^c\right)$$

with $b_n > 0$ a scalar gain.

3) Height control: The last row of (9) is used as the basis for a CoM height control law:

$$f_z = m\left(\ddot{z}_{\mathrm{d}} + g_z\right) + b_z\left(\dot{z}_{\mathrm{d}} - \dot{z}\right) + k_z\left(z_{\mathrm{d}} - z\right) \tag{21}$$

where f_z is the z-component of total ground reaction force $\mathbf{f}^c_{\mathrm{GR}} = \mathbf{f}_0 + \mathbf{f}_1$, $k_z > 0$ and $b_z > 0$ are PD gains, and z_{d}, $\dot{z}_{\mathrm{d}}$, and $\ddot{z}_{\mathrm{d}}$ are the desired CoM height and its first and second derivatives, respectively. These stem from a desired CoM height trajectory $z_{\mathrm{d}} = f\left(x\right)$, provided by the state machine (see section V), where x is the CoM position along the x-axis of Φ_w. We define the world frame Φ_w such that the x-axis points in the direction of the stairs, while the z-axis points opposite to the direction of gravity. The derivatives $\dot{z}_{\mathrm{d}}$ and $\ddot{z}_{\mathrm{d}}$ are obtained using the chain rule:

$$\dot{z}_{\mathrm{d}} = \frac{\mathrm{d}f\left(x\right)}{\mathrm{d}x}\dot{x}, \quad \ddot{z}_{\mathrm{d}} = \frac{\mathrm{d}^2f\left(x\right)}{\mathrm{d}x^2}\dot{x}^2 + \frac{\mathrm{d}f\left(x\right)}{\mathrm{d}x}\ddot{x}$$

We use the desired pseudo-CMP from (20) to compute a prediction of $\ddot{x}$, required to compute $\ddot{z}_{\mathrm{d}}$, using (10).

B. Ground reaction wrench distribution

In single support, the entire ground reaction wrench $\mathbf{W}^c_{\mathrm{GR}}$ is assigned to the stance foot. In double support however, the total ground reaction wrench should be distributed over the supporting feet.

Each foot ground reaction wrench is completely specified by the tuple $(\mathbf{r}_{\mathrm{CoP},i}, \Delta\mathbf{r}_{\mathrm{CMP},i}, k_i, \boldsymbol{\tau}_{\mathrm{n},i})$, as shown by (6). As noted in IV-A2, we choose $\Delta\mathbf{r}_{\mathrm{CMP},0} = \Delta\mathbf{r}_{\mathrm{CMP},1} = \Delta\mathbf{r}_{\mathrm{CMP}}$. Equation (18) shows that the foot CoPs must be chosen in such a way that the pseudo-CoP is a weighted average of the individual CoPs and hence lies on a line segment between $\mathbf{r}_{\mathrm{CoP},0}$ and $\mathbf{r}_{\mathrm{CoP},1}$. Choosing foot CoPs is done heuristically based on geometric relations, depicted and explained in Fig. 3. Given these foot CoPs and the overall pseudo-CoP, the weighting factor w is found by solving (18):

$$w = \frac{\|\mathbf{r}_{\mathrm{PCoP}} - \mathbf{r}_{\mathrm{CoP},1}\|}{\|\mathbf{r}_{\mathrm{CoP},0} - \mathbf{r}_{\mathrm{CoP},1}\|} \tag{22}$$

The z-component of the total ground reaction force, f_z, has been determined using (21). Using the fact that $\Delta z_{\mathrm{CMP}} = 0$, we can equate this to the z-component of $\mathbf{f}_0 + \mathbf{f}_1$ from (7):

$$f_z = k_0\left(z - z_{\mathrm{CoP},0}\right) + k_1\left(z - z_{\mathrm{CoP},1}\right) \tag{23}$$

allowing us to solve for k_0 and k_1:

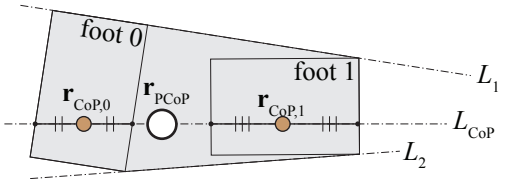

Figure 3. Overhead view of the support polygon showing the determination of individual foot CoPs during double support. Line L_{CoP} passes through the desired overall CoP and the intersection of lines L_1 and L_2, the edges of the support polygon that connect the foot polygons. The intersections of line L_{CoP} with the foot polygons define two line segments. The individual foot CoPs $\mathbf{r}_{\text{CoP},0}$ and $\mathbf{r}_{\text{CoP},1}$ are found as the center points of these line segments. If the desired overall CoP lies on the outside of a foot, so that it lies farther to the edge of the support polygon than the center of the line segment for that foot, then the CoP for that foot is chosen to be equal to the desired CoP (this case not shown). Note that the foot polygons reduce to line segments when the robot is on its toes.

$$k_0 = \frac{(1-w)\, f_z}{z - z_{\text{PCoP}}}, \quad k_1 = \frac{w f_z}{z - z_{\text{PCoP}}}$$

This also determines $\omega_0 = \sqrt{\frac{k_0+k_1}{m}}$. We use the same weighting factor w to distribute the normal moment $\boldsymbol{\tau}_n$ over the feet:

$$\boldsymbol{\tau}_{n,0} = (1-w)\, \boldsymbol{\tau}_n, \quad \boldsymbol{\tau}_{n,1} = w \boldsymbol{\tau}_n$$

This ensures that the foot which is planted more firmly is also used to exert most of the normal moment.

C. Compatible joint accelerations

Now that the external wrenches are known, we know the rate of change of momentum, according to (3). In this section, we compute joint accelerations that are compatible with this rate of change of momentum, while satisfying two secondary control goals: 1) foot position control, and 2) upper body control.

We will express the configuration of the robot as $\mathbf{Q} = \left(\mathbf{H}_p^w, \mathbf{q}_u, \mathbf{q}_l\right) \in \mathbb{R}^{4\times4} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_l}$, where $\mathbf{H}_p^w$ is the configuration of the pelvis with respect to the world, $\mathbf{q}_u \in \mathbb{R}^{n_u}$ contains the positions of the upper body joints (i.e. the set of all joints above the pelvis), and $\mathbf{q}_l \in \mathbb{R}^{n_l}$ contains the positions of all lower body joints (i.e. the set of all joints below the pelvis).

The robot's velocity is expressed non-redundantly as

$$\mathbf{v} = \begin{pmatrix} \mathbf{v}_p \\ \mathbf{v}_u \\ \mathbf{v}_l \end{pmatrix} \in \mathbb{R}^n \tag{24}$$

where $\mathbf{v}_p := \mathbf{T}_p^{w,w} \in \mathbb{R}^6$ is the twist of the pelvis with respect to the world, expressed in world frame, $\mathbf{v}_u = \dot{\mathbf{q}}_u$, $\mathbf{v}_l = \dot{\mathbf{q}}_l$ and $n = 6 + n_u + n_l$.

It can be shown that centroidal whole-body momentum $\mathbf{h}^c$ is linearly dependent on the velocity vector $\mathbf{v}$ [11], i.e.

$$\mathbf{h}^c = \mathbf{A}\left(\mathbf{Q}\right)\mathbf{v}$$

where $\mathbf{A}\left(\mathbf{Q}\right) \in \mathbb{R}^{6\times n}$ is called the Centroidal Momentum Matrix. Differentiating leads to:

$$\mathbf{A}\dot{\mathbf{v}} = \underbrace{\dot{\mathbf{h}}^c - \dot{\mathbf{A}}\mathbf{v}}_{\mathbf{b}} \tag{25}$$

where the dependence of $\mathbf{A}$ on $\mathbf{Q}$ has been omitted for clarity of presentation. Note that since (25) represents an underconstrained system of equations, we can satisfy secondary control goals. Using (24), we partition (25) as

$$\begin{pmatrix} \mathbf{A}_p & \mathbf{A}_u & \mathbf{A}_l \end{pmatrix} \begin{pmatrix} \dot{\mathbf{v}}_p \\ \dot{\mathbf{v}}_u \\ \dot{\mathbf{v}}_l \end{pmatrix} = \mathbf{b} \tag{26}$$

The upper body joints could be used to perform a manipulation task or to assist in balance. However, in the current controller implementation we will instead use a simple joint space PD control law:

$$\dot{\mathbf{v}}_u = k_u\left(\mathbf{q}_{u,\text{d}} - \mathbf{q}_u\right) + b_u\left(\mathbf{v}_{u,\text{d}} - \mathbf{v}_u\right) \tag{27}$$

where $k_u > 0$ and $b_u > 0$ are PD gains and $\mathbf{q}_{u,\text{d}}$ and $\mathbf{v}_{u,\text{d}}$ are desired positions and velocities, which are currently set to zero at all times.

Joint accelerations for the lower body are determined based on desired spatial accelerations of each foot with respect to the world, expressed in world frame, $\dot{\mathbf{T}}_i^{w,w}$. These desired foot spatial accelerations can be used for foot position control during the swing phase, as well as to satisfy a constraint such as zero acceleration with respect to the world during stance phase. The twist of foot i with respect to the pelvis, expressed in world frame, is

$$\mathbf{T}_i^{w,p} = \mathbf{T}_w^{w,p} + \mathbf{T}_i^{w,w} = -\mathbf{v}_p + \mathbf{T}_i^{w,w} \tag{28}$$

where we have used the definition of $\mathbf{v}_p$, as well as properties (1) and (2). The lower body joint velocity vector $\mathbf{v}_l$ can be further split up as

$$\mathbf{v}_l = \begin{pmatrix} \mathbf{v}_{l,0} \\ \mathbf{v}_{l,1} \end{pmatrix}$$

where $\mathbf{v}_{l,i}$ is the joint velocity vector for leg number i. The twist of foot i with respect to the pelvis is related to $\mathbf{v}_{l,i}$ by the Jacobian $\mathbf{J}_i^{w,p}$, which spans all the joints between the pelvis and foot i:

$$\mathbf{T}_i^{w,p} = \mathbf{J}_i^{w,p}\mathbf{v}_{l,i}$$

Differentiating this equation, substituting (28), and rearranging results in

$$\begin{aligned} \mathbf{J}_i^{w,p}\dot{\mathbf{v}}_{l,i} &= \dot{\mathbf{T}}_i^{w,p} - \dot{\mathbf{J}}_i^{w,p}\mathbf{v}_{l,i} \\ &= -\dot{\mathbf{v}}_p + \underbrace{\dot{\mathbf{T}}_i^{w,w} - \dot{\mathbf{J}}_i^{w,p}\mathbf{v}_{l,i}}_{\mathbf{c}_i} \end{aligned}$$

so that

$$\dot{\mathbf{v}}_l = \underbrace{\begin{pmatrix} \left(\mathbf{J}_0^{w,p}\right)^{-1}\mathbf{c}_0 \\ \left(\mathbf{J}_1^{w,p}\right)^{-1}\mathbf{c}_1 \end{pmatrix}}_{\mathbf{c}} - \underbrace{\begin{pmatrix} \left(\mathbf{J}_0^{w,p}\right)^{-1} \\ \left(\mathbf{J}_1^{w,p}\right)^{-1} \end{pmatrix}}_{\mathbf{J}} \dot{\mathbf{v}}_p \tag{29}$$

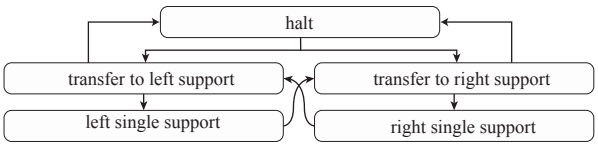

Figure 4. State machine. Boxes represent states and arrows represent possible state transitions. The halt and transfer states assume both feet are in contact with the ground. The transitions from the halt state to a transfer state and back are governed by a user command, telling the robot to start or stop walking. The transition from transfer to single support is detailed in section V-B. Finally, the transition from single support to transfer occurs when the foot trajectory has finished executing, or when the swing foot hits the ground early.

assuming the inverses of $\mathbf{J}_0^{w,p}$ and $\mathbf{J}_1^{w,p}$ exist. To mitigate issues at or near singularities, we use the damped least squares method [14], complemented by a singularity escape method [15]. The singularity escape method is used to start bending a stretched knee and consists of using an acceleration vector $\dot{\mathbf{v}}_{l,i}$ corresponding to the smallest singular value of the Jacobian.

Using (29), we can rewrite (26) as

$$\mathbf{A}_p\dot{\mathbf{v}}_p + \mathbf{A}_u\dot{\mathbf{v}}_u + \mathbf{A}_l\left(\mathbf{c} - \mathbf{J}\dot{\mathbf{v}}_p\right) = \mathbf{b}$$

which can be solved for the pelvis acceleration:

$$\dot{\mathbf{v}}_p = \left(\mathbf{A}_p - \mathbf{A}_l\mathbf{J}\right)^{-1}\left(\mathbf{b} - \mathbf{A}_u\dot{\mathbf{v}}_u - \mathbf{A}_l\mathbf{c}\right) \tag{30}$$

At this point, the acceleration vector $\dot{\mathbf{v}}$ is known.

D. Inverse dynamics

The distributed ground reaction wrenches from section IV-B and the desired joint accelerations from IV-C are used as the input to an inverse dynamics algorithm [10], resulting in joint torques $\boldsymbol{\tau}$ for all joints, as well as a residual wrench that should be exerted across the '6 DoF joint' that connects the pelvis to the world. Because the joint accelerations are compatible with the total ground reaction wrench, the residual wrench is identically zero.

V. State machine

A finite state machine (depicted and detailed in Fig. 4) was designed to implement stair climbing behavior. The output of the state machine is used in the lower level control framework presented in the previous section. The output consists of the desired spatial acceleration of each of the feet (section V-A), a desired CoM height trajectory (V-B), and a desired ICP trajectory (V-C).

A. Foot spatial acceleration

The desired spatial acceleration $\dot{\mathbf{T}}_i^{w,w}$ of foot i with respect to Φ_w is computed using PD control on $SE(3)$ with an added feed forward term. In each state, we determine a desired homogeneous transformation $\mathbf{H}_{i,\mathrm{d}}^w$, twist $\mathbf{T}_{i,\mathrm{d}}^{w,w}$, and feed forward spatial acceleration $\dot{\mathbf{T}}_{i,\mathrm{ff}}^{w,w}$. We then compute a tentative desired spatial acceleration $\dot{\tilde{\mathbf{T}}}_i^{w,w}$ using a double-geodesic PD control law [16], i.e.

$$\dot{\tilde{\mathbf{T}}}_i^{w,w} = \mathrm{PD}\left(\mathbf{H}_{i,\mathrm{d}}^w, \mathbf{T}_{i,\mathrm{d}}^{w,w}\right) + \dot{\mathbf{T}}_{i,\mathrm{ff}}^{w,w}$$

The motion of a swing foot is unconstrained, so we use $\dot{\mathbf{T}}_i^{w,w} = \dot{\tilde{\mathbf{T}}}_i^{w,w}$ in this case. On the other hand, the motion of a stance foot is constrained: the foot may only rotate about its front edge, which is in contact with the ground. In this case we project the tentative desired spatial acceleration onto the unconstrained motion subspace [10].

The desired values $\mathbf{H}_{i,\mathrm{d}}^w$, $\mathbf{T}_{i,\mathrm{d}}^{w,w}$, and $\dot{\mathbf{T}}_{i,\mathrm{ff}}^{w,w}$, are determined as follows. For the position trajectory of a swing foot, we use a polynomial spline from the initial actual foot position to a final desired position, via a waypoint. The final desired position is specified based on a constant step width and the constant tread and rise of the stairs, of which the robot has perfect knowledge. The waypoint is tuned so that unwanted collisions are avoided. Swing time is constant. The desired orientation of the swing foot is set using a constant pitch value, while roll and yaw are zero. For the stance foot (or feet), we only specify a desired foot pitch which depends on whether the leg is leading or trailing, and set $\mathbf{T}_{i,\mathrm{d}}^{w,w} = 0$, and $\dot{\mathbf{T}}_{i,\mathrm{ff}}^{w,w} = 0$.

B. CoM height trajectory

For the CoM height trajectory $z_\mathrm{d} = f(x)$, we alternate between a constant height and a fifth-order polynomial (see Fig. 5):

$$f(x) = \begin{cases} z_j & x_{\mathrm{initial},j} \le x < x_{\mathrm{final},j} \\ \sum_{i=0}^{5} a_{i,j}x^i & x_{\mathrm{final},j} \le x < x_{\mathrm{initial},j+1} \end{cases}$$

for $j = 0, 1, \ldots$. Here z_j is set to a constant height above the toe of the stance foot, $x_{\mathrm{initial},j}$ is the x-coordinate of the CoM at the start of the jth transfer state and $x_{\mathrm{final},j}$ is halfway between the x-coordinate of the toe of the stance foot and the toe of the upcoming planned footstep position. The coefficients $a_{i,j}$ of the polynomials are chosen such that the initial and final heights line up with the constant parts, and the initial and final slopes and second derivatives are zero. This trajectory keeps the height mostly constant during double support.

The transition from a transfer state to single support is based on the orbital energy associated with the polynomial part of the trajectory [17]. Orbital energy for a variable-height inverted pendulum is defined as:

$$E_\mathrm{orbit} := \frac{1}{2}\dot{s}^2h^2(s) + g_z s^2 f(s) - 3g_z\int_0^s f(\xi)\,\xi\mathrm{d}\xi \tag{31}$$

where

$$h(s) := f(s) - f'(s)\,s, \quad s := x - x_\mathrm{toe}$$

with x_toe the x-coordinate of the stance foot toe, which acts as the pivot of the inverted pendulum. For a polynomial height trajectory, (31) can be evaluated in closed form. If $E_\mathrm{orbit} > 0$, then there will be enough momentum

to make the CoM ground projection pass the toe and continue on its way, instead of falling back. Hence, we transfer to the single support state once the orbital energy associated with the polynomial part of the trajectory is greater than a specified positive limit.

A linear trajectory $f(x) = ax + b$ was also tried, but limited the robot's stair climbing ability due to ground collisions and the knee joint limit for the trailing leg.

C. ICP trajectory

To attain forward motion while maintaining balance, we specify a desired ICP trajectory that 'ping-pongs' from waypoint $\mathbf{r}_{\mathrm{ic,j}}$ to waypoint $\mathbf{r}_{\mathrm{ic,j+1}}$ along straight lines (see Fig. 7). The waypoints lie at the centers of the front edge of the foot polygons. The motion between waypoints follows an exponential time trajectory:

$$\mathbf{r}_{\mathrm{ic,d}}(t) = \frac{e^{\omega_0 t} - 1}{e^{\omega_0 t_\mathrm{f}} - 1}(\mathbf{r}_{\mathrm{ic,j+1}} - \mathbf{r}_{\mathrm{ic,j}}) + \mathbf{r}_{\mathrm{ic,j}} \qquad (32)$$

where t_f is the time at which the desired ICP will arrive at $\mathbf{r}_{\mathrm{ic},j+1}$. This trajectory is initialized at the start of single support. Its form comes from (15): we assume a constant value of ω_0 and a constant nominal pseudo-CMP. The relationship between this nominal pseudo-CMP and the final time t_f is found by solving (15) with $t = t_\mathrm{f}$:

$$t_\mathrm{f} = \ln\left(\frac{\|\mathbf{r}_{\mathrm{ic,j+1}} - \mathbf{Pr}_{\mathrm{PCMP}}\|}{\|\mathbf{r}_{\mathrm{ic,j}} - \mathbf{Pr}_{\mathrm{PCMP}}\|}\right)/\omega_0$$

We prefer to rely mostly on pseudo-CoP placement to obtain a given pseudo-CMP, because any nonzero CMP offset $\Delta\mathbf{r}_{\mathrm{CMP}}$ results in a change in angular momentum according to (16), which compromises orientation control. However, the pseudo-CoP must lie within the base of support. We therefore choose a nominal desired pseudo-CMP location inside the base of support. This determines t_f, which in turn determines trajectory (32).

VI. Results

We now present simulation results. First, we describe the simulation setup (section VI-A), followed by a high level description of the climbing ability (VI-B). Finally, we discuss the tracking behavior of the controller (VI-C).

A. Simulation setup

Simulations were performed using the Yobotics Simulation Construction Set [18], which implements Featherstone's articulated rigid body algorithm [10]. We use a fourth order Runge-Kutta integrator with a fixed timestep of 0.1 ms. Ground contact forces were calculated using a spring-damper ground model.

The simulated robot has 6 degrees of freedom in each leg, 3 in the spine, 3 in the neck, and 4 in each arm. There are three degrees of freedom at each hip, one at each knee, and two at each ankle. All joints are revolute. Each foot has four contact points at which the ground can exert forces: two at the toes and two at the heels. The total mass of the robot is 100.51 kg. Sensor noise was not simulated.

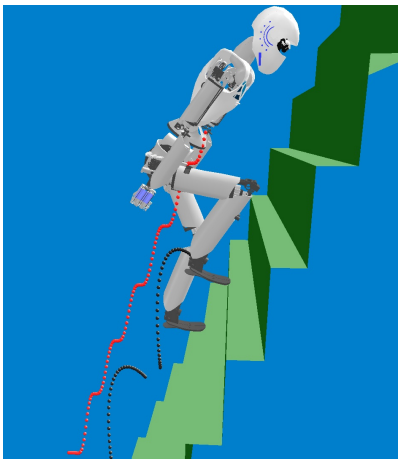

Figure 5. Robot climbing stairs. CoM height trajectory and trajectory of right foot are shown.

B. Overall stair climbing ability

The simulated robot can climb stairs with a tread of 15 cm and a rise of 35 cm at 7.9 s per ten steps. Swing time was set to 0.45 s. See Fig. 5 for a snapshot of the robot as the right foot is about to touch down. A video showing simulation results is also supplied with this paper.

At any given time, only the toes of a stance foot are in contact with the ground. The knee of the trailing leg is almost completely stretched at the end of the transfer state. The toes are very close to the edge of the step, which helps avoid shin and knee collisions and is enabled by excellent foot position and orientation tracking performance.

C. Tracking performance

Fig. 6 shows the states that the controller goes through and the tracking behavior during a 10-step stair climb. The robot is initialized with significant errors in CoM height, ICP, pelvis orientation, and foot orientation. After an initial settling period, the robot is commanded to start climbing at $t = 1.72$ s.

CoM height error is generally kept within 1 cm as the robot ascends the stairway.

Although the swing foot position error increases steeply at the start of each single support state due to the fact that the new swing leg is close to the knee-lock singularity, the position error at touchdown is only around 3 mm.

ICP tracking (see also Fig. 7) shows a similar trend. At state transitions, the ICP error increases markedly, because ω_0 in (13) changes rapidly due to force redistribution over the feet. The ICP quickly converges back to the desired value during transfer states, and at the transition to single support, the error is generally around 2.5 mm.

Pelvis orientation is controlled with low gain, as this produces a more natural looking climbing motion, and tracking is not critical. This explains the fairly large errors.

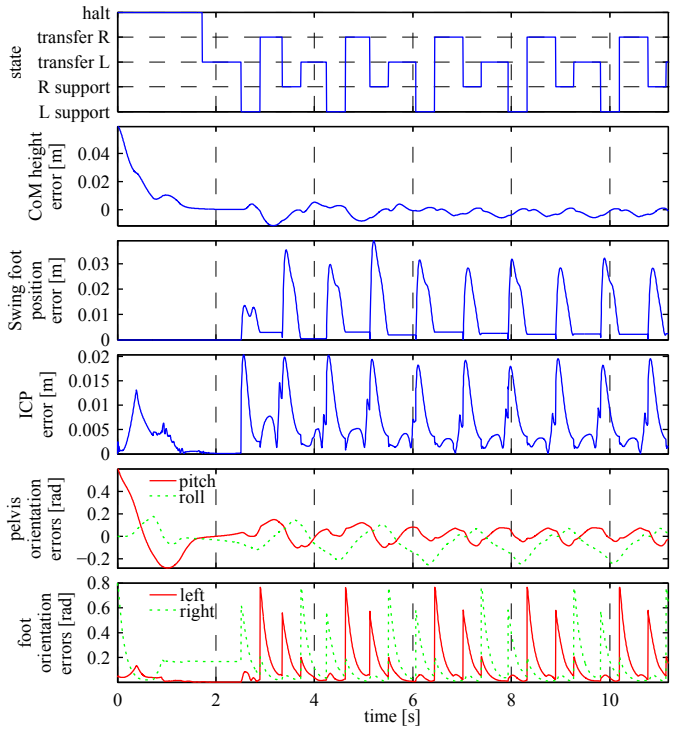

Figure 6. State evolution and tracking behavior of the controller.

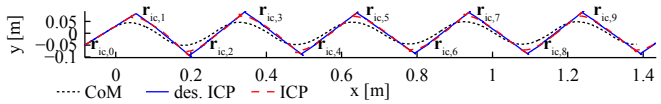

Figure 7. Overhead view of CoM and ICP trajectories. Not to scale.

Since we use a step signal for the foot orientation trajectory with jumps at the state changes, the foot orientation error signal looks like a series of step responses: after an initial spike, the error rapidly converges to a fraction of a radian.

VII. Discussion

The presented controller is model-based, and was tested without sensor noise and discrepancies between the controller's model and the simulated robot. This leads to question the controller's robustness, which has yet to be evaluated and will be the main focus of future work. To enhance robustness, we envision the use of ground reaction force feedback and adaptive control methods.

The two assumptions made in section IV-A concerning the direction of ground reaction forces and direction of normal ground reaction torque simplify the controller design (especially the distribution of ground reaction forces), but limit the controller's area of application somewhat. These assumptions are however not fundamental.

The current implementation of the controller lacks some useful features, such as a good way to avoid collisions between the robot and the ground, a way to avoid slamming into joint limits, and upper body control in task space. These features may be added later.

VIII. Conclusion

This paper introduced a novel force control scheme which allowed a simulated bipedal robot to quickly climb a steep set of stairs. Whole-body momentum was used as a unifying concept, allowing simultaneous achievement of balance control by tracking an instantaneous capture point trajectory using Centroidal Moment Pivot placement, height control, and pelvis orientation control. A simulation study without sensor noise and with perfect model knowledge showed excellent tracking behavior and a fairly natural-looking ascent.

References

[1] 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 *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, vol. 2. IEEE, 2003, pp. 1644–1650.

[2] S.-H. Lee and A. Goswami, "A Momentum-based Balance Controller for Humanoid Robots on Non-level and Non- stationary Ground," Accepted, Autonomous Robots, 2012.

[3] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schaffer, and G. Hirzinger, "Bipedal walking control based on Capture Point dynamics," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Sep. 2011, pp. 4420–4427.

[4] 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*, Osaka, Japan, 2012.

[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," *Int. J. Robotics Research*, vol. 31, no. 9, pp. 1094–1113, Jul. 2012.

[6] J. Pratt, T. Koolen, T. de Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus, "Capturability-Based Analysis and Control of Legged Locomotion, Part 2: Application to M2V2, a Lower Body Humanoid," Accepted, Int. J. Robotics Research, Aug. 2012.

[7] M. B. Popovic, A. Goswami, and H. Herr, "Ground Reference Points in Legged Locomotion: Definitions, Biological Trajectories and Control Implications," *Int. J. Robotics Research*, vol. 24, no. 12, pp. 1013–1032, 2005.

[8] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1994.

[9] V. Duindam, "Port-Based Modeling and Control for Efficient Bipedal Walking Robots," Ph.D. dissertation, University of Twente, 2006.

[10] R. Featherstone, *Rigid body dynamics algorithms*. Springer-Verlag New York Inc, 2008.

[11] D. E. Orin and A. Goswami, "Centroidal Momentum Matrix of a humanoid robot: Structure and properties," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2008, pp. 653–659.

[12] L. Sentis, J. Park, and O. Khatib, "Modeling and control of multi-contact centers of pressure and internal forces in humanoid robots," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, vol. 1, pp. 453–460, Oct. 2009.

[13] P. Sardain and G. Bessonnet, "Forces Acting on a Biped Robot. Center of Pressure-Zero Moment Point," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 5, pp. 630–637, Sep. 2004.

[14] S. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," University of California, San Diego, Tech. Rep., 2004.

[15] S. Oetomo, D. and Ang Jr, M. and Lim, "Singularity Handling on Puma in Operational Space Formulation," in *ISER '00 Experimental Robotics VII*. Springer, 2001, pp. 491–500.

[16] F. Bullo and R. Murray, "Proportional derivative (PD) control on the Euclidean group," California Institute of Technology, Tech. Rep., 1995.
[17] J. E. Pratt and S. V. Drakunov, "Derivation and Application of a Conserved Orbital Energy for the Inverted Pendulum Bipedal Walking Model," *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 1, no. 2, pp. 4653–4660, Apr. 2007.
[18] IHMC, "Simulation Construction Set," 2012. [Online]. Available: http://ihmc.us/groups/scs/