Noname manuscript No.
(will be inserted by the editor)

A Momentum-based Balance Controller for Humanoid Robots on Non-level and Non-stationary Ground

Sung-Hee Lee · Ambarish Goswami

Received: date / Accepted: date

Abstract Recent research suggests the importance of controlling rotational dynamics of a humanoid robot in balance maintenance and gait. In this paper, we present a novel balance strategy that controls both linear and angular momentum of the robot. The controller's objective is defined in terms of the desired momenta, allowing intuitive control of the balancing behavior of the robot. By directly determining the ground reaction force (GRF) and the center of pressure (CoP) *at each support foot* to realize the desired momenta, this strategy can deal with non-level and non-stationary grounds, as well as different frictional properties *at each foot-ground contact*. When the robot cannot realize the desired values of linear and angular momenta simultaneously, the controller attributes higher priority to linear momentum at the cost of compromising angular momentum. This creates a large rotation of the upper body, reminiscent of the balancing behavior of humans. We develop a computationally efficient method to optimize GRFs and CoPs at individual foot by sequentially solving two small-scale constrained linear least-squares problems. The balance strategy is demonstrated on a simulated humanoid robot under experiments such as recovery from unknown external pushes and balancing on non-level and moving supports.

S.-H. Lee
School of Information and Communications
Gwangju Institute of Science and Technology, South Korea
E-mail: shl@gist.ac.kr

A. Goswami
Honda Research Institute
Mountain View, CA, USA
E-mail: agoswami@honda-ri.com

1 Introduction

Even after several decades of research balance maintenance has remained one of the most important issues of humanoid robots. Although the basic dynamics of balance are currently understood, robust and general controllers that can deal with discrete and non-level foot support as well as large, unexpected and unknown external disturbances such as moving support, slip and trip have not yet emerged. Especially, in comparison with the elegance and versatility of human balance, present-day robots appear quite deficient. In order for humanoid robots to coexist with humans in the real world, more advanced balance controllers that can deal with a broad range of environment conditions and external perturbations need to be developed.

Until recently, most balance control techniques have attempted to maintain balance by controlling only the linear motion of a robot. For example, Kagami et al. [15] and Kudoh et al. [19] proposed methods to change the input joint angle trajectories to modify the position of the Center of Pressure (CoP), a point within the robot's support area through which the resultant Ground Reaction Force (GRF) acts. When the CoP, computed from the input joint motion, leaves the support base, indicating a possible toppling of a foot, the motion is modified to bring the CoP back inside the support base while the robot still follows the desired linear motion of the Center of Mass (CoM). The rotational motion of the robot remains more or less ignored in these approaches.

However, rotational dynamics of a robot plays a significant role in balance [18]. Experiments on human balance control also show that humans tightly regulate angular momentum during gait [34], which suggests the strong possibility that angular momentum may be important in humanoid movements.

In fact, both angular and linear momenta must be regulated to completely control the CoP. The fundamental quantities and the relations between them are schematically depicted in Fig. 1 and described subsequently.

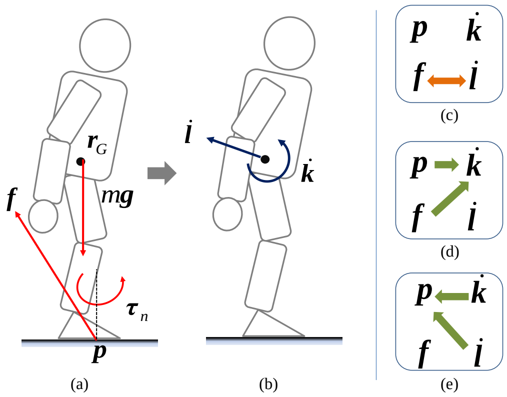


Fig. 1: The external forces and torques in (a) are solely responsible for the centroidal momentum rate change in (b). (c): Linear momentum rate change $\dot{\boldsymbol{l}}$ has a one-to-one correspondence with the GRF $\boldsymbol{f}$. (d): The centroidal angular momentum rate change $\dot{\boldsymbol{k}}$ is determined by both $\boldsymbol{f}$ and CoP location $\boldsymbol{p}$. (e): Conversely, $\boldsymbol{p}$ is determined by both $\dot{\boldsymbol{l}}$ and $\dot{\boldsymbol{k}}$.

Fig. 1(a) shows all the external forces that act on a freely standing humanoid: the GRF $\boldsymbol{f}$, the Ground Reaction Moment $\boldsymbol{\tau}_n$ normal to the ground, and the weight $m\boldsymbol{g}$ of the robot, where m is the total robot mass and $\boldsymbol{g}$ is the acceleration due to gravity. According to D'Alembert's principle, the sums of external moments and external forces, respectively, are equivalent to the rates of change of angular and linear momenta, respectively, of the robot. The mathematical expressions for these relationships are given by (1) and (2). Fig. 1(b) depicts the robot's rate of change of angular momentum about the CoM, $\dot{\boldsymbol{k}}$, and linear momentum, $\dot{\boldsymbol{l}}$, respectively.

$$\dot{\boldsymbol{k}} = (\boldsymbol{p} - \boldsymbol{r}_G) \times \boldsymbol{f} + \boldsymbol{\tau}_n \tag{1}$$

$$\dot{\boldsymbol{l}} = m\boldsymbol{g} + \boldsymbol{f} \tag{2}$$

In the above equations, $\boldsymbol{r}_G$ is the CoM location and $\boldsymbol{p}$ is the CoP location. Together $\boldsymbol{k}$ and $\boldsymbol{l}$ is a 6×1 vector called the spatial centroidal momentum $\boldsymbol{h} = [\boldsymbol{k}^T \; \boldsymbol{l}^T]^T$, which was studied in [30]. In this paper, we will call it spatial momentum, or simply the momentum of the robot. Note that the spatial centroidal momentunm is computed with respect to a frame which is aligned to the world frame and located at the overall CoM of the robot. Also the frame is instantaneously frozen with respect to the world frame.

Indeed, as noted in [24], the (spatial) momentum rate change has a one-to-one relationship with the GRF and CoP. From (2) and as shown in Fig. 1(c), $\dot{\boldsymbol{l}}$ is completely determined by $\boldsymbol{f}$ and vice versa. Furthermore, from (1) and Fig. 1(d), a complete description of $\dot{\boldsymbol{k}}$ needs both $\boldsymbol{f}$ and $\boldsymbol{p}$. Conversely, $\boldsymbol{p}$ depends on both $\dot{\boldsymbol{k}}$ and $\dot{\boldsymbol{l}}$, which is shown in Fig. 1(e).[1] This last sentence implies that a complete control of $\boldsymbol{p}$ is impossible without controlling both momenta.

Based on this fundamental relation researchers have developed balance maintenance methods that controls both the linear and angular components of the spatial momentum [1, 16, 24]. We will call balance controllers of this approach *momentum-based balance controllers*.

Some momentum-based balance control approaches define the desired rotational behavior of the controller in terms of the CoP [1,24] while others use angular momentum [16]. Although the GRF-CoP combination has a one-to-one relationship with momentum rate changes, their significance regarding balance are very different, and is worth discussing. Whereas the former characterize the magnitude, direction and point of application of the external forces, the latter describes the resulting motion of a robot. The unilateral nature of robot-ground contact and friction limits impose important direct constraints on the range of GRF and CoP. These influence the achievable range of momentum rate change, but only indirectly. On the other hand, it is more natural to describe the aggregate motion of a robot in terms of momentum.

In this paper we present a new momentum-based balance controller that uses both the momentum and the GRF-CoP for their respectively appropriate purposes: We use momentum to define control objectives as well as to compute joint motions while GRF and CoP are used as constraints.

In this method, we first specify the desired momentum rate change for balance (Sec. 3.2). However, the desired momentum rate change may not always be physically realizable due to several constraints on the foot-ground contact. First, the CoP cannot be outside the robot's support base.[2] Second, the GRF must be unilateral in nature, and can never attract the robot towards the ground. Third, the GRF must satisfy the friction limit of the foot-ground surface, so as not to cause slip.

[1] The normal torque $\boldsymbol{\tau}_n$ also affects $\dot{\boldsymbol{k}}$ in the transverse plane. Actually $\boldsymbol{f}$, $\boldsymbol{p}$, and $\boldsymbol{\tau}_n$ together constitute the 6 variables that correspond to the 6 variables of the spatial momentum. Usually $\boldsymbol{\tau}_n$ is omitted in the discussion for simplicity because its magnitude is small.

[2] During single support, the support base is identical to the foot contact area, whereas during double support on level ground, the support base is equivalent to the convex hull of the support areas of the two feet.

Therefore, in the next step we determine the admissible values of GRF and CoP that will create the desired momentum rate change as close as possible while being physically realizable. Specifically, in order to make the controller robustly applicable to non-level and non-stationary ground, we directly determine admissible *foot* GRFs and *foot* CoPs, without using more conventional *net* GRF and *net* CoP of the robot. Assuming planar contact between the ground and each foot, the foot GRF is the ground reaction force acting on an individual foot and foot CoP is the location where its line of action intersects the foot support plane. Using the values of admissible foot GRF and foot CoP we recalculate momentum rate changes – these are the *admissible* momentum rate changes (Sec. 3.4).

In the subsequent step we resolve the joint accelerations given the admissible momentum rate change, desired joint accelerations for the upper body, and desired motion of the feet. Finally we compute necessary joint torques to create the joint accelerations and the admissible external forces using inverse dynamics (Sec. 3.5). Fig. 2 shows the block diagram for the controller.

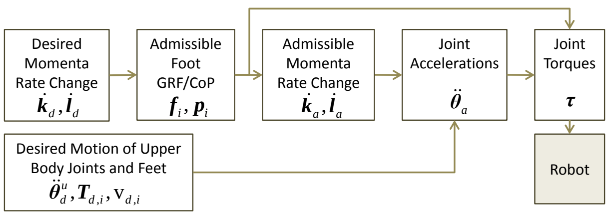


Fig. 2: Overview of Momentum-Based Balance Controller. $\ddot{\boldsymbol{\theta}}_d^u$ is the desired joint accelerations for the upper body. $\boldsymbol{T}_{d,i}$ and $\mathbf{v}_{d,i}$ are the desired configuration and spatial velocity of each foot ($i = r, l$). Subscripts d and a imply *"desired"* and *"admissible,"* respectively.

During double support, the computation of foot GRFs and foot CoPs from the desired momentum rate change is an under-determined problem. This allows us to pursue an additional optimality criterion in the solution. In this paper, we minimize the ankle torques while generating the desired momentum rate change. Minimizing ankle torque is important because typically the ankle torque is more constrained than others in that it should not cause foot tipping.

Specifically, we show that computing optimal foot GRFs and foot CoPs that minimizes ankle torques can be achieved by solving two simple constrained linear least-squares problems. Our simulation experiments show that this new optimization method is significantly faster than the conventional quadratic programming approach to solve the same problem.

The main unique contributions of this paper can be listed as follows:

1. Our momentum-based control framework determines the desired momenta but before attempting to reach them, it first checks to make sure that the momenta targets are physically attainable by computing their admissible values.
2. The optimal foot GRFs and foot CoPs are computed quickly by solving two small-scale linear least-squares problems.
3. The framework is sufficiently general to support a momentum-based stepping algorithm, as reported recently [49].

We will present a number of simulation experiments including pushing the single or double-supported robot in various directions, and maintaining balance when two feet are on separate moving supports with different inclinations and velocities.

The remainder of this paper is organized as follows. After discussing related work in Section 2, we detail the momentum-based balance control framework in Section 3. Section 4 reports the simulation experiments. Section 5 provides the discussion and the future work.

2 Related Work

Starting from the early work of [46], researchers have developed numerous techniques for biped balance control using various approaches. Among these are joint control strategies using ankle or hip [36,40], whole body control approaches [5, 15, 16, 32, 41, 44], methods that find optimal control policies [27, 50], and reflex controllers [12]. In this section we focus on the research relevant to momentum-based balance control.

The importance of angular momentum in humanoid walking was reported by Sano and Furusho as early as 1990 [36]. However, it was much later before its importance for balance maintenance of human and humanoid robots started to be seriously explored [10,16,34]. Sano and Furusho [36], and Mitobe et al. [26] showed that it is possible to generate the desired angular momentum by controlling the ankle torque. Kajita et al. [16] included angular momentum criteria into the whole body control framework for balance maintenance. After expressing desired linear and angular momenta as linear functions of the generalized velocities, they determined the joint velocities that achieved both momenta.

Komura et al. [18] presented a balance controller that can counteract rotational perturbations using the Angular Momentum Pendulum Model (AMPM). This model augments the well known 3D Linear Inverted

Pendulum Model (LIPM) [17] with the additional capability of possessing centroidal angular momentum. Naksuk et al. [29] proposed an iterative method to compute joint trajectories of humanoid robots to satisfy the desired CoM trajectory and to minimize the centroidal angular momentum. Other papers that deal with angular momentum for balance and gait include [3, 20, 38, 45, 48].

Abdallah and Goswami [1] defined balance control objectives in terms of CoM and CoP, and achieved this goal by controlling the rate of change of linear and angular momenta of a reduced model humanoid robot. The joint accelerations to generate the target momentum rate change were resolved using the Moore-Penrose pseudo-inverse.

In an important recent work in the field of animation Macchietto et al. [24] also defined balance control objectives in terms of CoM and CoP, and computed the desired momentum rate change. They employed the Centroidal Momentum Matrix [30] to compute joint accelerations, followed by computing necessary joint torques using inverse dynamics. We have adopted the same process to determine joint accelerations and torques.

Hofmann et al. [11] presented a method that controls CoM by modulating angular momentum under large external perturbations. It gives higher priority to controlling linear momentum over angular momentum to enhance the performance of the balance controller. We also give higher priority to attaining the desired linear momentum when both momenta cannot be simultaneously satisfied.

Similar to [1, 11, 16, 24], we also control both the linear and angular components of the spatial momentum of the robot for balance maintenance.

Our method improves the method in [16] by providing a step to check for the admissibility of the desired values of linear and angular momenta. Our work is also different from [1] and [24] in that we define the balance control objectives more intuitively in terms of linear and angular momenta and not in terms of the net CoP.

Furthermore, our method computes contact forces at each support foot, and therefore can be used both during double-support and single-support and also on non-level, discrete, and non-stationary grounds, whereas [1, 11, 24] consider only single-support.

Table 1 illustrates how the existing methods treat momentum, GRF, and CoP in formulating balance and gait strategies. Robot gait planning methods using reduced models such as [5, 17] (Table 1 (a)) compute the necessary CoM trajectory which ensures balance for a specified desired CoP trajectory. This is done using reduced models such as the LIPM. As can be seen from Fig. 1(e) CoP depends on both linear and angular momenta rate changes, so CoM cannot be uniquely determined solely from CoP. This was possible in [5, 17] because the reduced model used in those works approximated the robot as a point mass, which can only possess a *zero* angular momentum.

In the Resolved Momentum Control approach [16], both desired linear and angular momenta are used to determine joint motion for posture change (Table 1 (b)). However, the admissibility of the CoP is not considered so the robot may lose balance if the values of input desired momenta are high. The methods in Table 1 (c) determine the desired angular momentum rate change given desired CoP and linear momentum rate change. In our current method (Table 1 (d)), starting from the desired linear/angular momenta rate changes, we first determine admissible foot GRFs and CoPs, and then compute corresponding admissible momenta rate changes.

Table 1: The diagrams show how each method on balance control or gait pattern generation treats momenta ($\boldsymbol{k}, \boldsymbol{l}$), GRF ($\boldsymbol{f}$), and CoP ($\boldsymbol{p}$). A pair of solid lines determines the target value together. The dotted line shows the determination process of linear motion and force. The subscript "d" indicates a desired input to the controller and superscript "$*$" indicates that the quantity is used to determine control output such as joint torques.

(a) Kajita et al. [17], Choi et al. [5]	(b) Kajita et al. [16]	(c) Abdallah and Goswami [1], Macchhietto et al. [24]	(d) Lee and Goswami [22], this paper
$\begin{matrix} \boldsymbol{p}_d & \dot{\boldsymbol{k}}_d = 0 \\ \boldsymbol{f} & \dot{\boldsymbol{l}}^* \end{matrix}$	$\begin{matrix} \boldsymbol{p} & \dot{\boldsymbol{k}}^*_d \\ \boldsymbol{f} & \dot{\boldsymbol{l}}^*_d \end{matrix}$	$\begin{matrix} \boldsymbol{p}_d \rightarrow & \dot{\boldsymbol{k}}^* \\ & \uparrow \\ \boldsymbol{f} & \dot{\boldsymbol{l}}^*_d \end{matrix}$	$\begin{matrix} \dot{\boldsymbol{k}}_d \rightarrow \boldsymbol{p} \rightarrow \dot{\boldsymbol{k}}^* \\ \dot{\boldsymbol{l}}_d \dashrightarrow \boldsymbol{f} \dashrightarrow \dot{\boldsymbol{l}}^* \end{matrix}$

Hyon et al. [14] presented a method to resolve foot GRFs and foot CoPs such that they minimize the sum of the squared norm of forces at some points on the boundary of the foot sole while satisfying the desired net GRF and CoP.

This method can minimize each foot GRF if the contact points are well distributed over the foot-ground contact surface. Their passivity-based controller can remarkably adapt to unknown rough terrain and non-level ground [13].

In another important work on the control of external forces and torques at each individual foot, Fujimoto et al. [7–9] resolved foot GRFs and torques simultaneously using a quadratic programming method. In contrast, we resolve foot GRFs and foot CoPs sequentially, using two least-squares problems, each of which can be solved very quickly. Another notable difference between our work and that of [7–9] is that the latter computed external forces and torques from desired accelerations of CoM and trunk orientation whereas we compute them from desired linear and angular momenta rate changes. The advantage of computing desired trunk orientation from external forces and torques is that it can be done more intuitively than computing desired angular momentum, the latter having no direct visible reference. On the other hand, our approach is based on Newton's law, i.e., momentum rate change is completely determined by the external forces and torques. In contrast, the angular acceleration of the trunk cannot be completely determined from the external forces and torques unless the accelerations of other joints are also specified. If these joint accelerations are not incorporated in the force/torque computation, the computed values would be valid only for motions with negligible joint accelerations.

Abe et al. [2] represented foot GRFs and CoPs in a similar manner to [14]. Sentis et al. [37] developed a method to precisely control contact CoPs in a more general setting of multicontact interaction between a humanoid robot and the environment using a virtual-linkage model. Park et al. [31] showed that many balancing problems can be framed as the second-order cone programming problem.

Unlike the above-mentioned approaches which involve distributing the net GRF and net CoP to the supporting feet, Sugihara and Nakamura [42, 43] take a different approach that computes the desired acceleration of CoM from the desired foot GRFs and foot CoPs, and then resolves the joint motion to realize the desired acceleration of CoM.

This method has the merit of offering an easy manipulation of contact state between individual foot and the ground but, as mentioned in [42,43] by the authors themselves, is not guaranteed to realize the desired foot CoP and GRF during double support.

3 Momentum-based Balance Control Framework

This is the main section of the paper which provides step-by-step details of how the joint torques for the controller are determined.

3.1 Control Framework

We will represent the configuration of a humanoid robot as $Q = (\boldsymbol{T}_0, \boldsymbol{\theta}) \in \mathsf{SE}(3) \times \mathbb{R}^n$, where $\boldsymbol{T}_0 = (\boldsymbol{R}_0, \boldsymbol{p}_0) \in \mathsf{SO}(3) \times \mathbb{R}^3$ denotes the base frame (trunk) configuration, $\boldsymbol{\theta} \in \mathbb{R}^n$ is the vector of joint angles, and n is the total number of joint DoFs. The subscripts 0 and s denote the base frame and joints, respectively, with s implying "shape" associated with the joint angles in geometric dynamics [4]. The total DoFs of the robot is thus $6 + n$, because the floating base has 6 DoFs. The generalized velocity can be written as $\dot{\boldsymbol{q}} = (\mathbf{v}_0, \dot{\boldsymbol{\theta}}) \in \mathbb{R}^{6+n}$ where $\mathbf{v}_0 = (\boldsymbol{\omega}_0, \boldsymbol{v}_0)$ is the spatial velocity of the trunk with respect to the body frame and expressed as: [3]

$$[\boldsymbol{\omega}_0]_\times = \boldsymbol{R}_0^T \dot{\boldsymbol{R}}_0 \tag{3}$$

$$\boldsymbol{v}_0 = \boldsymbol{R}_0^T \dot{\boldsymbol{p}}_0 \tag{4}$$

Then, assuming stationary ground, the constraint equations due to ground contacts and the joint space equations of motion of the robot are as follows:

$$\mathbf{0} = \boldsymbol{J}(Q)\dot{\boldsymbol{q}} \tag{5}$$

$$\boldsymbol{\tau} = \boldsymbol{H}(Q)\ddot{\boldsymbol{q}} + \boldsymbol{C}(Q, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{\tau}_g(Q) - \boldsymbol{J}^T \boldsymbol{f}_c \tag{6}$$

where $\boldsymbol{\tau} \in \mathbb{R}^{6+n}$ denotes the generalized forces, $\boldsymbol{H}$ is the joint space inertia matrix, $\boldsymbol{C}\dot{\boldsymbol{q}}$ includes Coriolis and centrifugal terms and $\boldsymbol{\tau}_g$ is the gravity torque. $\boldsymbol{f}_c$ is a vector representing external "constraint" forces from the ground, determined by foot GRFs and CoPs, and the Jacobian $\boldsymbol{J} \in \mathbb{R}^{c\times(6+n)}$ transforms $\boldsymbol{f}_c$ to the generalized forces. The number of constraint equations c depends on the nature of constraint at the foot-ground contact. For example, when both the linear and angular motion of the support foot are constrained due to foot-ground contact, $c = 6$ for single support and $c = 12$ for double support. In this case, $\boldsymbol{J}\dot{\boldsymbol{q}}$ denotes the linear and angular velocities of the support foot given $\dot{\boldsymbol{q}}$, and $\mathbf{0}$ in (5) denotes zero velocity of the support foot.

Since the robot base is free floating, the first six elements of $\boldsymbol{\tau}$ are zero, i.e., $\boldsymbol{\tau}^T = [\mathbf{0}^T \; \boldsymbol{\tau}_s^T]$. Hence, we can divide (6) into two parts, one corresponding to the base, denoted by the subscript 0, and the other, subscripted with s, for the joints. Then (5) and (6) are rewritten as follows:

$$\mathbf{0} = \boldsymbol{J}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}} \tag{7}$$

$$\mathbf{0} = \boldsymbol{H}_0\ddot{\boldsymbol{q}} + \boldsymbol{C}_0\dot{\boldsymbol{q}} + \boldsymbol{\tau}_{g,0} - \boldsymbol{J}_0^T \boldsymbol{f}_c \tag{8}$$

$$\boldsymbol{\tau}_s = \boldsymbol{H}_s\ddot{\boldsymbol{q}} + \boldsymbol{C}_s\dot{\boldsymbol{q}} + \boldsymbol{\tau}_{g,s} - \boldsymbol{J}_s^T \boldsymbol{f}_c \tag{9}$$

where (7) is the time derivative of (5).

Due to the high DoFs of humanoid robots, balance controllers usually solve an optimization problem. However, the computational cost of the optimization increases rapidly as the dimension of the search space increases. Even the simplest optimization problem such

[3] $\dot{\boldsymbol{q}}$ is a slight abuse of notation because we do not define nor use a vector $\boldsymbol{q}$. However, since $\mathsf{se}(3)$, the Lie algebra of $\mathsf{SE}(3)$, is isomorphic to $\mathbb{R}^6$, we will use a single vector form of $\dot{\boldsymbol{q}} \in \mathbb{R}^{6+n}$ for convenience. $[\boldsymbol{\omega}_0]_\times$ represents a skew-symmetric matrix of a vector $\boldsymbol{\omega}_0$.

as the least-squares problem has order $O(n^3)$ time complexity. Therefore, aiming for computational efficiency, we have adopted a sequential approach; we divide the balance control problem into three smaller sub-problems, which can be solved serially. The balance controller determines the control input $\boldsymbol{\tau}_s$ through the following steps:

- **Step 1**: foot GRFs and foot CoPs (hence $\boldsymbol{f}_c$) are computed from the desired momentum rate change.
- **Step 2**: joint accelerations $\ddot{\boldsymbol{q}}$ are determined such that they satisfy both (7) and (8). Actually, as will be explained in Sec. 3.5, instead of directly using (8), we use the centroidal momentum equation (32), which is a slight variation of (8). In general, if the total number of robot DoFs is greater than or equal to $c + 6$, a solution to $\ddot{\boldsymbol{q}}$ exists.
- **Step 3**: the required joint torques $\boldsymbol{\tau}_s$ satisfying (9) are computed from $\boldsymbol{f}_c$ and $\ddot{\boldsymbol{q}}$ using an inverse dynamics algorithm.

Note that, by computing $\boldsymbol{f}_c$ and $\ddot{\boldsymbol{q}}$ first, we can use efficient linear-time algorithms for inverse dynamics in **Step 3**, without having to compute the joint space equations of motion (6) which have a quadratic time complexity.

3.2 Desired Momentum for Balance Control

The overall behavior of the robot against external perturbations is determined by the desired momentum rate change. We employ the following feedback control policy:

$$\dot{\boldsymbol{k}}_d = \boldsymbol{\Gamma}_{11}(\boldsymbol{k}_d - \boldsymbol{k}) \tag{10}$$

$$\dot{\boldsymbol{l}}_d/m = \boldsymbol{\Gamma}_{21}(\dot{\boldsymbol{r}}_{G,d} - \dot{\boldsymbol{r}}_G) + \boldsymbol{\Gamma}_{22}(\boldsymbol{r}_{G,d} - \boldsymbol{r}_G) \tag{11}$$

where $\dot{\boldsymbol{k}}_d$ and $\dot{\boldsymbol{l}}_d$ are the independently specified desired rates of change of centroidal angular and linear momenta. In other words $\dot{\boldsymbol{k}}_d$ and $\dot{\boldsymbol{l}}_d$ are not time derivatives of $\boldsymbol{k}_d$ and $\boldsymbol{l}_d$. Additionally, $\boldsymbol{r}_{G,d}$ is the desired CoM position. $\boldsymbol{\Gamma}_{ij}$ represents 3×3 diagonal matrix of feedback gain parameters. Note that unlike the linear position feedback term in (11), we do not have an angular position feedback in (10). This is because a physically meaningful angular "position" cannot be defined corresponding to angular momentum [47]. For postural balance maintenance experiments we set $\boldsymbol{k}_d$ and $\dot{\boldsymbol{r}}_{G,d}$ to zero and $\boldsymbol{r}_{G,d}$ to be above the mid-point of the geometric centers of the two feet. For other cases, these values may be determined from the desired motion.

It is to be noted that, despite the various studies on angular momentum in humanoid motions [1,3,10,11,16,18,20,24,26,29,34,36,45,48], the issue of how to set the desired angular momentum for more complex motions such as locomotion has not been fully explored, and remains an important future work.

3.3 Prioritization between Linear and Angular Momenta

Given the desired momentum rate change, we determine admissible foot GRF and foot CoP such that the resulting momentum rate change is as close as possible to the desired value. If the desired GRF and CoP computed from $\dot{\boldsymbol{k}}_d$ and $\dot{\boldsymbol{l}}_d$ violate physical constraints (e.g., GRF being outside friction cone, normal component of GRF being negative, or CoP being outside support base), it is not possible to generate those $\dot{\boldsymbol{k}}_d$ and $\dot{\boldsymbol{l}}_d$. In this case we must strike a compromise and decide which quantity out of $\dot{\boldsymbol{k}}_d$ and $\dot{\boldsymbol{l}}_d$ is more important to preserve.

Fig. 3 illustrates one case where the desired CoP, $\boldsymbol{p}_d$, computed from the desired momentum rate change is outside the support base, indicating that it is not admissible. Two different solutions are possible. The first solution, shown in Fig. 3, left, is to translate the CoP to the closest point of the support base while keeping the magnitude and line of action of the desired GRF $\boldsymbol{f}_d$ unchanged. In this case the desired linear momentum is attained but the desired angular momentum is compromised. The behavior emerging from this choice is characterized by a trunk rotation. This strategy can be observed in the human when the trunk yields in the direction of the push to maintain balance. Alternatively, in addition to translating the CoP to the support base, as before, we can rotate the direction of the GRF, as shown in Fig. 3, right. In this case the desired angular momentum is attained while the desired linear momentum is compromised. With this strategy the robot must move linearly along the direction of the applied force due to the residual linear momentum, making it necessary to step forward to prevent falling.

In this paper, we give higher priority to preserving linear momentum over angular momentum because it increases the capability of postural balance without involving a stepping. Ideally, a smart controller should be able to choose optimal strategies depending on the environment conditions and the status of the robot. The approaches that give higher priority to linear momentum and sacrifice angular momentum can also be found in the literature [11,41] and in our recent work on stepping [49].

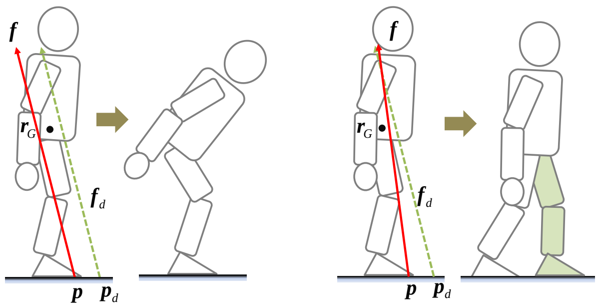


Fig. 3: When the desired GRF, $\boldsymbol{f}_d$ and the desired CoP, $\boldsymbol{p}_d$ computed from the desired momentum rate change are not simultaneously admissible, as indicated by $\boldsymbol{p}_d$ being outside the support base, momenta objectives need to be compromised for control law formulation. Two extreme cases are illustrated. Left: linear momentum is respected while angular momentum is compromised. Right: angular momentum is respected while linear momentum is compromised.

3.4 Admissible Foot GRF, Foot CoP, and Momentum Rate Change

Given the desired momentum rate change, we determine admissible foot GRF and CoP such that the resulting momentum rate change is as close as possible to the desired value. Admissible momentum rate change is determined by the admissible foot GRF and foot CoP.

3.4.1 Single Support Case

Dealing with single support case is straightforward because the foot GRF and CoP are uniquely determined from the desired momentum rate change, from (1) and (2):

$$\boldsymbol{f}_d = \dot{\boldsymbol{l}}_d - m\boldsymbol{g} \tag{12}$$

$$p_{d,X} = r_{G,X} - \frac{1}{\dot{l}_{d,Y} - mg}(f_{d,X}\, r_{G,Y} - \dot{k}_{d,Z}) \tag{13}$$

$$p_{d,Z} = r_{G,Z} - \frac{1}{\dot{l}_{d,Y} - mg}(f_{d,Z}\, r_{G,Y} + \dot{k}_{d,X}) \tag{14}$$

where the Y-axis is parallel to the direction of gravity vector, i.e., $\boldsymbol{g} = (0, g, 0)$.

If $\boldsymbol{f}_d$ and $\boldsymbol{p}_d$ computed above are valid, then we directly use these values. Otherwise, as mentioned previously, we give higher priority to linear momentum. If $\boldsymbol{f}_d$ is outside the friction cone, we project it onto the friction cone to prevent foot slipping.

3.4.2 Double Support Case

Determining foot GRFs and foot CoPs for double support is more involved. Let us first rewrite (1) and (2) for the double support case. Following [36], we will express the GRF at each foot in terms of the forces and torques *applied to the corresponding ankle* (Fig. 4). The benefit of this representation is that we can explicitly express the torques applied to the ankles.

$$\dot{\boldsymbol{k}} = \dot{\boldsymbol{k}}_f + \dot{\boldsymbol{k}}_\tau \tag{15}$$

$$\dot{\boldsymbol{k}}_f = (\boldsymbol{r}_r - \boldsymbol{r}_G) \times \boldsymbol{f}_r + (\boldsymbol{r}_l - \boldsymbol{r}_G) \times \boldsymbol{f}_l \tag{16}$$

$$\dot{\boldsymbol{k}}_\tau = \boldsymbol{\tau}_r + \boldsymbol{\tau}_l \tag{17}$$

$$\dot{\boldsymbol{l}} = m\boldsymbol{g} + \boldsymbol{f}_r + \boldsymbol{f}_l \tag{18}$$

In (15), we have divided $\dot{\boldsymbol{k}}$ into two parts, $\dot{\boldsymbol{k}}_f$, due to the ankle force, and $\dot{\boldsymbol{k}}_\tau$, due to ankle torque. This division enables us to take ankle torques into account in determining foot GRFs. $\boldsymbol{f}_r$ and $\boldsymbol{f}_l$ are the GRFs at the right and left foot, respectively, and $\boldsymbol{r}_r$, $\boldsymbol{r}_l$ are the positions of the body frames of the foot, located at the respective ankle joints.

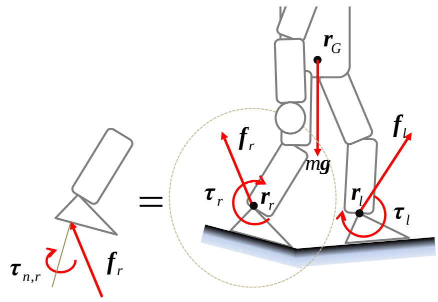


Fig. 4: By expressing GRF applied to each foot with respect to the local frame of the foot located at the ankle, we can factor out the moments $\boldsymbol{\tau}_r$, $\boldsymbol{\tau}_l$ applied to the ankle by the foot GRFs $\boldsymbol{f}_r$ and $\boldsymbol{f}_l$. $\boldsymbol{r}_r$ and $\boldsymbol{r}_l$ are the positions of the ankles.

The ankle torques $\boldsymbol{\tau}_i$, $(i = r, l)$ are expressed in terms of foot GRF and foot CoP as follows (Fig. 5):

$$\boldsymbol{\tau}_i = (\boldsymbol{R}_i \boldsymbol{d}_i) \times \boldsymbol{f}_i + \boldsymbol{R}_i \boldsymbol{\tau}_{n,i}\,, \tag{19}$$

where $\boldsymbol{R}_i$ is the orientation of the foot, $\boldsymbol{d}_i$ is the foot CoP in body frame, and $\boldsymbol{\tau}_{n,i} = (0, 0, \tau_{n,i})$ is the normal torque in body frame.

Given $\dot{\boldsymbol{k}}$ and $\dot{\boldsymbol{l}}$, solving for foot GRFs and foot CoPs is an underdetermined problem, which lets us prescribe additional optimality criteria to find a solution. If we incorporate minimal ankle torques into the optimality condition, we could express the objective function as follows:

$$w_l||\dot{\boldsymbol{l}}_d - \dot{\boldsymbol{l}}(\boldsymbol{f}_r, \boldsymbol{f}_l)||^2 + w_k||\dot{\boldsymbol{k}}_d - \dot{\boldsymbol{k}}(\boldsymbol{f}_r, \boldsymbol{f}_l, \boldsymbol{\tau}_r, \boldsymbol{\tau}_l)||^2$$
$$+ w_f\left(||\boldsymbol{f}_r||^2 + ||\boldsymbol{f}_l||^2\right) + w_\tau\left(||\boldsymbol{\tau}_r||^2 + ||\boldsymbol{\tau}_l||^2\right)$$
$$\text{s.t. } \boldsymbol{f}_i \text{ and } \boldsymbol{\tau}_i \text{ are admissible}$$

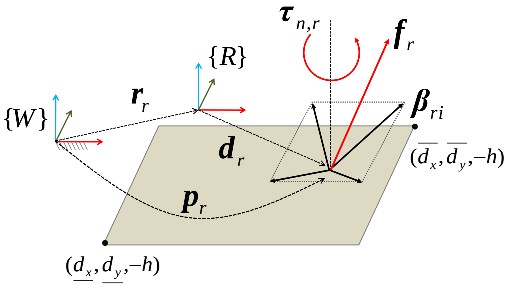


Fig. 5: We represent the foot/ground interaction forces on the right foot using foot CoP, whose location with respect to the right foot frame $\{R\}$ is denoted by $\boldsymbol{d}_r = (\boldsymbol{d}_{r,X}, \boldsymbol{d}_{r,Y}, -h)$, ground reaction moment normal to the ground $\boldsymbol{\tau}_{n,r} = (0, 0, \tau_{n,r})$, and the GRF $\boldsymbol{f}_r$. $\boldsymbol{f}_r$ is represented using four basis vectors $\boldsymbol{\beta}_{rj}$ $(j = 1 \ldots 4)$ that approximate the friction cone of the ground, i.e., $\boldsymbol{f}_r = \sum_j \boldsymbol{\beta}_{rj}\rho_{rj}$, where ρ_{rj} (≥ 0) is the magnitude in the direction of $\boldsymbol{\beta}_{rj}$. Therefore, the ground pressure is defined by 7 parameters, $(\rho_{r1}, \ldots, \rho_{r4}, d_{r,X}, d_{r,Y}, \tau_{n,r})$. This representation is compact, having only one more parameter than the minimum (3 for force and 3 for torque), and constraint can be expressed in a very simple form for a rectangular convex hull of the foot sole, i.e., $\rho_j \geq 0$, $\underline{d_j} \leq d_j \leq \overline{d_j}$, and $|\tau_n| < \mu_\tau f_{r,N}$ where $f_{r,N}$ is the normal component of $\boldsymbol{f}_r$, i.e., the Z-coordinate of $\boldsymbol{R}_r^T \boldsymbol{f}_r$ with $\boldsymbol{R}_r$ being the orientation of the right foot. μ_τ is a friction coefficient for torque and h is the height of foot frame from the foot sole. Note that $\boldsymbol{d}_r$ and $\boldsymbol{\tau}_{n,r}$ are expressed with respect to the body frame $\{R\}$ while $\boldsymbol{r}_r$, $\boldsymbol{p}_r$, $\boldsymbol{f}_r$, and $\boldsymbol{\beta}_{rj}$ are with respect to the world frame.

(20)

where the first two terms aim to achieve the desired momentum rate change, the third term regularizes foot GRFs, and the last term tries to minimize ankle torques. w's are weighting factors among the different objectives.

Eq. (20) represents a nonlinear problem and it especially contains nonlinear cross product terms; this makes it difficult to use in a real-time controller. Additionally, (due to ???) the form of (20) is not quadratic. ?? Comments about convexity? One solution is to convert this general nonlinear optimization problem to easier ones that can be solved using least-squares or quadratic programming methods. This can be achieved by expressing the foot GRF and foot CoP using the forces at certain specific locations on the boundary of the foot soles [14,33]. However, this approach increases the dimension of the search space significantly. For example, [33] used 16 variables to model the GRF and CoP of one foot, which is 10 more than the dimension of the unknowns.

We develop a different approach. Instead of increasing the search space to make the optimization problem easier, we approximate (20) with two constrained least-squares problems, one for determining the foot GRFs, and the other for determining the foot CoPs. This way the number of variables is kept small. Additionally, we attempt to minimize the ankle torques. Minimizing ankle torques is meaningful because large ankle torques can cause foot slipping.

Our approach can be intuitively understood as follows. In order to minimize the ankle torques ($\dot{\boldsymbol{k}}_\tau \to \boldsymbol{0}$), the foot GRFs $\boldsymbol{f}_r$ and $\boldsymbol{f}_l$ should create $\dot{\boldsymbol{k}}_f$ as close to the desired angular momentum rate change ($\dot{\boldsymbol{k}}_f \to \dot{\boldsymbol{k}}_d$) as possible while satisfying $\dot{\boldsymbol{l}}_d$. If $\dot{\boldsymbol{k}}_f = \dot{\boldsymbol{k}}_d$, the ankle torques can vanish. If $\dot{\boldsymbol{k}}_f \neq \dot{\boldsymbol{k}}_d$, we compute the ankle torques that are necessary to generate the residual angular momentum rate change, $\dot{\boldsymbol{k}}_d - \dot{\boldsymbol{k}}_f$. In other words, by reducing burdens on the ankle torques to create $\dot{\boldsymbol{k}}_d$, our approach can be understood as solving (20) for the case in which minimizing ankle torques has higher priority than regularizing foot GRFs.

Determination of Foot GRFs

In order to compute the foot GRFs, $\boldsymbol{f}_r$ and $\boldsymbol{f}_l$, we solve the optimization problem below:

$$\min||\dot{\boldsymbol{l}}_d - \dot{\boldsymbol{l}}(\boldsymbol{f}_r, \boldsymbol{f}_l)||^2 + w_k||\dot{\boldsymbol{k}}_d - \dot{\boldsymbol{k}}_f(\boldsymbol{f}_r, \boldsymbol{f}_l)||^2 + \epsilon_f(||\boldsymbol{f}_r||^2 + ||\boldsymbol{f}_l||^2)\,, \tag{21}$$

where w_k and ϵ_f $(w_k \gg \epsilon_f > 0)$ are weighting factors for angular momentum and the regularization of foot GRFs, respectively. Note that, if $\dot{\boldsymbol{k}}_d = \dot{\boldsymbol{k}}_f$, the ankle torques $\boldsymbol{\tau}_i$ become zero. Each foot GRF is modeled using four basis vectors $\boldsymbol{\beta}_{ij}$ and their magnitudes ρ_{ij} that approximate the friction cone (an inverted pyramid in Fig. 5) on the ground

$$\boldsymbol{f}_i = \sum_{j=1}^{4} \boldsymbol{\beta}_{ij}\rho_{ij} := \boldsymbol{\beta}_i\boldsymbol{\rho}_i\,, \tag{22}$$

where $\boldsymbol{\beta}_i = [\boldsymbol{\beta}_{i1} \ \cdots \ \boldsymbol{\beta}_{i4}]$.

Note that $\boldsymbol{r}_r$ and $\boldsymbol{r}_l$ are determined by the configuration of the robot; they are constants when solving this problem. Therefore $\dot{\boldsymbol{k}}_f$ becomes a linear equation of $\boldsymbol{\rho}_i$ when we substitute (22) into (16). Rearranging into a matrix equation, we can turn the optimization problem (21) into a linear least-squares problem with non-negativity constraints where the only unknowns are the $\boldsymbol{\rho}_i$:

$$\min||\boldsymbol{\Phi}\boldsymbol{\rho} - \boldsymbol{\xi}||^2 \text{ s.t. } \boldsymbol{\rho}_i \geq 0\,, \tag{23}$$

where [4]

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\beta}_r & \boldsymbol{\beta}_l \\ w_f \boldsymbol{\delta}_r & w_f \boldsymbol{\delta}_l \\ \multicolumn{2}{c}{\epsilon_f \mathbf{1}} \end{bmatrix} \in \mathbb{R}^{(3+3+8)\times(4+4)} \tag{24}$$

$$\boldsymbol{\xi} = \begin{bmatrix} \dot{\boldsymbol{l}}_d - m\boldsymbol{g} \\ w_f \dot{\boldsymbol{k}}_d \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{(3+3+8)}$$

$$\boldsymbol{\rho} = \left[\boldsymbol{\rho}_r^T \ \boldsymbol{\rho}_l^T\right]^T \in \mathbb{R}^8 \tag{25}$$

$$\boldsymbol{\delta}_i = [\boldsymbol{r}_i - \boldsymbol{r}_G]_\times \boldsymbol{\beta}_i \tag{26}$$

Determination of Foot CoPs

In general, the desired angular momentum rate change cannot be fully generated only by $\boldsymbol{f}_r$ and $\boldsymbol{f}_l$, so the residual, $\dot{\boldsymbol{k}}_{\tau,d} = \dot{\boldsymbol{k}}_d - \dot{\boldsymbol{k}}_f$, should be generated by the ankle torques. To this end, we determine the location of each foot CoP such that they create $\dot{\boldsymbol{k}}_{\tau,d}$ while minimizing each ankle torque. It is to be noted that, after having determined $\boldsymbol{f}_i$, (19) can be written as a linear function of $\boldsymbol{d}_i$ and $\boldsymbol{\tau}_{n,i}$:

$$\boldsymbol{\tau}_i = -[\boldsymbol{f}_i]_\times \boldsymbol{R}_i \boldsymbol{d}_i + \boldsymbol{R}_i \boldsymbol{\tau}_{n,i}\,, \tag{27}$$

so that we can express the optimization problem as a least-squares problem with upper and lower bounds:

$$\min ||\boldsymbol{\Psi}\boldsymbol{\eta} - \boldsymbol{\kappa}||^2 \text{ s.t. } \underline{\boldsymbol{\eta}} \le \boldsymbol{\eta} \le \overline{\boldsymbol{\eta}}\,, \tag{28}$$

where

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\Psi}_k \\ \epsilon_p \mathbf{1} \end{bmatrix} \in \mathbb{R}^{(3+6)\times 6},\ \boldsymbol{\kappa} = \begin{bmatrix} \boldsymbol{\kappa}_k \\ \epsilon_p \boldsymbol{\eta}_d \end{bmatrix} \in \mathbb{R}^{(3+6)} \tag{29}$$

$$\boldsymbol{\eta} = [d_{r,X}\ d_{r,Y}\ \tau_{n,r}\ d_{l,X}\ d_{l,Y}\ \tau_{n,l}]^T \in \mathbb{R}^6\,, \tag{30}$$

where the elements of the constant matrix $\boldsymbol{\Psi}_k \in \mathbb{R}^{3\times 6}$ and $\boldsymbol{\kappa}_k$ are determined from (27).[5]

$\underline{\boldsymbol{\eta}}$ and $\overline{\boldsymbol{\eta}}$ are determined from foot geometry, friction coefficient, and the normal component of foot GRF (see Fig. 5). $\boldsymbol{\eta}_d$ is chosen such that $\boldsymbol{\tau}_i$ is zero, i.e., the line of action of $\boldsymbol{f}_i$ intersects the ankle. Note that both the least-squares problems have a small number of variables, so the optimization can be carried out quickly.

[4] The vector $\boldsymbol{\delta}_i$ expresses angular momentum rate change (16) in terms of $\boldsymbol{\rho}_i$ as follows:

$$(\boldsymbol{r}_r - \boldsymbol{r}_G) \times \boldsymbol{f}_r = (\boldsymbol{r}_r - \boldsymbol{r}_G) \times (\boldsymbol{\beta}_r \boldsymbol{\rho}_r) = \underbrace{[\boldsymbol{r}_i - \boldsymbol{r}_G]_\times \boldsymbol{\beta}_r}_{\boldsymbol{\delta}_r} \boldsymbol{\rho}_r := \boldsymbol{\delta}_r \boldsymbol{\rho}_r$$

[5] Specifically, $\boldsymbol{\Psi}_k = [\boldsymbol{\Psi}_k^0 \ldots \boldsymbol{\Psi}_k^5]$ where

$$\boldsymbol{\Psi}_k^0 = -\boldsymbol{R}_r^1 \boldsymbol{f}_{r,Z}^b + \boldsymbol{R}_r^2 \boldsymbol{f}_{r,Y}^b, \quad \boldsymbol{\Psi}_k^1 = \boldsymbol{R}_r^0 \boldsymbol{f}_{r,Z}^b - \boldsymbol{R}_r^2 \boldsymbol{f}_{r,X}^b, \quad \boldsymbol{\Psi}_k^2 = \boldsymbol{R}_r^2,$$
$$\boldsymbol{\Psi}_k^3 = -\boldsymbol{R}_l^1 \boldsymbol{f}_{l,Z}^b + \boldsymbol{R}_l^2 \boldsymbol{f}_{l,Y}^b, \quad \boldsymbol{\Psi}_k^4 = \boldsymbol{R}_l^0 \boldsymbol{f}_{l,Z}^b - \boldsymbol{R}_l^2 \boldsymbol{f}_{l,X}^b, \quad \boldsymbol{\Psi}_k^5 = \boldsymbol{R}_l^2,$$

and $\boldsymbol{\kappa}_k = \dot{\boldsymbol{k}}_{\tau,d} + h(\boldsymbol{R}_r^1 \boldsymbol{f}_{r,X}^b - \boldsymbol{R}_r^0 \boldsymbol{f}_{r,Y}^b + \boldsymbol{R}_l^1 \boldsymbol{f}_{l,X}^b - \boldsymbol{R}_l^0 \boldsymbol{f}_{l,Y}^b)$. $\boldsymbol{R}_i^j$ is j-th column vector of $\boldsymbol{R}_i$ $(i = r, l)$, $\boldsymbol{f}_i^b = \boldsymbol{R}_i^T \boldsymbol{f}_i$, and h is the height of foot frame from the foot sole.

Admissible Momentum Rate Change

After determining admissible foot GRF and foot CoP, the admissible momentum rate change $\dot{\boldsymbol{h}}_a = [\dot{\boldsymbol{k}}_a^T\ \dot{\boldsymbol{l}}_a^T]^T$ is also computed using (1) and (2) for single support, or (15) and (18) for double support.

3.5 Determination of Joint Accelerations and Torques

After determining the admissible foot GRFs and foot CoPs, and admissible momentum rate changes, we compute joint accelerations and torques to realize them. In this step, we adopt a procedure similar to that used in [24].

First, we resolve the desired joint accelerations $\ddot{\boldsymbol{q}}$ for balance such that they satisfy (7) and a variation of (8). To explain the latter let us first express the spatial centroidal momentum $\boldsymbol{h} = [\boldsymbol{k}^T\ \boldsymbol{l}^T]^T$ in terms of the generalized velocities:

$$\boldsymbol{h} = \boldsymbol{A}(Q)\dot{\boldsymbol{q}}\,, \tag{31}$$

where $\boldsymbol{A} \in \mathbb{R}^{6\times(6+n)}$ is the centroidal momentum matrix [30] that linearly maps the generalized velocities to the spatial momentum. Differentiating (31), we obtain

$$\dot{\boldsymbol{h}} = \boldsymbol{A}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{A}}\dot{\boldsymbol{q}}\,. \tag{32}$$

If we replace $\dot{\boldsymbol{h}}$ with external forces using Newton's law (refer to (1) and (2)), then (32) expresses the aggregate motion of the dynamic system due to the external forces, which is exactly same as what (8) represents. Note that the joint torques are not included in (8). The only difference is the reference frame: (32) is expressed with respect to a frame at the CoM whereas (8) is written with respect to the base frame. While either (8) or (32) can be used, we choose to use (32) because our balance controller defines its objectives in terms of centroidal momenta.

Specifically, we compute the output accelerations of the balance controller $\ddot{\boldsymbol{\theta}}$ such that they minimize the following objective function:

$$\begin{aligned} & w_b||\dot{\boldsymbol{h}}_a - \boldsymbol{A}\ddot{\boldsymbol{q}} - \dot{\boldsymbol{A}}\dot{\boldsymbol{q}}|| + (1-w_b)||\ddot{\boldsymbol{\theta}}_d^u - \ddot{\boldsymbol{\theta}}^u|| \\ & \quad \text{s.t. } \boldsymbol{J}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}} = \boldsymbol{a}_d \quad \text{and} \quad \ddot{\boldsymbol{\theta}}_l \le \ddot{\boldsymbol{\theta}} \le \ddot{\boldsymbol{\theta}}_u\,, \end{aligned} \tag{33}$$

where $\dot{\boldsymbol{h}}_a$ is the admissible momentum rate change. The output acceleration vector which is the solution to (33) is called the admissible output acceleration and termed as $\ddot{\boldsymbol{\theta}}_a$. Note that $\ddot{\boldsymbol{\theta}}_a$ contains all the joint accelerations except those of the floating joints.

Note that, because there can be infinitely many solutions for $\ddot{\boldsymbol{\theta}}_a$ that create $\dot{\boldsymbol{h}}_a$, we have an additional optimality criteria in (33), which is to follow the desired joint acceleration of the upper body $\ddot{\boldsymbol{\theta}}_d^u$ as closely as possible. One can set $\ddot{\boldsymbol{\theta}}_d^u$ to specify an upper-body task, or set $\ddot{\boldsymbol{\theta}}_d^u = \mathbf{0}$ to minimize the movement. The parameter $w_b (0 < w_b < 1)$ controls the relative importance between the balance objective (the first term) and the prescribed motion objective associated with the kinematic task (the second term). It is to be noted that w_b should be close to 1 in order to create admissible momentum rate, but it cannot be exactly 1 because in this case (33) becomes indeterminate. $\boldsymbol{a}_d = [\boldsymbol{a}_{d,r}^T \ \boldsymbol{a}_{d,l}^T]^T$ is the desired accelerations of the right and left feet. Section 3.6 details how to determine $\boldsymbol{a}_d$. Equation (33) can be easily converted to a least-squares problem with linear equality and bound constraints, and many solvers (e.g., [23]) are available for this type of problem.

We set $\ddot{\boldsymbol{\theta}}_l$ and $\ddot{\boldsymbol{\theta}}_u$, the lower and upper bound of the joint accelerations, somewhat heuristically such that the joint limit constraints are satisfied (e.g., $\ddot{\boldsymbol{\theta}}_u$ decreases when a joint angle approaches its upper limit).

Finally, we compute the feedforward torque input $\boldsymbol{\tau}_{ff}$ from $\ddot{\boldsymbol{\theta}}_a$ and the admissible external forces by performing inverse dynamics. Since external forces are explicitly specified for the support feet by (23) and (28) and joint accelerations are set by (33), we have all the necessary information for inverse dynamics. Specifically, we use the hybrid system dynamics algorithms [6], which is useful for performing inverse dynamics for floating-base mechanisms.

Overall torque input is determined by adding feedback terms:

$$\boldsymbol{\tau}_s = \boldsymbol{\tau}_{ff} + \boldsymbol{\tau}_{fb} \tag{34}$$

$$\boldsymbol{\tau}_{fb} = \boldsymbol{\Gamma}_p(\boldsymbol{\theta}^* - \boldsymbol{\theta}) + \boldsymbol{\Gamma}_d(\dot{\boldsymbol{\theta}}^* - \dot{\boldsymbol{\theta}}) \,, \tag{35}$$

where $\boldsymbol{\Gamma}_p = \mathrm{diag}(\gamma_{p,i})$ and $\boldsymbol{\Gamma}_d = \mathrm{diag}(\gamma_{d,i})$ are proportional and derivative gains, respectively. Position and velocity commands $\boldsymbol{\theta}^*$, $\dot{\boldsymbol{\theta}}^*$ are determined from the time integration of $\ddot{\boldsymbol{\theta}}_a$.

3.6 Desired Motion of the Feet

We set the desired foot accelerations $\boldsymbol{a}_d$ such that each foot has the desired configuration $\boldsymbol{T}_d \in \mathsf{SE}(3)$ and velocity $\mathbf{v}_d \in \mathsf{se}(3)$. Specifically, for each foot, we use the following feedback rule:

$$\boldsymbol{a}_{d,i} = k_p \log(\boldsymbol{T}_i^{-1}\boldsymbol{T}_{d,i}) + k_d(\mathbf{v}_{d,i} - \mathbf{v}_i) \,, \tag{36}$$

for $i \in \{r, l\}$ where k_p and k_d are proportional and derivative feedback gains, respectively. The $\log : \mathsf{SE}(3) \rightarrow \mathsf{se}(3)$ function computes the twist coordinates corresponding to a transformation matrix [28]. The current configuration $\boldsymbol{T}$ and velocity $\mathbf{v}$ of a foot can be computed from the forward kinematics operation assuming that the robot can either measure or estimate the joint angles and velocities as well as the configuration and velocity of the trunk, e.g., from an accelerometer and a gyroscope.

3.7 Controller Block Diagram

Fig. 6 shows a detailed block diagram of our balance controller. Inputs to the controller are the desired configuration and velocity of the feet $(\boldsymbol{T}_d, \mathbf{v}_d)$, CoM position and velocity $(\boldsymbol{r}_{G,d}, \dot{\boldsymbol{r}}_{G,d})$, angular momentum $(\boldsymbol{k}_d)$, and the upper body joint motion $(\ddot{\boldsymbol{\theta}}_d^u)$. Thus the balance control framework allows for the incorporation of specific motions of the head, arm and swing foot to perform some given tasks.

Using the sensory data of joint angles and trunk velocity, the kinematics solver computes CoM and its velocity, angular momentum, and the configuration and velocity of the feet.

4 Simulation Results

We tested the balance controller by simulating a full-sized humanoid robot (Fig. 7) using our high fidelity simulator called Locomote. Locomote is based on the commercial mobile robot simulation software package called Webots [25], which,in turn, uses a popular and free dynamics package called Open Dynamics Engine (ODE). The total mass of the robot is about 50 kg and each leg has 6 DoFs. The control command was generated every 1 msec. In the examples of this paper, we excluded the robot arms from the balance controller because arms may have other tasks to carry out simultaneously. Also, the arms are relatively light-weight and do not affect the state of balance all that much.

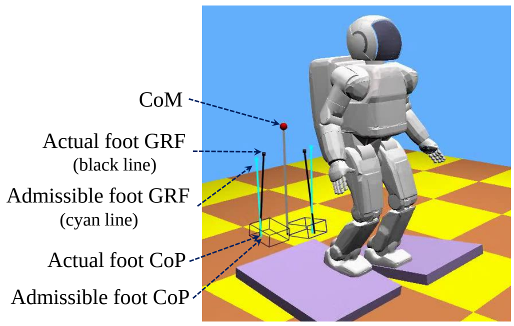


Fig. 7: The balance controller can deal with different non-level ground slopes at each foot. The cyan lines show the admissible GRF and CoP at each support foot determined by the balance controller, and the black lines show the actual GRF and CoP measured during the simulation.

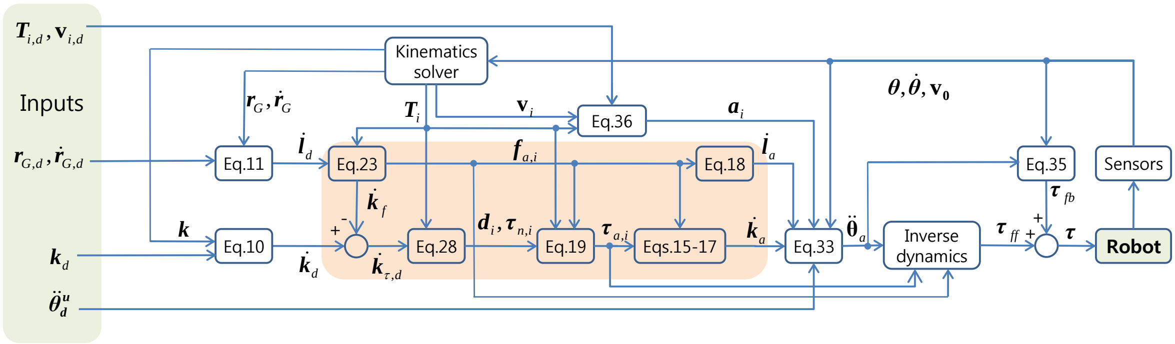


Fig. 6: Controller Block Diagram. Subscripts *'d'* and *'a'* refer to desired and admissible values, respectively. Subscript *'i'* refers to the left and right foot. Note that the blocks in the shaded area in the middle of the figure applies to double support case. For single support, the admissible foot GRF, CoP, and momenta rate changes are determined as described in Sec. 3.3.

4.1 Push Recovery on Stationary Support

In the first set of experiments the robot is subjected to pushes from various directions while it is standing on a stationary support. The directions, magnitudes, and the locations of the push are all unknown to the controller. As shown in Fig. 6, we assume that only the joint angles, joint velocities, and trunk velocity can be either measured or calculated using sensor data.

When the push magnitude is small, the desired GRF and CoP computed from the desired momentum rate change are both admissible, and thus the robot can achieve the desired values for both linear and angular momentum. When the perturbation is larger, the desired values are different from the admissible values, and in order to maintain balance without stepping, the controller tries to preserve the CoM location by modulating angular momentum by rotating the upper body. The resulting motion of the robot is similar to that of a human rotating the trunk in the direction of the push to maintain balance.

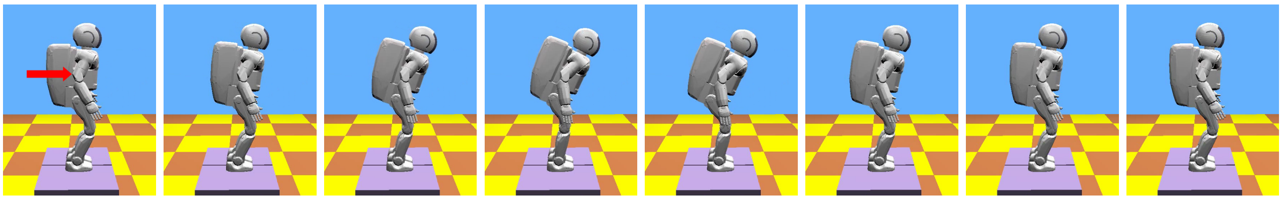
The top row of Fig. 8 shows a series of snapshots illustrating this when the robot is subjected to an external push (120 N, 0.1 sec) applied at the CoM in the forward direction. Before 0.2 sec and after 0.65 sec in the test, the admissible foot GRF and foot CoP can be determined such that they create the desired momentum rate change, so the admissible momentum rate change during that time period is nearly identical to its desired value (Figs. 8 (c, f)). However, from 0.2 to 0.65 sec, the admissible foot CoP (Fig. 8 (g)) is kept on the front border of the safe region of the support, marked with dotted line. Our controller gives higher priority to linear momentum so the admissible linear momentum rate change is still same as the desired value, while the angular momentum objective is compromised, as shown by the difference between the desired and admissible values of angular momentum rate change in Fig. 8 (f).

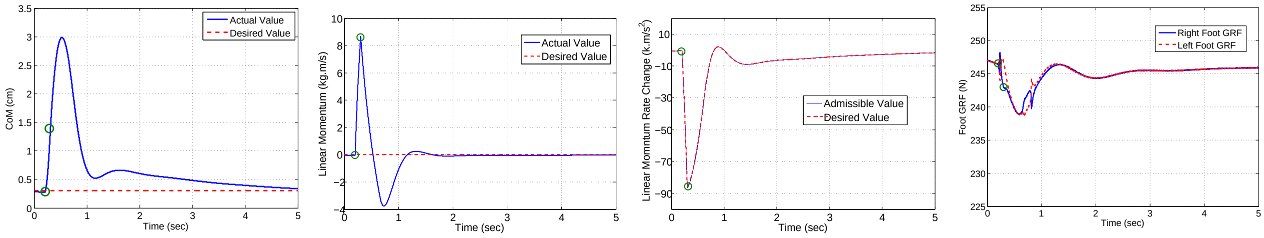
Fig. 8 (d) shows foot GRFs in vertical direction. The right and left foot GRFs have similar values and they smoothly return to the stable values after perturbation. Foot GRFs in forward direction have the same pattern with the linear momentum rate change (Fig. 8 (b)) because there exist no other external forces in forward direction.

Fig. 8 (g) shows the measured foot CoP, which is calculated using the contact force information during the simulation. Ideally it should be the same as the admissible foot CoP, but actually they are slightly different because of the inclusion of the prescribed motion objective in (33) as well as the numerical error of the simulation. Fig. 8 (h) shows the joint torque at the right ankle and, naturally, its trajectory has the same pattern with that of the foot CoP.

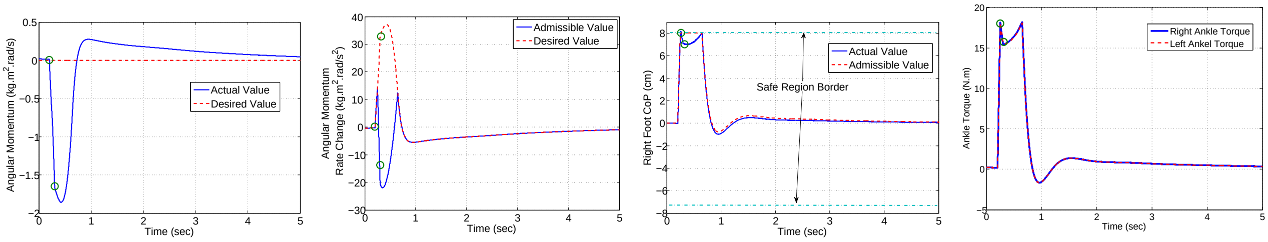
Fig. 9 shows the balance control behavior when the single-supported robot is pushed laterally. In this case the robot maintains balance by rotating the trunk in the coronal plane. Although compared to double support, the range of admissible CoP location is smaller during single-support, it is possible to create larger angular momentum through swing leg movement.

The trajectories of CoM, foot CoP, foot GRF, momentum, and ankle torques of this experiment are shown in Figs. 9 (a-h). These trajectories exhibit patterns very similar to the case when the double-support robot is pushed forward (Fig. 8). A notable difference is that for lateral push it takes about twice the time to get back to the desired pose than the forward push, as can be seen by the trajectory of CoM in Fig. 9 (a) compared with Fig. 8 (a), because the robot rotates more. The reason that the single-support robot rotates more in lateral direction is because the safe region of the foot support base of our robot model is narrower in lateral direction than frontal direction, thus the robot needs more angular momentum in lateral direction to keep the foot CoP





(a) CoM (forward direction) (b) Linear Momentum (forward direction) (c) Linear Momentum Rate Change (forward direction) (d) Foot GRFs (vertical direction)



(e) Angular Momentum (sagittal plane) (f) Angular Momentum Rate Change (sagittal plane) (g) Right Foot CoP (forward direction) (h) Ankle Torques (sagittal plane)

Fig. 8: Top row: Given a forward push (120 N, 0.1 sec), the balance controller controls both linear and angular momentum, and generates a motion comparable to human's balance control behavior. The robot is standing on stationary level platforms. (a-h): Trajectory of important physical properties of the experiment. Small circles in each figure indicate the start and end of the push. The dotted line in (g) indicates the front and rear borders of the safe region of the support, which is set to a few millimeters inside of the edge of the foot base. Left foot CoP is very similar to the right foot CoP.

inside the safe region. The slope in Fig. 9 (d) before the external perturbation is due to the planned movement of the robot in vertical direction.

4.2 Postural Balance on Moving Support

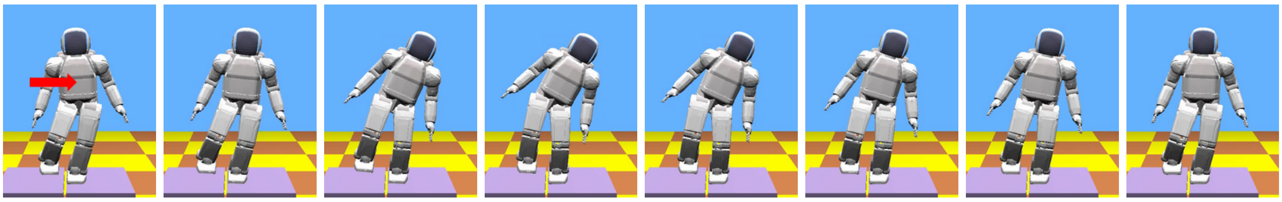
The second experiment is balance maintenance on non-level and non-stationary supports as shown in Fig. 10. In this case the two feet of the robot are supported on two surfaces of different inclination angles (+10 degrees and -10 degrees) and they receive continuous independent perturbations. In Fig. 10 (top row), both supports are moving in synchrony back and forth in a sinusoid pattern. With the amplitude of 1 m, the robot can endure the frequency up to about 0.12 Hz. The robot needs to generate fairly large trunk rotation to keep balance.

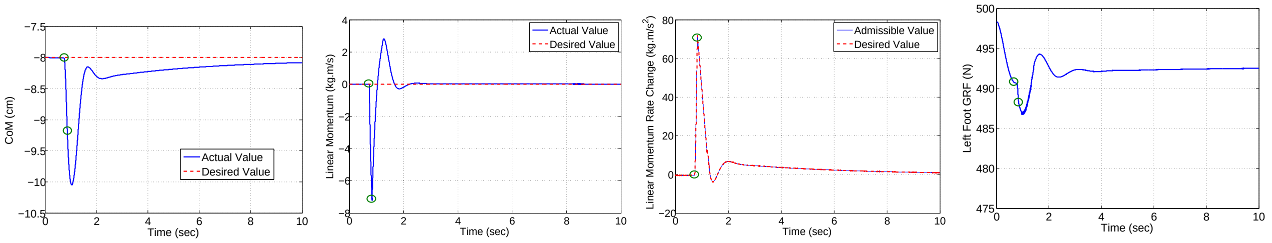
In Fig. 10 (bottom row) the two foot supports not only have different inclination angles (± 10 degrees) but are translating back and forth with out of phase velocities: when one support moves forward, the other moves backward. With a 0.4 m translation amplitude of the support, the robot can maintain balance up to about 1 Hz of frequency.

When a foot rests on a moving support, we need to estimate the motion of the support to set the desired motion of the foot properly. We use the following rule: if the measured CoP is inside the safe region of the support foot, we determine that the foot is not tipping but stably resting on the moving support. In this case, we update the desired configuration and velocity of the support foot to its current configuration and velocity, i.e., $\mathbf{v}_d = \mathbf{v}$ and $\boldsymbol{T}_d = \boldsymbol{T}$.

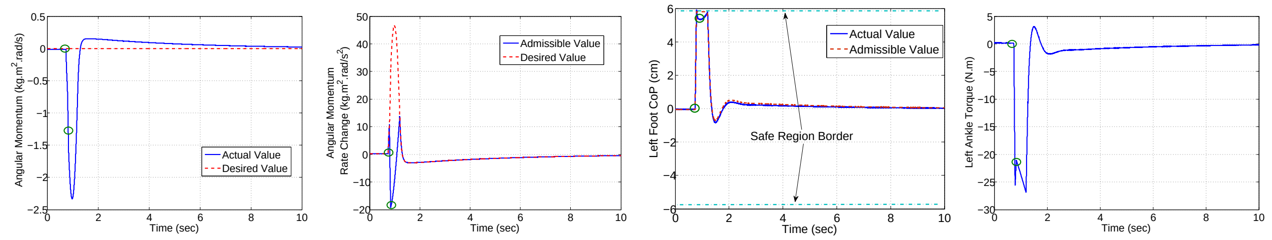
The desired horizontal location of CoM is set to the middle of the geometric centers of the two feet, and the desired velocity of CoM is set to the mean velocity of the two feet.

In all the experiments above, the following parameters are used: $\boldsymbol{\Gamma}_{11} = \text{diag}\{5{,}5{,}5\}$ in (10), $\boldsymbol{\Gamma}_{21} = \text{diag}\{40{,}20{,}40\}$ and $\boldsymbol{\Gamma}_{22} = \text{diag}\{8{,}3{,}8\}$ in (11), $w_k = 0.1$, $\epsilon_f = 0.01$ in (21), and $\epsilon_p = 0.01$ in (29).





(a) CoM (lateral direction) (b) Linear Momentum (lateral direction) (c) Linear Momentum Rate Change (lateral direction) (d) Left Foot GRF (vertical direction)



(e) Angular Momentum (coronal plane) (f) Angular Momentum Rate Change (coronal plane) (g) Left Foot CoP (lateral direction) (h) Left Ankle Torque (coronal plane)

Fig. 9: Top row: A leftward push (100 N, 0.1 sec) is applied to the single-supported robot on stationary level support. (a-h): Trajectory of important physical properties of the experiment. Small circles in each figure indicate the start and end of the push.

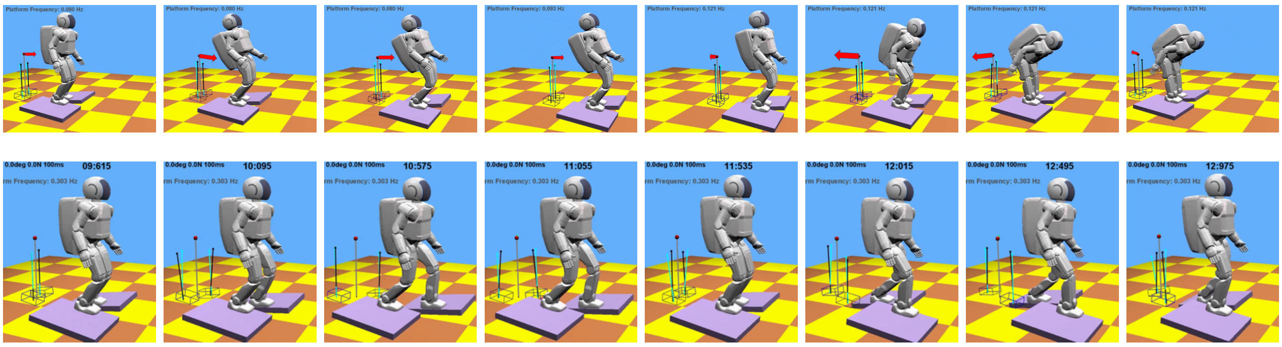


Fig. 10: Top row: The two supports translate forward and backward with the same speed. In order to maintain balance, the robot rotates its trunk in a periodic manner. The red arrows indicate the direction and magnitude of the linear momentum of the robot. Bottom row: The robot maintains balance on moving supports. The two foot support surfaces have different inclination angles and out of phase front-back velocities.

4.3 Computational Costs of Optimization Processes

Our framework includes solving three optimization problems and each problem can be solved efficiently. We solve (23) using the Non-Negative Least-Squares algorithm [21]. In our experiment, it takes about 0.009 millisecond to solve (23) (www.netlib.org/lawson-hanson/all). Equation (28) can be solved using the Bounded-variable least squares algorithm [39] (http://lib.stat.cmu.edu/general/bvls) and the computation time varied from 0.006 to 0.01 millisecond. Altogether, the two optimization problems take less than 0.02 millisecond. This is significantly less than what quadratic programming (QP) methods would take. In our experiment, Goldfarb-Idnani dual QP solver (http://sourceforge.net/projects/quadprog/) took 0.03 milliseconds to solve the problem, which is about 50% slower than our sequential method.

Equation (33) takes the longest, naturally because of the highest dimension of the unknowns, and it took about 0.11 millisecond with our rather naive implementation of the least-squares solver. We experimented using Intel's 2.66GHz Core2 Quad CPU without utilizing multi-core functionality.

5 Discussion and Future Work

In this paper, we introduced a novel balance control method for humanoid robots on non-level, non-continuous, and non-stationary grounds. By controlling both linear and angular momenta of the robot, this whole body balance controller can maintain balance under relatively large perturbations and often generates human-like balancing behavior. The controller can deal with different ground geometry and ground frictions at each foot by determining the GRF and CoP at each support foot. For efficient optimization for the foot GRFs and CoPs during double support, we developed a novel method to determine the foot GRFs and CoPs sequentially by solving two small constrained linear least-squares problems. We showed the performance of the balance controller through a number of simulation experiments.

The characteristic features of the presented controller are as follows:

- Both angular and linear momenta of the robot are controlled for balance maintenance and the control policy is defined in terms of the desired momenta.
- One can choose to satisfy linear and angular momenta in different proportions, as the situation demands.
- Desired foot GRF and foot CoP are directly computed from the desired momentum without requiring to compute the net GRF and net CoP, which makes the framework applicable to non-level ground at each foot without having to compute rather complex convex hulls made by contact points to check feasibility of the net CoP.
- For double support, we compute foot GRFs and foot CoPs that minimize the ankle torques.

Fig. 11 contains two plots showing the performance limits of the balance controller on stationary floor and corresponds to a forward push on the robot. The first plot shows the maximum impulse, which is the product of the magnitude of an impact force and its duration, that the balance controller can survive. The second plot shows the maximum duration for which a given impact force can be survived.

According to the first plot, the maximum impulse that the robot can successfully handle drops quite rapidly and becomes more or less constant for a force larger than 80 N. The maximum duration of push that the robot can handle drops even more precipitously as the force magnitude increases.

The message delivered by the plots is that the balance control strategy is somewhat weak against a long duration push. A major reason for this is in general, a humanoid robot can generate angular momentum for only a short time: the trunk and the legs, which are the most effective body parts in creating large angular momentum, cannot rotate indefinitely due to the joint limits and self-collision possibility. Therefore, the strategy of modulating angular momentum for balance maintenance has limitations against a continuous push.

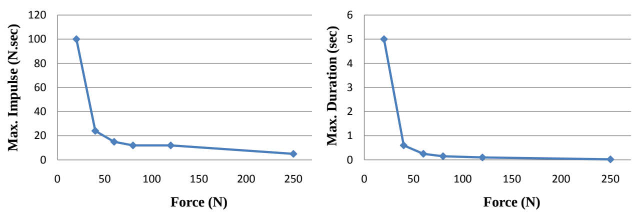


Fig. 11: Maximum impulse (left) and duration (right) of forward push (Fig. 8, top row) that the balance controller can handle for given magnitude of force.

A possible remedy for surviving a long-duration push would be to take a different strategy rather than modulating angular momentum. For example, in the situation of Fig. 10 (top row), if the robot could estimate the inertial force, it could maintain balance by leaning the body *against* the accelerating direction of the moving platforms, instead of rotating its trunk as the current controller does.

Our algorithm currently does not utilize the force and torque sensor data when it determines the desired and admissible GRF and CoP. While this can be regarded as a feature of our method for a robot that is not equipped with force-torque sensors, the difference between the actual and desired values can become significant as the errors in physical parameter of the dynamics model increase. As many of today's humanoid robots are equipped with force-torque sensors at the foot, using the sensory information as a feedback data is available to many humanoid robots and can help reduce the difference between the actual and desired GRF and CoP. Also the sensory data could be further used for estimating the direction and magnitude of external perturbations.

The proposed method takes an inverse dynamics-based approach, which depends on an accurate knowledge of the physical parameters of the robot. Since a large modeling error may negatively influence the performance of the controller, it is an important future work to improve the balance controller to be more robust against modeling errors.

Due to the unilateral nature of the robot-ground contact, all postural balance controllers have intrinsic limitations. Therefore, another important venue of future work is to develop a different type of balance controller that will deal with the larger external disturbance than the postural balance controller can handle. For example, balance maintenance through stepping (Fig. 3) can cope with larger perturbations and will increase the push-robustness of the robot significantly [35].

References

1. Abdallah, M., Goswami, A.: A biomechanically motivated two-phase strategy for biped robot upright balance control. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 3707–3713 (2005, Barcelona, Spain)
2. Abe, Y., da Silva, M., Popović, J.: Multiobjective control with frictional contacts. In: Eurographics/ACM SIGGRAPH Symposium on Computer Animation, pp. 249 – 258 (2007)
3. Ahn, K.H., Oh, Y.: Walking control of a humanoid robot via explicit and stable CoM manipulation with the angular momentum resolution. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2006)
4. Bloch, A.M., Krishnaprasad, P.S., Marsden, J.E., Murray, R.M.: Nonholonomic mechanical systems with symmetry. Archive for Rational Mechanics and Analysis **136**(1), 21–99 (Dec. 1996)
5. Choi, Y., Kim, D., Oh, Y., You, B.J.: Posture/walking control for humanoid robot based on kinematic resolution of CoM Jacobian with embedded motion. IEEE Trans. on Robotics **23**(6), 1285–1293 (2007)
6. Featherstone, R.: Robot Dynamics Algorithms. Kluwer Academic Publishers (1987)
7. Fujimoto, Y.: Study on biped walking robot with environmental force interaction. Ph.D. thesis, Yokohama National University (1998)
8. Fujimoto, Y., Kawamura, A.: Simulation of an autonomous biped walking robot including environmental force interaction. IEEE Robotics & Automation Magazine **5**(2), 33–41 (1998)
9. Fujimoto, Y., Obata, S., Kawamura, A.: Robust biped walking with active interaction control between foot and ground. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 2030–2035 (1998)
10. Goswami, A., Kallem, V.: Rate of change of angular momentum and balance maintenance of biped robots. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 3785–3790 (2004)
11. Hofmann, A., Popovic, M., Herr, H.: Exploiting angular momentum to enhance bipedal center-of-mass control. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 4423–4429 (2009)
12. Huang, Q., Nakamura, Y.: Sensory reflex control for humanoid walking. IEEE Trans. on Robotics **21**(5), 977–984 (Oct. 2005)
13. Hyon, S.H.: Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces. IEEE Trans. on Robotics **25**(1), 171–178 (2009)
14. Hyon, S.H., Hale, J., Cheng, G.: Full-body compliant human–humanoid interaction: Balancing in the presence of unknown external forces. IEEE Trans. on Robotics **23**(5), 884–898 (2007)
15. Kagami, S., Kanehiro, F., Tamiya, Y., Inaba, M., Inoue, H.: AutoBalancer: An online dynamic balance compensation scheme for humanoid robots. In: Proc. of the 4th International Workshop on Algorithmic Foundation on Robotics (2000)
16. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Resolved momentum control: Humanoid motion planning based on the linear and angular momentum. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS), vol. 2, pp. 1644–1650 (2003, Las Vegas, NV, USA)
17. Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., Hirukawa, H.: The 3D linear inverted pendulum model: A simple modeling for a biped walking pattern generator. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS), pp. 239–246 (2001, Maui, Hawaii)
18. Komura, T., Leung, H., Kudoh, S., Kuffner, J.: A feedback controller for biped humanoids that can counteract large perturbations during gait. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 2001–2007 (2005, Barcelona, Spain)
19. Kudoh, S., Komura, T., Ikeuchi, K.: The dynamic postural adjustment with the quadratic programming method. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2002)
20. de Lasa, M., Mordatch, I., Hertzmann, A.: Feature-based locomotion controllers. ACM Transactions on Graphics **29**(3) (2010)
21. Lawson, C.L., Hanson, R.J.: Solving least squares problems. Prentice-Hall (1974)
22. Lee, S.H., Goswami, A.: Ground reaction force control at each foot: A momentum-based humanoid balance controller for non-level and non-stationary ground. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2010)
23. Lourakis, M.: levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++. [web page] `http://www.ics.forth.gr/~lourakis/levmar/` (Jul. 2004)
24. Macchietto, A., Zordan, V., Shelton, C.R.: Momentum control for balance. ACM Transactions on Graphics **28**(3), 80:1–80:8 (2009)
25. Michel, O.: Webots: Professional mobile robot simulation. International Journal of Advanced Robotic Systems **1**(1), 39–42 (2004)
26. Mitobe, K., Capi, G., Nasu, Y.: A new control method for walking robots based on angular momentum. Mechatronics **14**(2), 163–174 (2004)
27. Muico, U., Lee, Y., Popović, J., Popović, Z.: Contact-aware nonlinear control of dynamic characters. ACM Transactions on Graphics **28**(3) (2009)
28. Murray, R.M., Li, Z., Sastry, S.S.: A Mathematical Introduction to Robotic Manipulation. CRC Press (1994)
29. Naksuk, N., Mei, Y., Lee, C.: Humanoid trajectory generation: an iterative approach based on movement and angular momentum criteria. In: IEEE/RAS Intn'l Conf. on Humanoid Robots, pp. 576–591 (2004)

30. Orin, D., Goswami, A.: Centroidal momentum matrix of a humanoid robot: Structure and properties. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2008, Nice, France)
31. Park, J., Han, J., Park, F.: Convex optimization algorithms for active balancing of humanoid robots. IEEE Trans. on Robotics **23**(4), 817–822 (2007)
32. Park, J., Youm, Y., Chung, W.K.: Control of ground interaction at the zero-moment point for dynamic control of humanoid robots. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 1724 – 1729 (2005)
33. Pollard, N.S., Reitsma, P.S.A.: Animation of humanlike characters: Dynamic motion filtering with a physically plausible contact model. In: Yale Workshop on Adaptive and Learning Systems (2001)
34. Popovic, M., Hofmann, A., Herr, H.: Angular momentum regulation during human walking: Biomechanics and control. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 2405–2411 (2004)
35. Pratt, J., Carff, J., Drakunov, S., Goswami, A.: Capture point: A step toward humanoid push recovery. In: EEE-RAS/RSJ International Conference on Humanoid Robots (2006)
36. Sano, A., Furusho, J.: Realization of natural dynamic walking using the angular momentum information. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 1476 – 1481 (1990)
37. Sentis, L., Park, J., Khatib, O.: Compliant control of multicontact and center-of-mass behaviors in humanoid robots. IEEE Trans. on Robotics **26**(3), 483–501 (2010)
38. Sian, N.E., Yokoi, K., Kajita, S., Kanehiro, F., Tanie, K.: Whole body teleoperation of a humanoid robot -a method of integrating operators intention and robot's autonomy. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA) (2003)
39. Stark, P.B., Parker, R.L.: Bounded-variable least-squares: an algorithm and applications. Computational Statistics **10**, 129–141 (1995)
40. Stephens, B.: Integral control of humanoid balance. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2007)
41. Stephens, B.J., Atkeson, C.G.: Dynamic balance force control for compliant humanoid robots. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2010)
42. Sugihara, T.: Mobility enhancement control of humanoid robot based on reaction force manipulation via whole body motion. Ph.D. thesis, University of Tokyo (2003)
43. Sugihara, T., Nakamura, Y.: Variable impedant inverted pendulum model control for a seamless contact phase transition on humanoid robot. In: IEEE-RAS/RSJ International Conference on Humanoid Robots (2003)
44. Sugihara, T., Nakamura, Y., Inoue, H.: Realtime humanoid motion generation through ZMP manipulation based on inverted pendulum control. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA), pp. 1404–1409 (2002)
45. Ugurlu, B., Kawamura, A.: Eulerian ZMP resolution based bipedal walking: Discussions on the rate change of angular momentum about center of mass. In: IEEE Intn'l Conf. on Robotics and Automation (ICRA) (2010)
46. Vukobratović, M., Juričić, D.: Contribution to the synthesis of biped gait. IEEE Trans. Bio-Medical Eng. **16**(1) (1969)
47. Wieber, P.B.: Holonomy and nonholonomy in the dynamics of articulated motion. In: Fast Motions in Biomechanics and Robotics, Heidelberg, Germany (2005)
48. Ye, Y., Liu, C.K.: Optimal feedback control for character animation using an abstract model. ACM Transactions on Graphics **29**(3) (2010)
49. Yun, S.K., Goswami, A.: Momentum-based reactive stepping controller on level and non-level ground for humanoid robot push recovery. In: IEEE/RSJ Intn'l Conf. on Intelligent Robots and Systems (IROS) (2011)
50. Zhou, C., Meng, Q.: Dynamic balance of a biped robot using fuzzy reinforcement learning agents. Fuzzy Sets and Systems **134**(1), 169–187 (2003)