Centroidal dynamics of a humanoid robot

David E. Orin · Ambarish Goswami · Sung-Hee Lee

Received: 20 September 2012 / Accepted: 1 June 2013 / Published online: 19 June 2013

Abstract The center of mass (CoM) of a humanoid robot occupies a special place in its dynamics. As the location of its effective total mass, and consequently, the point of resultant action of gravity, the CoM is also the point where the robot's aggregate linear momentum and angular momentum are naturally defined. The overarching purpose of this paper is to refocus our attention to *centroidal dynamics*: the dynamics of a humanoid robot projected at its CoM. In this paper we specifically study the properties, structure and computation schemes for the centroidal momentum matrix (CMM), which projects the generalized velocities of a humanoid robot to its spatial centroidal momentum. Through a *transformation diagram* we graphically show the relationship between this matrix and the well-known joint-space inertia matrix. We also introduce the new concept of "average spatial velocity" of the humanoid that encompasses both linear and angular components and results in a novel decomposition of the kinetic energy. Further, we develop a very efficient $O(N)$ algorithm, expressed in a compact form using spatial notation, for computing the CMM, centroidal momentum, centroidal inertia, and average spatial velocity. Finally, as a practical use of centroidal dynamics we show that a momentum-based balance controller that directly employs the CMM can significantly reduce unnecessary trunk bending during balance maintenance against external disturbance.

Keywords Centroidal momentum matrix · Angular momentum · Robot dynamics algorithms · Average spatial velocity · Humanoid balance controller · Momentum based balance control

1 Motivation

The center of mass (CoM) of a humanoid robot is a uniquely important point in its dynamics. First of all, it is the effective location of the robot's total mass, and therefore, the point where its aggregate linear momentum is naturally defined. It is also the point through which the resultant gravity force acts. It should then come as no surprise that virtually all reduced humanoid models and control algorithms contain the CoM as an integral component.

In the well-known example of a freely flying multi-link chain, the average behavior of the chain can be adequately described in terms of its CoM. While the dynamics of individual member links can be quite complex, the motion of the CoM follows a point-mass trajectile profile which can be easily described and communicated. Additionally, the rotational motion of the aggregate chain obeys the conservation of angular momentum about the CoM or, the centroidal angular momentum. For many applications, such reduced description is instrumental in the analysis and control of the system.

In a similar manner, surprisingly deep insight into the dynamics of a humanoid robot can be obtained simply by following the trajectory of its CoM, center of pressure (CoP),

Electronic supplementary material The online version of this article (doi:10.1007/s10514-013-9341-4) contains supplementary material, which is available to authorized users.

D. E. Orin
Department of Electrical and Computer Engineering,
The Ohio State University, Columbus, OH 43210, USA
e-mail: orin.1@osu.edu

A. Goswami (✉)
Honda Research Institute, Mountain View, CA 94043, USA
e-mail: agoswami@honda-ri.com

S.-H. Lee
Graduate School of Culture Technology, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea
e-mail: leesunghee@gmail.com

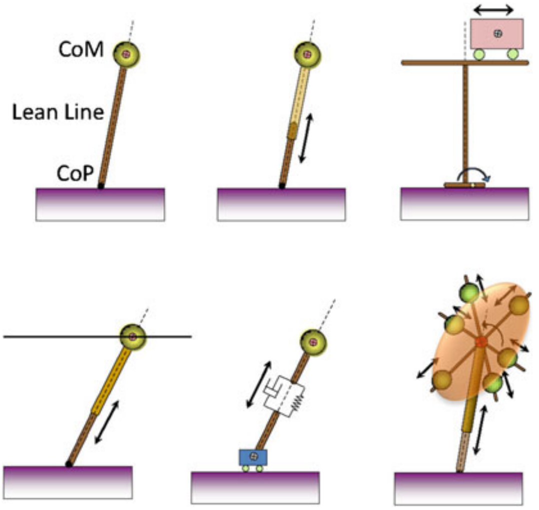

Fig. 1 Schematic depiction of some reduced one-legged models used in human and humanoid balance and gait analysis. The models are from *top left*: a rigid inverted pendulum, a telescopic inverted pendulum, a cart-table model, a linear inverted pendulum model (LIPM), a variable impedance inverted pendulum and a reaction mass pendulum (RMP). Note that all models are based upon the locations of the CoM, CoP and the "lean line" connecting them. Only the RMP model contains an extended rigid-body mass and the centroidal angular momentum

and the "lean line" connecting these two points. This has been known for a long time and has been utilized in the study of human motion. The study of humanoid dynamics has also inherited this trend and a number of progressively complex models, some of which are listed in Fig. 1, are currently used for analysis and control.

The purpose of this paper is to explore *centroidal dynamics*: the dynamics of a humanoid robot projected at its CoM. We believe that a proper understanding of centroidal dynamics and associated computational and algorithmic tools will help create better robot controllers. Our position is schematically depicted in Fig. 2 and is motivated by a system level view of the humanoid dynamics (Goswami and Kallem 2004). In this view, we consider all the external forces on the humanoid system, which include the gravity force $M\boldsymbol{g}$ at the humanoid CoM, as well as the interaction force/moment between the humanoid and its environment. The interaction force/moment includes the so-called ground reaction forces (GRF) between the robot feet and the support surface and all the task-related or other accidental forces applied on the robot hand or other limbs. According to a fundamental principle of dynamics (Newton's laws of motion) the rate of change of linear and angular momentum at the CoM, given by $\dot{\mathbf{l}}_G$ and $\dot{\mathbf{k}}_G$, respectively, is equivalent to the resultant effect of all the external forces (Sciavicco and Siciliano 2005).

Taking a fresh interest in the centroidal dynamics is not a big departure from the existing literature, but rather a refocusing of our attention. In a sense the CoM space, in which the centroidal dynamics is described, is but an example of the task space or the operational space (Khatib 1987), which have been described for traditional manipulators and for floating-base robots (Mistry and Righetti 2012). True, there are differences between the CoM space, the coordinate frame of which is purely computational and often floats in space, and the typical task space, for which the coordinate frame is rigidly attached to the physical end effector of a manipulator. Yet, the same concept of recasting the dynamics from its native joint space to a different space applies in both cases.

Seen from another angle, the exploitation and control of centroidal dynamics is already practiced when a researcher migrates from the tedious joint-trajectory based ZMP control of humanoids in favor of direct CoM control using simple models, such as the LIPM model. The CoM control is directly related to the linear momentum control, and the quantity of linear momentum is meaningfully defined only at the CoM of the robot. The next natural step in this direction seems to be the control of angular momentum.

However, additional questions concerning centroidal angular momentum have prevented its adoption in some previous control strategies. First, angular momentum has no a-priori affinity with the CoM, and some have seen this as a possible limit to its usefulness. While it appears that the angular momentum expressed at the CoM should not be privileged over any other point, some studies have shown otherwise (Herr and Popovic 2008). Moreover, the angular momentum is not integrable. That is, while the linear momentum can be integrated to yield the CoM position, angular momentum cannot be integrated to yield any meaningful orientation of the humanoid as a function of its configuration. We can numerically integrate angular momentum, but the result would become dependent on joint trajectory history and not simply on the configuration (Papadopoulos 1990).

Despite these challenges we are able to demonstrate in this paper that a clear advantage exists in the concerted control of the linear and centroidal angular momentum of a humanoid for whole body control including balance maintenance. In an example we show that such a controller is able to reduce unnecessary trunk sway of a robot by almost 10° compared to a traditional joint control under the same external perturbation.

A further point to note is that any control approach that is based on a reduced model of the humanoid robot, such as the ones using the centroidal dynamics, cannot immediately provide full joint control of the robot. For example, a CoM-based linear momentum control is directly concerned only with the manipulation of the CoM motion. The actual joint trajectories are subsequently obtained by imposing additional constraints or tasks. This is to be viewed as the precise purpose and a merit of such methods, which influence only a small number of variables that are sufficient for the core task

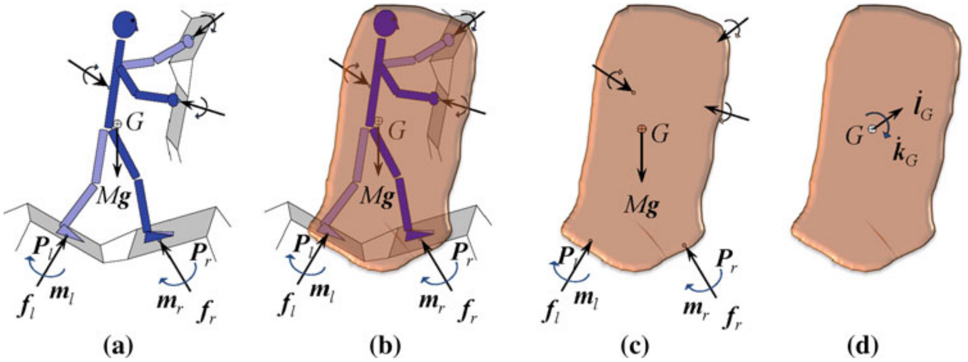

Fig. 2 Importance of centroidal dynamics of humanoid robot. From *left*: **a** A humanoid is subjected to both internal joint force/torque as well as external forces: gravity force, ground reaction force, and interaction force. **b** We treat the humanoid as a system and only consider the external interaction. **c** The view in **b** is emphasized by only showing the external interaction forces. **d** The resultant effect of the all the external interaction forces in **c** is a force and a moment, or equivalently, the linear and angular momentum rate changes, $\dot{\mathbf{l}}_G$ and $\dot{\mathbf{k}}_G$, at the CoM

(e.g., balance maintenance), leaving the rest of the degrees of freedom free to accomplish additional tasks.

2 Contribution and related work

In this paper we particularly study the properties, structure and computation schemes for the centroidal momentum matrix (CMM) (Orin and Goswami 2008), which projects the generalized velocities of a humanoid robot to its spatial centroidal momentum. Our interest in the study of the CMM is motivated by our desire to exploit the centroidal momentum as a means to control the robot's postural balance.

The importance of angular momentum in humanoid walking was reported by Sano and Furusho (1990) as early as 1990. However, it was much later before its importance for balance maintenance for humans and humanoid robots started to be seriously explored (Nishiwaki et al. 2002; Goswami and Kallem 2004; Naksuk et al. 2004, 2005; Komura et al. 2005; Abdallah and Goswami 2005; Popovic et al. 2004, 2005). Sano and Furusho (1990) and Mitobe et al. (2004) showed that it is possible to generate the desired angular momentum by controlling the ankle torque. Kajita et al. (2003) included angular momentum criteria into the whole body control framework for balance maintenance.

The field of computer animation has been employing angular momentum for realistic dynamic movements for some time as described in a recent survey paper Zordan (2010). In an important recent work Macchietto et al. (2009) define balance control objectives through desired momentum rate change. They employ the CMM to compute joint accelerations, followed by computing necessary joint torques using inverse dynamics. Hofmann et al. (2009) presented a method that controls the CoM by modulating angular momentum under large external perturbations. Ugurlu and Kawamura have studied bipedal walking that specifically controls the centroidal angular momentum (Ugurlu and Kawamura 2010). Relatively recently we have seen a comprehensive study of angular momentum during human gait (Herr and Popovic 2008).

This paper also addresses the important concept of "average" angular velocity of a humanoid. Although the average linear velocity of a multibody system can be uniquely described in terms of the velocity of its CoM, no such fundamental description exists regarding its angular velocity. Following the approach of Essén (1993) for a system of particles we derive a principled definition of average spatial velocity that results in a novel decomposition of kinetic energy. This definition generalizes the concept of linear velocity and encompasses both linear and angular components.

Another contribution of this paper is the *transformation diagram*, which we use to pictorially represent the relations between a number of important vectors and transformation matrices in a unified scheme. Finally, we also present an efficient $O(N)$ algorithm for computing the CMM, centroidal momentum, centroidal inertia, and average spatial velocity where N is the number of links in the humanoid.

Although we have not stated so explicitly, our approach implicitly assumes a robot model which possesses an extended mass with non-zero inertia. The search for an appropriate reduced model must factor in both the utility of the adopted model and its complexity. Depending on the application domain of a model and the physical effects the model is intended to capture, there is more than one way to improve the simple "point mass" inverted pendulum, each with its unique pros and cons. We have chosen to incorporate elements to model a variable inertia matrix at the CoM. This is primarily due to our interest in exploring and controlling centroidal inertia and centroidal angular momentum. Another approach may put priority on the nonholonomic factors and the coupling effects, and may include an eccentric inertia—a dangling link—as proposed in Wieber (2005, 2008). There would presumably be yet other ways to extend the simple model, and perhaps combinations of different approaches.

The organization of this paper is as follows. First, the velocity and momentum equations for a humanoid robot are derived using spatial notation. This is followed by a description of the structure and properties of the CMM. Next, we introduce the novel concept of "average" spatial (centroidal) velocity of a humanoid, which is a generalization of the CoM velocity. Then we introduce the transformation diagram, which is a pictorial representation of the inter-relationships among motion and momentum variables. In the two subsequent sections we present efficient computational algorithms for several of the quantities that appear in this paper, and also show a scheme to address the constraints resulting from single and double support of the humanoid. We finally demonstrate the utility of centroidal dynamics through a momentum-based balance controller that explicitly uses the CMM. We show that in comparison with the traditional joint control, the centroidal dynamics based control can reduce unnecessary trunk roll of the robot by as much as 10° under the same external perturbation.

3 Humanoid robot model

In order to develop the dynamic model of a humanoid robot, the approach taken in Featherstone and Orin (2008) for rigid-body systems will be used. Spatial notation (Featherstone 2008; Featherstone and Orin 2008) is a concise vector notation for describing rigid-body velocity, acceleration, inertia, etc., using 6D vectors and tensors, and is an integral part of the approach. While spatial notation is particularly convenient here, note that the same theoretical framework can be developed for centroidal dynamics in more traditional 3D notation with no change in the central contributions of the paper.

A humanoid can be modeled as a set of $N+1$ links interconnected by N joints, of up to six degrees of freedom each, forming a tree-structure topology. The motion of the links is referenced to a fixed base (inertial frame) which is labeled 0 while the links are labeled from 1 through N. Numbering of the links may be done in any manner such that link i's predecessor toward the root (link 0), indicated by $p(i)$, is always less than i. Joints in the tree are numbered such that joint i connects link i to link $p(i)$. A coordinate frame is attached to each link to provide a reference for quantities associated with the link.

The relationship between connected links in the tree structure is described using the general joint model of Roberson and Schwertassek (1988). An $n_i \times 1$ vector $\dot{\mathbf{q}}_i$ relates the velocity of link i to the velocity of its predecessor, link $p(i)$, where n_i is the number of degrees of freedom at the joint connecting the two links. The free modes of the joint are represented by the $6 \times n_i$ matrix $\mathbf{\Phi}_i$, such that the spatial velocity of link i is given as follows:

$$\mathbf{v}_i = \begin{bmatrix} \boldsymbol{\omega}_i \\ \boldsymbol{v}_i \end{bmatrix} = {}^i\mathbf{X}_{p(i)}\, \mathbf{v}_{p(i)} + \mathbf{\Phi}_i\, \dot{\mathbf{q}}_i \ , \tag{1}$$

where $\boldsymbol{\omega}_i$ and $\boldsymbol{v}_i$ are the angular and linear velocities of link i, respectively, as referenced to the link coordinate frame. ${}^i\mathbf{X}_{p(i)}$ is a 6×6 spatial transform which transforms spatial motion vectors from $p(i)$ to i coordinates. The matrix $\mathbf{\Phi}_i$ depends on the type of joint (Roberson and Schwertassek 1988; Featherstone and Orin 2008). It has full column rank, as does the orthogonal matrix $\mathbf{\Phi}_i^c$ representing the constrained modes of the joint, such that $\begin{bmatrix} \mathbf{\Phi}_i & \mathbf{\Phi}_i^c \end{bmatrix}$ is a basis of $\mathbb{R}^6$ and is invertible.

In order to model a humanoid when in flight, one of the links is modeled as a floating base (typically the torso) and numbered as link 1. A fictitious six degree-of-freedom (DoF) joint is inserted between the floating base and fixed base. In this case, $\mathbf{\Phi}_1 = \mathbf{1}_{6\times 6}$ where $\mathbf{1}_{6\times 6}$ is the identity matrix. The Denavit–Hartenberg convention is used for single DoF joints, such that $\mathbf{\Phi}_i = [0\,0\,1\,0\,0\,0]^T$ for a revolute joint. The total number of degrees of freedom in the humanoid is n where $n = \sum n_i$. Note that n includes the six degrees of freedom for the floating base.

The spatial transform ${}^i\mathbf{X}_{p(i)}$ may be composed from the position vector ${}^{p(i)}\mathbf{p}_i$ from the origin of coordinate frame $p(i)$ to the origin of i, and the 3×3 rotation matrix ${}^i\mathbf{R}_{p(i)}$ which transforms 3D vectors from coordinate frame $p(i)$ to i:

$${}^i\mathbf{X}_{p(i)} = \begin{bmatrix} {}^i\mathbf{R}_{p(i)} & \mathbf{0} \\ {}^i\mathbf{R}_{p(i)}\, \mathbf{S}({}^{p(i)}\mathbf{p}_i)^T & {}^i\mathbf{R}_{p(i)} \end{bmatrix} . \tag{2}$$

The quantity $\mathbf{S}(\mathbf{p})$ is the skew-symmetric matrix that satisfies $\mathbf{S}(\mathbf{p})\, \boldsymbol{\omega} = \mathbf{p} \times \boldsymbol{\omega}$ for any 3D vector $\boldsymbol{\omega}$.

3.1 Spatial momentum

The spatial momentum of each link may be computed from the spatial velocity as follows (see Fig. 3):

$$\boldsymbol{h}_i = \begin{bmatrix} \mathbf{k}_i \\ \mathbf{l}_i \end{bmatrix} = \mathbf{I}_i\, \mathbf{v}_i \ , \tag{3}$$

where $\mathbf{k}_i$ is the angular momentum, $\mathbf{l}_i$ is the linear momentum, and $\mathbf{I}_i$ is the spatial inertia for link i. The spatial inertia may be composed from the mass m_i, position vector $\mathbf{c}_i$ to the CoM of link i, and 3×3 rotational inertia $\bar{\mathbf{I}}_i$, all relative to coordinate frame i:

$$\mathbf{I}_i = \begin{bmatrix} \bar{\mathbf{I}}_i & m_i\, \mathbf{S}(\mathbf{c}_i) \\ m_i\, \mathbf{S}(\mathbf{c}_i)^T & m_i\, \mathbf{1} \end{bmatrix}, \tag{4}$$

where

$$\bar{\mathbf{I}}_i = \bar{\mathbf{I}}_i^{cm} + m_i\, \mathbf{S}(\mathbf{c}_i)\, \mathbf{S}(\mathbf{c}_i)^T \ , \tag{5}$$

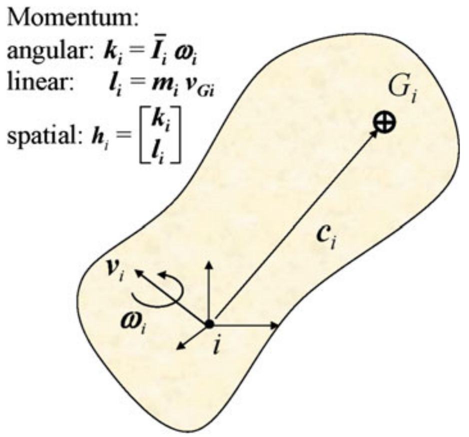
Momentum:
angular: $k_i = \bar{I}_i \, \omega_i$
linear: $l_i = m_i \, v_{Gi}$
spatial: $h_i = \begin{bmatrix} k_i \\ l_i \end{bmatrix}$
G_i
c_i
v_i
ω_i
i

Fig. 3 Schematic depiction of a single rigid body: spatial momentum contains the angular and linear momenta

and $\bar{\mathbf{I}}_i^{cm}$ is the rotational inertia about the CoM. Recall that if the origin of coordinate frame i is chosen at the CoM, the off-diagonal blocks $m_i\,\mathbf{S}(\mathbf{c}_i)$ reduce to zero. If, in addition, the axes of coordinate frame i are oriented along the principal axes of inertia, $\bar{\mathbf{I}}_i$ becomes a 3×3 diagonal matrix and $\mathbf{I}_i$ a 6×6 diagonal matrix.

3.2 Global notation: system Jacobian and system inertia

It is possible to combine the equations for the velocity or momentum for all the links into a global set of equations (Featherstone and Orin 2000). To do so, composite vectors and matrices are defined, and this was the starting point of the spatial operator algebra developed by Rodriguez et al. (1991). Global notation is useful in developing a system Jacobian and system inertia which leads to an expression for the centroidal momentum matrix (CMM).

Gathering all of the link velocities and joint velocities together, the system Jacobian $\boldsymbol{J}$ can be defined to give the relationship between the two[1]:

$$\mathbf{v} = \boldsymbol{J}\,\dot{\mathbf{q}}\,, \tag{6}$$

where

$$\mathbf{v} = \left[\mathbf{v}_1^T,\, \mathbf{v}_2^T, \ldots \mathbf{v}_i^T, \ldots \mathbf{v}_N^T\right]^T \tag{7}$$

$$\dot{\mathbf{q}} = \left[\dot{\mathbf{q}}_1^T,\, \dot{\mathbf{q}}_2^T, \ldots \dot{\mathbf{q}}_i^T, \ldots \dot{\mathbf{q}}_N^T\right]^T \ . \tag{8}$$

The elements of the system Jacobian are just the Jacobians for each of the links:

$$\boldsymbol{J} = \left[\mathbf{J}_1^T,\, \mathbf{J}_2^T, \ldots \mathbf{J}_i^T, \ldots \mathbf{J}_N^T\right]^T \ . \tag{9}$$

The momenta of all the links in the system may be determined as the product of the system velocity vector $\mathbf{v}$ and the system inertia $\boldsymbol{I}$; gathering all:

$$\boldsymbol{h} = \boldsymbol{I}\,\mathbf{v}\,, \tag{10}$$

where $\boldsymbol{h}$ is the $6N \times 1$ system momentum vector:

$$\boldsymbol{h} = \left[\boldsymbol{h}_1^T,\, \boldsymbol{h}_2^T, \ldots \boldsymbol{h}_i^T, \ldots \boldsymbol{h}_N^T\right]^T\,, \tag{11}$$

and the $6N \times 6N$ system inertia matrix is defined as:

$$\boldsymbol{I} = \mathrm{diag}\left[\mathbf{I}_1,\, \mathbf{I}_2, \ldots \mathbf{I}_i, \ldots \mathbf{I}_N\right] \ . \tag{12}$$

In conclusion to this section, let us note that spatial notation results in compact equations whose vectors and matrices contain both angular and linear parts. Further, the use of global notation illuminates the underlying structure of centroidal dynamics as we will see in the next section.

4 Structure and properties of centroidal momentum matrix (CMM)

The aggregate momentum of a humanoid may be obtained by summing up all of the angular and linear momenta contributed by the individual link segments. The link momenta need to be projected to a common reference point and because of its special properties, the CoM, or centroid, is used for this purpose. The 6×1 centroidal momentum vector $\boldsymbol{h}_G$, which consists of the linear and centroidal angular momenta of the robot, is related to its $n \times 1$ joint velocity vector $\dot{\mathbf{q}}$ as:

$$\boldsymbol{h}_G = \boldsymbol{A}_G(\mathbf{q})\,\dot{\mathbf{q}}\,. \tag{13}$$

The $6 \times n$ matrix $\boldsymbol{A}_G$ is called the CMM and in this section we study its structure and properties. In our formulation, $\boldsymbol{A}_G$ contains contributions both from the inter-segmental joint variables of the robot as well as from the fictitious joint connecting the floating base of the humanoid to the inertial frame (detailed in Sect. 3). Note that $\boldsymbol{A}_G$ is identical to the large matrix in the RHS of Eq. 1 of Kajita et al. (2003), with only the linear and angular parts interchanged.

The literature contains relatively few references to matrices that map joint rates into aggregate momenta of a multibody dynamic system. In Fang and Pollard (2003) the “linear momentum Jacobian” is obtained as an intermediate step towards computing what the authors refer to as a force Jacobian. This formulation is used for animating articulated figures and does not contain angular momentum. In Morita and Ohnishi (2003) the “angular momentum Jacobian” matrix is used to control the flight phase of a hopping robot. Finally, for resolved momentum control of humanoid robots, use has been made of “matrices which indicate how the joint speeds affect the linear momentum and angular momentum” (Kajita

[1] The system Jacobian is not to be confused with the manipulator Jacobian in traditional fixed-based manipulators. The system Jacobian is an extension of the manipulator Jacobian and can contain it as one of its blocks if the corresponding coordinate frame is located at the task point.

et al. 2003). Although they have been called "inertia matrices" in this work, these matrices are identical to the "momentum Jacobian" matrices mentioned before.

Is $\boldsymbol{A}_G$ an inertia matrix or a Jacobian matrix? In the following development, we will show that $\boldsymbol{A}_G$ can be represented as the product of a spatial transformation matrix, an inertia matrix, and a Jacobian matrix. In Sect. 6, we show its relationship to other important quantities including the joint-space inertia matrix.

As shown in Eq. 13, the CMM gives the relationship between the joint rates and centroidal momentum. In order to find the relationship between this matrix and the link inertias and Jacobians, the concept of the *system momentum matrix* $\boldsymbol{A}$ is first presented. The system momentum matrix $\boldsymbol{A}$ expresses the relationship between the system momentum vector and the joint rates: $\boldsymbol{h} = \boldsymbol{A}\,\dot{\mathbf{q}}$. Substituting the expression for the system velocity in Eq. 6 into Eq. 10, and using the definition of the system momentum matrix, we can write:

$$\boldsymbol{A} = \boldsymbol{I}\ \boldsymbol{J}\ . \tag{14}$$

The system momentum matrix is just the product of the system inertia matrix and the system Jacobian and is of size $6N \times n$.

As defined, the spatial momentum of each link $\boldsymbol{h}_i$ is most naturally expressed in its own coordinate system. As a measure of dynamic stability or for control, it is useful to combine the momenta for the links by projecting the momenta to a common coordinate frame. A convenient frame is one set at the instantaneous CoM or the centroid of the system G, and whose coordinate axes are parallel to those of the inertial coordinate frame 0. Noting that the spatial momentum may be projected as any other force-type vector (Featherstone and Orin 2008), the following equation may be used to calculate the spatial momentum at the centroid of the system (see Fig. 4):

$$\boldsymbol{h}_G = \sum_{i=1}^{N} {}^{i}\mathbf{X}_G^T\,\boldsymbol{h}_i = \mathbf{X}_G^T\,\boldsymbol{h}\ , \tag{15}$$

where $\mathbf{X}_G$ is defined as the projection matrix, for motion vectors, from centroidal coordinates to link coordinates and is given as follows:

$$\mathbf{X}_G = \left[\,{}^{1}\mathbf{X}_G^T,\ {}^{2}\mathbf{X}_G^T,\ \ldots {}^{i}\mathbf{X}_G^T,\ \ldots {}^{N}\mathbf{X}_G^T\,\right]^T\ . \tag{16}$$

The centroidal momentum may also be expressed as a function of the system momentum matrix $\boldsymbol{A}$:

$$\boldsymbol{h}_G = \mathbf{X}_G^T\,\boldsymbol{A}\,\dot{\mathbf{q}}\ . \tag{17}$$

Noting Eqs. 13 and 17, and using the expression for $\boldsymbol{A}$ in Eq. 14, the CMM, $\boldsymbol{A}_G$, may then be defined as:

$$\boldsymbol{A}_G = \mathbf{X}_G^T\,\boldsymbol{A} = \mathbf{X}_G^T\,\boldsymbol{I}\ \boldsymbol{J}\ , \tag{18}$$

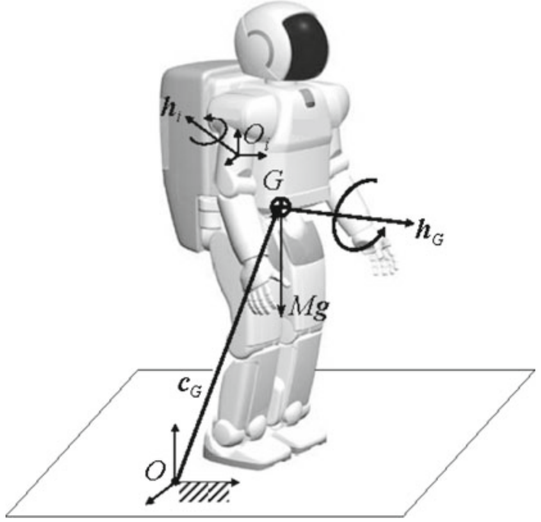

Fig. 4 Humanoid robot showing link and centroidal momentum vectors. The inertial frame is located at O and the position vector to the robot CoM, G, is given by $\boldsymbol{c}_G$. The reference frame of link i is located at O_i. The centroidal momentum $\boldsymbol{h}_G$ can be obtained from either Eqs. 13 or 15

which shows the relationship between $\boldsymbol{A}_G$ and the system inertia and Jacobian. Furthermore, time differentiation of Eq. 13 results in the following relation which forms the basis of our momentum-based balance controller presented in Sect. 9:

$$\dot{\mathbf{h}}_G = \boldsymbol{A}_G\,\ddot{\mathbf{q}} + \dot{\boldsymbol{A}}_G\,\dot{\mathbf{q}}\ . \tag{19}$$

Finally, since $\mathbf{f} = \dot{\mathbf{h}}_G$ (Newton's equations of motion) where $\mathbf{f}$ is the net external force/moment on the system (Sciavicco and Siciliano 2005), then it may be noted that $\boldsymbol{A}_G$ gives the relationship between the net external force/moment and the joint accelerations.

5 "Average spatial velocity" of a humanoid

Note that the average (linear) position of a humanoid, or any multibody system, is accepted to be the position of its CoM. Consequently, the average linear velocity is the velocity of the CoM. However, universally accepted concepts for neither an average angular position nor an average angular velocity exist for a multibody system. Essén (1993) defined an average angular velocity for a system of particles. We will advance one more step and define an "average spatial velocity" for a multibody system which includes both the linear and angular parts. However, unlike the linear velocity of the CoM, it is generally not possible to integrate the angular velocity to define an average angular position (Wieber 2005).

To determine an average spatial velocity of the CoM, $\mathbf{v}_G$, a common vector of link velocities, $\mathbf{v}^c$, will be considered so that all of the links will move as one body; that is, $\mathbf{v}^c =$

$\mathbf{X}_G\,\mathbf{v}_G$. The average spatial velocity may then be determined so as to give the same centroidal momentum as the original system. From Eqs. 10 and 15, the centroidal momentum $\boldsymbol{h}_G$ may be determined as a function of the link velocities:

$$\boldsymbol{h}_G = \mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{v}\,. \tag{20}$$

The centroidal momentum can also be determined from the common set of link velocities and thus $\mathbf{v}_G$:

$$\boldsymbol{h}_G = \mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{v}^c \;=\; \mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{X}_G\,\mathbf{v}_G\,. \tag{21}$$

Let the spatial inertia of the system at its CoM be defined as:

$$\boldsymbol{I}_G = \mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{X}_G\,. \tag{22}$$

This centroidal inertia is also called the centroidal composite rigid body inertia (CCRBI) matrix in Lee and Goswami (2007). Using the definition for the centroidal inertia, Eq. 21 results in:

$$\boldsymbol{h}_G = \boldsymbol{I}_G\,\mathbf{v}_G\,. \tag{23}$$

Finally, the average spatial velocity may be simply computed from the centroidal inertia and centroidal momentum:

$$\mathbf{v}_G = (\boldsymbol{I}_G)^{-1}\,\boldsymbol{h}_G\,. \tag{24}$$

In the above development, note that the link velocities in Eq. 20 are weighted by the link inertias so that the resulting centroidal velocity is an "inertia-weighted average spatial velocity".

The properties of the average spatial velocity may be noted by first expanding out the centroidal momentum into its angular and linear parts:

$$\boldsymbol{h}_G = \begin{bmatrix}\mathbf{k}_G\\ \mathbf{l}_G\end{bmatrix} = \mathbf{I}_G\,\mathbf{v}_G = \begin{bmatrix}\bar{\mathbf{I}}_G & \mathbf{0}\\ \mathbf{0} & M\,\mathbf{1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\omega}_G\\ \boldsymbol{v}_G\end{bmatrix}, \tag{25}$$

where $M = \sum m_i$, $\bar{\mathbf{I}}_G = \sum {}^G\bar{\mathbf{I}}_i$, and ${}^G\bar{\mathbf{I}}_i$ is the rotational inertia of the ith link projected to the CoM. Note that the cross terms, between the angular and linear velocities in the centroidal momentum equation, vanish at the CoM. As expected, the linear part of the average spatial velocity $\boldsymbol{v}_G$ is just the translational velocity of the CoM while the angular part is just the rotational velocity of an equivalent single rigid body with the same centroidal rotational inertia, $\bar{\mathbf{I}}_G$.

Equation 25 indicates that the centroidal momentum of the system may be derived from a single rigid body which has an equivalent inertia as the system, and is moving with the system's average spatial velocity. In order to examine the properties of the average spatial velocity even further, the relationship between the kinetic energy and average spatial velocity will be developed here. The kinetic energy for the system is given through the following equation:

$$T = \frac{1}{2}\,\mathbf{v}^T\,\boldsymbol{I}\,\mathbf{v}\,. \tag{26}$$

Noting in general that

$$\mathbf{v} = \mathbf{v}^c\; + \mathbf{v}' \;=\; \mathbf{X}_G\,\mathbf{v}_G\; + \,\mathbf{v}'\,, \tag{27}$$

where $\mathbf{v}'$ is the relative link velocity, an expression for the kinetic energy may be derived as a function of the average spatial velocity. Substituting Eq. 27 into 26 gives:

$$\begin{aligned} T = {} & \frac{1}{2}\,\mathbf{v}_G^T\,\mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{X}_G\,\mathbf{v}_G \;+\; \frac{1}{2}\,(\mathbf{v}')^T\,\boldsymbol{I}\,\mathbf{X}_G\,\mathbf{v}_G \\ & +\frac{1}{2}\,\mathbf{v}_G^T\,\mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{v}' \;+\; \frac{1}{2}\,(\mathbf{v}')^T\,\boldsymbol{I}\,\mathbf{v}'\,. \end{aligned} \tag{28}$$

The middle two terms in this expression may be eliminated by noting the following from Eqs. 20 and 21

$$\mathbf{X}_G^T\,\boldsymbol{I}\,(\mathbf{v}\;-\;\mathbf{v}^c) \;=\; \mathbf{X}_G^T\,\boldsymbol{I}\,\mathbf{v}' \;=\; \mathbf{0}\,. \tag{29}$$

That is, the centroidal momentum resulting from relative motion is zero. Using Eq. 22, the final expression for the kinetic energy then is:

$$T = \frac{1}{2}\,\mathbf{v}_G^T\,\boldsymbol{I}_G\,\mathbf{v}_G \;+\; \frac{1}{2}\,(\mathbf{v}')^T\,\boldsymbol{I}\,\mathbf{v}'\,. \tag{30}$$

Note the differences in the dimensions of the velocities in this equation. In particular, the average spatial velocity vector $\mathbf{v}_G$ is 6×1 while the relative velocity vector $\mathbf{v}'$ is $6N \times 1$.

Unlike centroidal momentum, and as that goes potential energy, the kinetic energy in the system cannot be completely characterized using only centroidal quantities. In particular, a second term appears in the equation which is related to the relative motion between the links. However, note that the kinetic energy is minimum when there is no relative motion; that is, $\mathbf{v}' = \mathbf{0}$. The minimum kinetic energy, then, is a direct function of centroidal quantities, namely the average spatial velocity and CCRBI.

Some further elaboration on the terms in the above equation for kinetic energy may be helpful. Noting the expressions for $\boldsymbol{I}_G$ and $\mathbf{v}_G$ in Eq. 25, the first term in the above equation gives the rotational and translational kinetic energy for an equivalent single rigid body. However, the second term in Eq. 30 is needed in the case of relative motion for a system of rigid bodies. As an example, if the system included two equal-size bodies with rotational and translational movement in opposite directions, then the average spatial velocity would be zero. However, the net kinetic energy would not be zero, and the second term correctly accounts for the kinetic energy due to the relative motion (relative to the average *spatial* velocity).

6 Transformation diagram

We have found it useful to pictorially capture the relations between the CMM and other matrices using what we call the

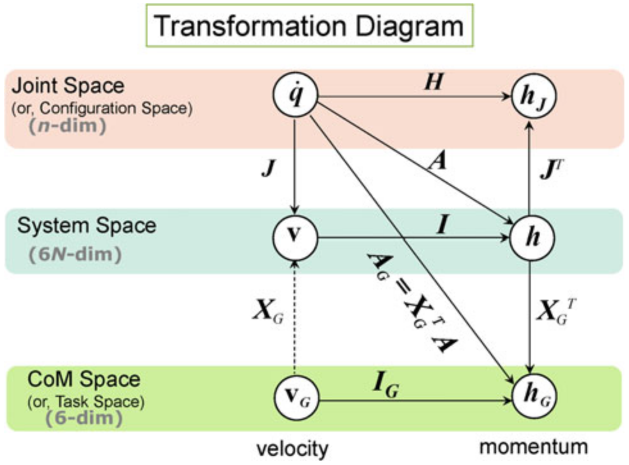

Fig. 5 Transformation diagram showing the relations among the velocities and momenta of a robot. These vector quantities can be expressed in joint space, system space, or the CoM space of the robot. The matrices representing the linear transformations between velocities and momenta in different spaces are also shown in this diagram. The *dashed line* at the lower *left* of the diagram represents a minimum kinetic energy transformation, which is not a general transformation, as discussed in the text

transformation diagram. As shown in Fig. 5, the transformation diagram represents the relations among the velocity and momentum vectors, as well as the associated matrices, within and across three spaces. The spaces are the n-dimensional joint space or configuration space, the $6N$-dimensional system space, and the 6-dimensional CoM space. The joint space contains the robot's generalized coordinates, the system space hosts the motion components of each rigid body link of the robot, and the CoM space defines a coordinate frame located at the robot CoM which is instantaneously oriented identical to the inertial reference frame (Frame 0). The CoM space is an example of what is more commonly known as the task space or the operational space (Khatib 1987).

The transformation diagram contains three rows which correspond, from top to bottom, to the joint space, the system space and the CoM space, respectively. Each row contains the velocity and the momentum vectors corresponding to that space. The mapping between any two vectors, within the same space or across two different spaces, is given by a matrix and shown with an arrow.

Three types of transformations, which we call horizontal, vertical and diagonal, are depicted in the transformation diagram. A *horizontal transformation* takes place within the same space (i.e., within the same row) and it maps a velocity vector to a momentum vector through a square inertia matrix. Equations 10 and 23 are examples of horizontal transformation in the system space and the CoM space, respectively. The mapping from $\dot{\mathbf{q}}$ to $\boldsymbol{h}_J$, given by

$$\boldsymbol{h}_J = \boldsymbol{H}\,\dot{\mathbf{q}}\,, \tag{31}$$

is the horizontal transformation within the joint space. The matrix $\boldsymbol{H}$ is the joint-space inertia matrix, which is well-known from the standard equations of motion of a robot[2]. The generalized momenta $\boldsymbol{h}_J$, which is also called the canonical momenta (Naudet 2005), has not been exploited much for humanoid analysis and control, but has been used in collision detection (De Luca et al. 2006) and space robotics (Nenchev et al. 1992).

A *vertical transformation* is given either by a non-square Jacobian or a spatial transformation matrix; it relates two velocity vectors or two momentum vectors, Naturally, a vertical transformation takes place between two different spaces. The vertical transformations corresponding to Eqs. 6 and 15 are shown, respectively, at the top left and bottom right in Fig. 5. Finally, a *diagonal transformation* relates two dissimilar vectors between two different spaces. The matrices $\boldsymbol{A}$ and $\boldsymbol{A}_G$, shown, respectively, in Eqs. 14 and 13 fall in this category.

Using the transformation diagram we can compare and contrast between $\boldsymbol{H}$ and $\boldsymbol{A}_G$, which was one of our earliest motivations behind this work. While $\boldsymbol{H}$ is an $n \times n$ square inertia matrix representing a velocity → momentum mapping within the joint space, the $6 \times n$ matrix $\boldsymbol{A}_G$ maps the joint space velocity to the CoM space momentum. $\boldsymbol{H}$ and $\boldsymbol{A}_G$ are both related to the matrix $\boldsymbol{A}$ which maps the joint space velocity to the system space momentum. The former relationship can be obtained from the transformation diagram:

$$\boldsymbol{H} = \boldsymbol{J}^T \boldsymbol{A} \tag{32}$$

and the latter is given by Eq. 18.

Figure 5 contains two vertical transformations, at top right and bottom left, which are given by

$$\boldsymbol{h}_J = \boldsymbol{J}^T \boldsymbol{h} \tag{33}$$

and

$$\mathbf{v} = \mathbf{v}^c = \mathbf{X}_G\,\mathbf{v}_G\,. \tag{34}$$

Equation 33 can be obtained by equating the expressions for kinetic energy which are independently derived in the joint space and the system space. This is possible because each space represents a complete description of the motion of the robot.

Equation 34 can be thought of as the collection of all link velocities, each of the form $\mathbf{v}_i^c = {}^i\mathbf{X}_G\,\mathbf{v}_G$. This equation describes the system moving as a single rigid body with the

[2] The equations of motion for an n-dof robot can be expressed as:

$$\boldsymbol{\tau} = \boldsymbol{H}(\boldsymbol{q})\,\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\,\dot{\boldsymbol{q}} + \boldsymbol{\tau}_g(\boldsymbol{q})\,,$$

where $\boldsymbol{H}$ is the $n \times n$ symmetric, positive-definite joint-space inertia matrix, $\boldsymbol{C}$ is an $n \times n$ matrix such that $\boldsymbol{C}\,\dot{\boldsymbol{q}}$ is the vector of Coriolis and centrifugal terms (collectively known as *velocity product* terms), and $\boldsymbol{\tau}_g$ is the vector of gravity terms.

CCRBI, $\boldsymbol{I}_G$, and possesses minimum kinetic energy. Because this is a particular solution satisfying a specific condition, and not a general solution, we show this transformation with a dashed line in Fig. 5.

Overall, the transformation diagram pictorially summarizes several of the mathematical relationships between the quantities developed in this paper.

7 Efficient recursive algorithm

In order to use the CMM in real-time control of a humanoid robot, it is important to have an efficient algorithm to compute it. In this section, an efficient recursive algorithm will be developed for the matrix using spatial notation. The use of spatial notation results in a particularly compact form for the algorithm. Further, efficient realization of the 6D spatial operations is provided in Featherstone and Orin (2008) so that the computation essentially reduces to 3D implementation of the angular and linear parts of the matrix.

In addition to efficient computation of the CMM, the algorithm also computes the centroidal momentum, the centroidal composite rigid body inertia (CCRBI), and average spatial velocity with little additional computation required. The main part of the algorithm is the recursive computation of the CCRBI, and most other quantities are derived from it. Much of the efficiency of the algorithm results from expressing the CCRBI for subtrees of links, in local coordinates.

To develop the algorithm, note that Eq. 13 can be written in the following form:

$$\boldsymbol{h}_G = \sum_{i=1}^{N} (\boldsymbol{A}_G)_i \ \dot{\mathbf{q}}_i \ , \tag{35}$$

where $(\boldsymbol{A}_G)_i$ refers to the ith set of n_i columns of $\boldsymbol{A}_G$ that are associated with joint i. That is, the centroidal momentum can be computed by taking the individual joint motion contributions and summing them.

Each individual joint contribution results when the joint rate is set as $\dot{\mathbf{q}} = \left[0\,0\ldots\dot{\mathbf{q}}_i^T\ldots 0\right]^T$. In this case, there is no motion in the humanoid except at joint i. This separates the humanoid into two separate composite rigid bodies (CRBs) connected at joint i, and the dynamics of the humanoid are much simpler for this case. The spatial velocity of the composite rigid body which is in motion $\mathbf{v}_i^C$ is:

$$\mathbf{v}_i^C = \mathbf{v}_i = \boldsymbol{\Phi}_i \ \dot{\mathbf{q}}_i \ . \tag{36}$$

Note that $\mathbf{v}_i^C$ is determined at the origin of the ith coordinate frame.

The contribution to the centroidal momentum due to motion at joint i, $(\boldsymbol{h}_G)_i$, can be computed from the spatial momentum of the ith CRB, $\boldsymbol{h}_i^C$, as follows:

$$(\boldsymbol{h}_G)_i = {}^i\mathbf{X}_G^T \ \boldsymbol{h}_i^C \ . \tag{37}$$

With the simplified dynamics for a single CRB, $\boldsymbol{h}_i^C$ may be determined from:

$$\boldsymbol{h}_i^C = \boldsymbol{I}_i^C \ \mathbf{v}_i^C = \boldsymbol{I}_i^C \ \boldsymbol{\Phi}_i \ \dot{\mathbf{q}}_i \ , \tag{38}$$

where $\boldsymbol{I}_i^C$ is the spatial inertial for the ith CRB, and may be computed by combining the spatial inertias for all the links in the subtree rooted at link i. Substituting $\boldsymbol{h}_i^C$ from Eq. 38 into Eq. 37, and noting the similarity of the resulting equation with Eq. 35, the following expression for $(\boldsymbol{A}_G)_i$ is developed:

$$(\boldsymbol{A}_G)_i = {}^i\mathbf{X}_G^T \ \boldsymbol{I}_i^C \ \boldsymbol{\Phi}_i \ . \tag{39}$$

Equations 35 and 39 provide the basis for the efficient recursive algorithm for $\boldsymbol{h}_G$ and $\boldsymbol{A}_G$ given in Table 1. The algorithm consists of two main recursions. After the algorithm initializes the inertia for the ith CRB, $\boldsymbol{I}_i^C$, the first recursion follows and is an inward recursion from the leaf links in the tree to the floating base to compute $\boldsymbol{I}_i^C$. After that is an outward recursion to compute the spatial transform ${}^i\mathbf{X}_G$, the components of the CMM, $\boldsymbol{A}_G$, and centroidal momentum $\boldsymbol{h}_G$.

The equation to compute the spatial inertia for the ith CRB is given as:

$$\mathbf{I}_{p(i)}^C = \mathbf{I}_{p(i)}^C + {}^i\mathbf{X}_{p(i)}^T \ \mathbf{I}_i^C \ {}^i\mathbf{X}_{p(i)} \ . \tag{40}$$

Using spatial notation, a simple congruence transform using ${}^i\mathbf{X}_{p(i)}$ is all that is needed to transform inertias from one coordinate system to another (Featherstone and Orin 2008). Also, Eq. 40 is the same equation that is used in the composite-rigid-body algorithm (CRBA) for computing the joint-space inertia matrix $\boldsymbol{H}$ in Featherstone and Orin (2008). The CRBA was first developed by Walker and Orin (1982) and is one of the most efficient algorithms for computing $\boldsymbol{H}$.

Note that ${}^0\mathbf{X}_G$ is needed in the last recursion for $i = 1$. Since the coordinate axes of frames 0 and G are set to be parallel, and noting the form for $\mathbf{X}$ as given in Eq. 2, then

$${}^0\mathbf{X}_G = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{S}({}^G\mathbf{p}_0)^T & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{S}({}^0\mathbf{p}_G) & \mathbf{1} \end{bmatrix} . \tag{41}$$

The vector ${}^0\mathbf{p}_G$ is just the position vector from the reference frame 0 to the centroid of the system; i.e., ${}^0\mathbf{p}_G = \mathbf{c}_G$, and it may be easily derived from $\mathbf{I}_0^C$. Noting the form for the spatial inertia given in Eq. 4, the CCRBI $\mathbf{I}_0^C$ is given as:

$$\mathbf{I}_0^C = \begin{bmatrix} \bar{\mathbf{I}}_0^C & M\,\mathbf{S}(\mathbf{c}_G) \\ M\mathbf{S}(\mathbf{c}_G)^T & M\,\mathbf{1} \end{bmatrix} . \tag{42}$$

In fact, in the efficient 3D realization of the 6D spatial operations (Featherstone and Orin 2008) for the algorithm, M, $(M\,\mathbf{c}_G)$, and $\bar{\mathbf{I}}_0^C$ are stored in place of $\mathbf{I}_0^C$. So $\mathbf{c}_G$ may be easily computed by simply dividing the first moment $M\mathbf{c}_G$ by the total mass M.

Table 1 Recursive algorithm for computing the CMM, centroidal momentum, centroidal composite rigid body inertia (CCRBI), and average spatial velocity

```
inputs: $model$, $\boldsymbol{\Phi}_i$, ${}^i\mathbf{X}_{p(i)}$, $\dot{\mathbf{q}}_i$
output: $\boldsymbol{A}_G$, $\boldsymbol{h}_G$, $\mathbf{I}_G$, $\mathbf{v}_G$
$model\ data$: $N$, $p(i)$, $\mathbf{I}_i$
$\mathbf{I}_0^C = \mathbf{0}$
for $i = 1$ to $N$ do
    $\mathbf{I}_i^C = \mathbf{I}_i$
end
for $i = N$ to $1$ do
    $\mathbf{I}_{p(i)}^C = \mathbf{I}_{p(i)}^C + {}^i\mathbf{X}_{p(i)}^T \, \mathbf{I}_i^C \, {}^i\mathbf{X}_{p(i)}$
end
$\boldsymbol{h}_G = \mathbf{0}$
for $i = 1$ to $N$ do
    ${}^i\mathbf{X}_G = {}^i\mathbf{X}_{p(i)} \, {}^{p(i)}\mathbf{X}_G$
    $(\boldsymbol{A}_G)_i = {}^i\mathbf{X}_G^T \, \mathbf{I}_i^C \, \boldsymbol{\Phi}_i$
    $\boldsymbol{h}_G = \boldsymbol{h}_G + (\boldsymbol{A}_G)_i \, \dot{\mathbf{q}}_i$
end
$\mathbf{I}_G = {}^0\mathbf{X}_G^T \, \mathbf{I}_0^C \, {}^0\mathbf{X}_G$
$\mathbf{v}_G = (\mathbf{I}_G)^{-1} \, \boldsymbol{h}_G$
```

The computation in the algorithm in Table 1 grows linearly with the number of joints so that the complexity is $O(N)$. Furthermore, the spatial operations in the table can be implemented with optimum efficiency using the equivalent 3D operations given in Featherstone and Orin (2008). Table 2.1 in Featherstone and Orin (2008) provides the 3D representations for $\mathbf{X}$, $\mathbf{I}$, and other 6D spatial quantities, so that the following operations in the algorithm are efficiently implemented: $\mathbf{X}^T\mathbf{I}\mathbf{X}$, $\mathbf{X}_1\mathbf{X}_2$, and $\mathbf{X}^T\mathbf{I}\boldsymbol{\Phi}$.

Note that the equivalent 3D operations for computing the CRB inertias are similar to that given by Kajita et al. (2003). However, considerable savings in computation over that of Kajita et al. (2003) is provided here by expressing the CRB inertias in local coordinates. In particular, for a humanoid with a number of simple revolute joints, the (equivalent 3D) congruence computation needed in Eq. 40 requires many fewer arithmetic operations than the case when the inertias, and thus transforms, are relative to ground-fixed coordinates (McMillan and Orin 1995). The computation of $(\boldsymbol{A}_G)_i$ in Eq. 39 is also much simpler for the case of a revolute joint since $\boldsymbol{\Phi}_i = [0\,0\,1\,0\,0\,0]^T$ when expressed in local coordinates.

The computation time for calculating the CMM, centroidal momentum, CCRBI, and average spatial velocity has been tested on an HP Intel Dual Core (2.8 GHz) PC with 3 Gbytes of RAM. For a humanoid model with $N = 21$ links and $n = 26$ degrees of freedom (including the 6 DOFs of the floating base), the PC was able to compute the centroidal dynamics according to the algorithm in Table 1 at a rate of 3.0 KHz. This is comparable to an inverse dynamics algorithm based on the recursive Newton-Euler Algorithm (Bin Hammam et al. 2010; Featherstone and Orin 2008) which ran at 3.6 KHz. This demonstrates the feasibility of using centroidal dynamics for real-time control.

8 Constraints for single or double support

With the humanoid in single or double support, there are constraints imposed on the joint velocities, including the spatial velocity of the floating base link (typically the torso). In this section, the constrained CMM, $\mathbf{A}_G^c$, is derived for the constrained system.

The constraints on the velocities can be expressed in the form of a linear equation (Featherstone and Orin 2008),

$$\mathbf{L}\,\dot{\mathbf{q}} = \mathbf{0}\,, \tag{43}$$

where $\mathbf{L}$ is an $n^c \times n$ matrix, n is the number of degrees of freedom in the unconstrained system, and n^c is the number of constraints. Note that the right side of the equation may, for instance, represent the zero velocity of a foot relative to the ground. However, if the relative velocity at the constraint is nonzero, this can easily be incorporated into the development.

To proceed, extract m linearly independent rows from $\mathbf{L}$, where $m = \text{rank}(\mathbf{L})$, and partition the joint velocities into $n - m$ primary variables, $\dot{\mathbf{q}}_P$ and m secondary variables, $\dot{\mathbf{q}}_S$:

$$\dot{\mathbf{q}} = \mathbf{Q}\begin{bmatrix}\dot{\mathbf{q}}_S\\ \dot{\mathbf{q}}_P\end{bmatrix}, \tag{44}$$

where $\mathbf{Q}$ is an $n \times n$ permutation matrix. Equation 43 may be written in terms of these variables:

$$[\mathbf{L}_S\,\mathbf{L}_P]\begin{bmatrix}\dot{\mathbf{q}}_S\\ \dot{\mathbf{q}}_P\end{bmatrix} = \mathbf{0}\,, \tag{45}$$

or

$$\mathbf{L}_S\,\dot{\mathbf{q}}_S + \mathbf{L}_P\,\dot{\mathbf{q}}_P = \mathbf{0}\,, \tag{46}$$

where $\mathbf{L}_S$ is an $m \times m$ matrix formed by extracting m independent rows and columns from $\mathbf{L}$ and $\mathbf{L}_P$ is $m \times (n - m)$. Efficient procedures for choosing the primary variables (generalized coordinates) and secondary variables and associated constraint matrices are given in Nakamura and Yamane (2000). Since $\mathbf{L}_S$ is always invertible, the secondary joint velocities may be expressed in terms of the primary velocities as:

$$\dot{\mathbf{q}}_S = -\mathbf{L}_S^{-1}\,\mathbf{L}_P\,\dot{\mathbf{q}}_P\,. \tag{47}$$

Using Eqs. 13 and 44, the centroidal momentum may be expressed as a function of the primary and secondary joint velocity variables:

$$\boldsymbol{h}_G = \boldsymbol{A}_G\,\dot{\mathbf{q}} = \boldsymbol{A}_G\,\mathbf{Q}\begin{bmatrix}\dot{\mathbf{q}}_S\\ \dot{\mathbf{q}}_P\end{bmatrix} = \mathbf{A}_{GS}\,\dot{\mathbf{q}}_S + \mathbf{A}_{GP}\,\dot{\mathbf{q}}_P\,, \tag{48}$$

where $\mathbf{A}_{GS}$ and $\mathbf{A}_{GP}$ are $6 \times m$ and $6 \times (n - m)$ matrices, respectively, that are formed from the columns of $\mathbf{A}_G$. Substituting $\dot{\mathbf{q}}_S$ from Eq. 47 into this equation gives:

$$\boldsymbol{h}_G = \left(\mathbf{A}_{GP} - \mathbf{A}_{GS}\,\mathbf{L}_S^{-1}\,\mathbf{L}_P\right)\,\dot{\mathbf{q}}_P = \mathbf{A}_G^c\,\dot{\mathbf{q}}_P\ , \tag{49}$$

which gives an expression for $\mathbf{A}_G^c$:

$$\mathbf{A}_G^c = \mathbf{A}_{GP} - \mathbf{A}_{GS}\,\mathbf{L}_S^{-1}\,\mathbf{L}_P\ . \tag{50}$$

As an example, consider the case where the velocity of the kth foot $\mathbf{v}_{F,k}$ for a 6 degree-of-freedom leg is zero (single support):

$$\mathbf{v}_{F,k} = \mathbf{J}_{F,k}\,\dot{\mathbf{q}} = \mathbf{0}\ , \tag{51}$$

where $\mathbf{J}_{F,k}$ is the Jacobian for the foot. If the secondary joint velocities are chosen as the kth leg joint velocities, then

$$\dot{\mathbf{q}}_S = \begin{bmatrix} \dot{\mathbf{q}}_{j,k} \\ \dot{\mathbf{q}}_{j+1,k} \\ \vdots \\ \dot{\mathbf{q}}_{j+5,k} \end{bmatrix}\ , \tag{52}$$

where $\dot{\mathbf{q}}_{j,k}$ is the velocity of the first degree of freedom (DoF) of leg k. The primary and secondary parts of the constraint matrix $\mathbf{L}$ are then given as:

$$\mathbf{L}_P = \left[\mathbf{J}_{F,k}^1\,\mathbf{0}\cdots\mathbf{0}\right]\ , \tag{53}$$

and

$$\mathbf{L}_S = \left[\mathbf{J}_{F,k}^j\,\mathbf{J}_{F,k}^{j+1}\cdots\mathbf{J}_{F,k}^{j+5}\right]\ , \tag{54}$$

where $\mathbf{J}_{F,k}^1$ is the 6×6 Jacobian matrix giving the contribution of the spatial velocity of the floating base link (link 1) to the foot velocity, and $\mathbf{J}_{F,k}^j$ is the column of the Jacobian for foot k which is associated with the first degree of freedom in leg k, etc.. The column of the Jacobian for joint j may be computed through the following equation:

$$\mathbf{J}_{F,k}^j = {}^{F,k}\mathbf{X}_j\ \boldsymbol{\Phi}_j\ , \tag{55}$$

with the computations of the other joints following the same form. Note that the constraints for double support are developed in a similar manner. Finally, singularities in the legs are readily managed through the general algorithms presented in Nakamura and Yamane (2000) for deriving generalized coordinates for closed kinematic chains.

9 Example: centroidal momentum for postural balance control

Equation 13 is useful for computing joint velocities when the desired centroidal momentum is given. Since postural balance can be defined in terms of centroidal momentum, Eq. 13 can be effectively used to design a whole body balance controller. Thus we developed a postural balance controller using the CMM as reported in Lee and Goswami (2012). Here we present a brief description of the balance controller and apply it to a new example of lateral balance control. This example highlights the usefulness of the CMM when the dynamic action of multiple limbs is needed for balance. More details of the controller can be found in Lee and Goswami (2012).

As a control policy, we define the desired rate of change of centroidal momentum $\dot{\mathbf{h}}_{G,d}$ such that it realizes the desired position and velocity of the CoM, $\mathbf{c}_{G,d}$ and $\mathbf{v}_{G,d}$, and the desired centroidal angular momentum $\mathbf{k}_{G,d}$ as follows:

$$\dot{\mathbf{h}}_{G,d} = \begin{bmatrix} \dot{\mathbf{k}}_{G,d} \\ \dot{\mathbf{l}}_{G,d} \end{bmatrix}\ , \tag{56}$$

$$\dot{\mathbf{l}}_{G,d}/M = \boldsymbol{\Gamma}_{11}\,(\mathbf{v}_{G,d} - \mathbf{v}_G) + \boldsymbol{\Gamma}_{12}(\mathbf{c}_{G,d} - \mathbf{c}_G)\ , \tag{57}$$

$$\dot{\mathbf{k}}_{G,d} = \boldsymbol{\Gamma}_{21}\,(\mathbf{k}_{G,d} - \mathbf{k}_G)\ , \tag{58}$$

where M is the total mass of the robot and $\boldsymbol{\Gamma}_{ij}$ is a 3×3 diagonal matrix representing feedback gain parameters. Note that unlike in Eq. 57 we do not have angular orientation feedback in Eq. 58 because there is no special orientation corresponding to angular momentum.

The desired momentum rate change can only be realized by controlling the net external force and moment. However, due to the unilateral contact constraint between the humanoid robot and the ground, a limitation exists on the range of feasible external forces and moments that can be generated by the robot, and thus the $\dot{\mathbf{l}}_{G,d}$ and $\dot{\mathbf{k}}_{G,d}$ computed by Eqs. 57 and 58 may not be physically realizable. Therefore, for the balance controller we compute the admissible value of the centroidal momentum rate change that is close to the desired value while physically realizable, hence dubbed the *admissible* linear and angular momenta rate change, denoted by $\dot{\mathbf{l}}_{G,a}$ and $\dot{\mathbf{k}}_{G,a}$, respectively. While there can be many methods to determine $\dot{\mathbf{l}}_{G,a}$ and $\dot{\mathbf{k}}_{G,a}$, we chose to give a higher priority to linear momentum over angular momentum: we set $\dot{\mathbf{l}}_{G,a}$ first to be as close as possible to $\dot{\mathbf{l}}_{G,d}$ and then resolve $\dot{\mathbf{k}}_{G,a}$ next such that $\dot{\mathbf{k}}_{G,a}$ is physically realizable given $\dot{\mathbf{l}}_{G,a}$ (See Lee and Goswami 2012 for details.)

Next, we need to compute the joint acceleration vector $\ddot{\mathbf{q}}$ which will realize the admissible momentum rate change. For this, we determine $\ddot{\mathbf{q}}_P$, the acceleration of the primary variables, from the differentiation of Eq. 49 as

$$\mathbf{A}_G^c\ddot{\mathbf{q}}_P = \dot{\mathbf{h}}_{G,a} - \dot{\mathbf{A}}_G^c\,\dot{\mathbf{q}}_P\ , \tag{59}$$

where $\dot{\mathbf{h}}_{G,a} = [\ \dot{\mathbf{l}}_{G,a}^T, \dot{\mathbf{k}}_{G,a}^T\ ]^T$. Note that, from Eq. 59, $\dot{\mathbf{A}}_G^c\,\dot{\mathbf{q}}_P$ is equivalent to $\dot{\mathbf{h}}_{G,a}$ when $\ddot{\mathbf{q}}_P$ is zero. Since $\ddot{\mathbf{q}}_S$ must satisfy

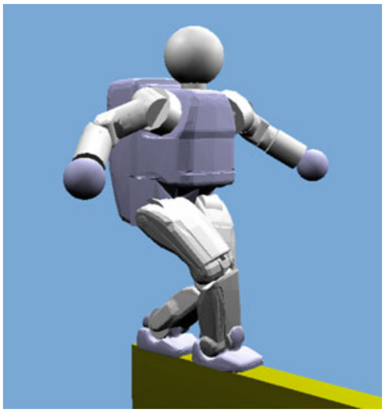

Fig. 6 The centroidal momentum-based controller can maintain postural balance of the humanoid robot under various disturbance forces

$\ddot{\mathbf{q}}_S = -\mathbf{L}_S^{-1}\,(\dot{\mathbf{L}}_P\,\dot{\mathbf{q}}_P + \dot{\mathbf{L}}_S\,\dot{\mathbf{q}}_S)$ when $\ddot{\mathbf{q}}_P$ is zero[3], $\dot{\mathbf{A}}_G^c\,\dot{\mathbf{q}}_P$ is computed as the rate of change of angular and linear momenta ($\dot{\mathbf{h}}$) when $\ddot{\mathbf{q}}_P = 0$ and $\ddot{\mathbf{q}}_S$ is set as above, given the current state $\mathbf{q}$ and $\dot{\mathbf{q}}$.

For typical humanoid robots, $\ddot{\mathbf{q}}_P$ has a higher dimension compared to that of $\dot{\mathbf{h}}$. Consequently, infinitely many solutions of $\ddot{\mathbf{q}}_P$ exist. This allows us to pursue an optimal solution under some additional objectives.

If we have desired joint accelerations $\ddot{\mathbf{q}}_{U,d}$ for the upper body joints $\mathbf{q}_U$ ($\subset \mathbf{q}_P$) that are either given from a predefined motion or are generated in real time to achieve a certain task, one way to determine $\ddot{\mathbf{q}}_P$ (hence $\ddot{\mathbf{q}}$) is to minimize the following objective function:

$$w||\dot{\mathbf{h}}_{G,a} - \mathbf{A}_G^c\,\ddot{\mathbf{q}}_P - \dot{\mathbf{A}}_G^c\,\dot{\mathbf{q}}_P|| + (1-w)||\ddot{\mathbf{q}}_{U,d} - \ddot{\mathbf{q}}_U|| \quad (60)$$

where the first term tries to achieve $\dot{\mathbf{h}}_{G,a}$ while the second term tries to follow the desired upper body motion. In our experiment, $\ddot{\mathbf{q}}_{U,d}$ is determined so as to maintain the desired pose as shown in Fig. 6. w controls the weighting between the two objectives.

We tested the momentum-based balance controller by simulating a humanoid robot model as seen in Fig. 6. In the simulation experiment the robot is subjected to a push from the lateral direction while standing on a narrow support, which is even slightly narrower than the width the robot's feet. In this environment, the robot must rotate its upper body in order to maintain balance, and our controller based on the CMM creates such a whole body motion in which the whole body segments including the trunk and arms are engaged to create the necessary admissible momentum rate change $\dot{\mathbf{h}}_{G,a}$. The top row of Fig. 7 shows a series of snapshots illustrating this when the robot is subjected to an external push (115 N, 0.1 s) which is applied at the robot's CoM in the lateral direction from the robot's right side.

The resulting motion of the robot is similar to that of a human rotating the trunk and the arms in the direction of the push to maintain balance. Note that the arms of the robot move in an asymmetric fashion to counteract the destabilizing effect of the disturbance force. The right arm of the robot (seen at left in the figure) is raised higher, while the left arm is brought closer to the body. This is reminiscent of the windmilling effect of the arms, which is commonly seen in the sagittal plane. The dynamic motion of the arms is not separately created but is the natural outcome of the use of the CMM. The final motion is the outcome of a number of competing requirements such as the CoP motion, the linear and angular momenta, and the joint angle trajectories. It is not always possible to explain them through simple intuitive arguments.

Figure 7a–f show the trajectories of some important physical quantities which are recorded during this test. The trajectories of the CoM, linear momentum, and angular momentum in Figs 7a–c show that they return to the desired values rather smoothly after the perturbation. The trajectory of the average angular velocity of the robot, $\boldsymbol{\omega}_G$, given in Eq. 25 is shown in Fig. 7d. Since the rotational inertia of the robot does not change significantly, the pattern of the $\boldsymbol{\omega}_G$ trajectory is similar to that of the angular momentum.

Figure 7e shows the measured foot CoP, which is calculated using contact force information during the simulation. The controller controls the linear and angular momenta rate changes such that the CoP is kept inside the safety margin. We set the CoP safety margin to be well inside the width of the support beam.

Figure 7f compares the trajectories of the trunk bend angles when the arms are locked with respect to the trunk (dotted line) and when they are independently engaged by the balance controller (solid line). We can observe that the robot's trunk has to bend more (about 10°) to make up for the fact that the arms cannot move.

The robot successfully survived the perturbation, which is confirmed from Fig. 7b in which the linear momentum is seen to return to zero after about 17 s. Additionally, the robot returns to its original posture due to the effect of the second term in Eq. 60.

For this simulation we use the following parameters: $\boldsymbol{\Gamma}_{11} = \text{diag}\{40, 20, 40\}/M$ and $\boldsymbol{\Gamma}_{12} = \text{diag}\{8, 3, 8\}/M$ in Eq. 57, $\boldsymbol{\Gamma}_{21} = \text{diag}\{20, 20, 20\}$ in Eq. 58. In our experiment, w can vary between 0 and 0.999 depending on the distance of the CoP from the perimeter of the base support. The closer the CoP to the support base perimeter, the higher is the value of w.

[3] For a stationary support foot, $\ddot{\mathbf{q}}_S$ should satisfy the constraint equation following from the differentiation of Eq. 46, *i.e.*,

$$\begin{aligned}\ddot{\mathbf{q}}_S &= -\mathbf{L}_S^{-1}\,(\mathbf{L}_P\,\ddot{\mathbf{q}}_P + \dot{\mathbf{L}}_P\,\dot{\mathbf{q}}_P + \dot{\mathbf{L}}_S\,\dot{\mathbf{q}}_S)\\ &= -\mathbf{L}_S^{-1}\,(\dot{\mathbf{L}}_P\,\dot{\mathbf{q}}_P + \dot{\mathbf{L}}_S\,\dot{\mathbf{q}}_S)\,.\end{aligned}$$

(Note that $\dot{\mathbf{q}}_P$ and $\dot{\mathbf{q}}_S$ are given by the system state.)

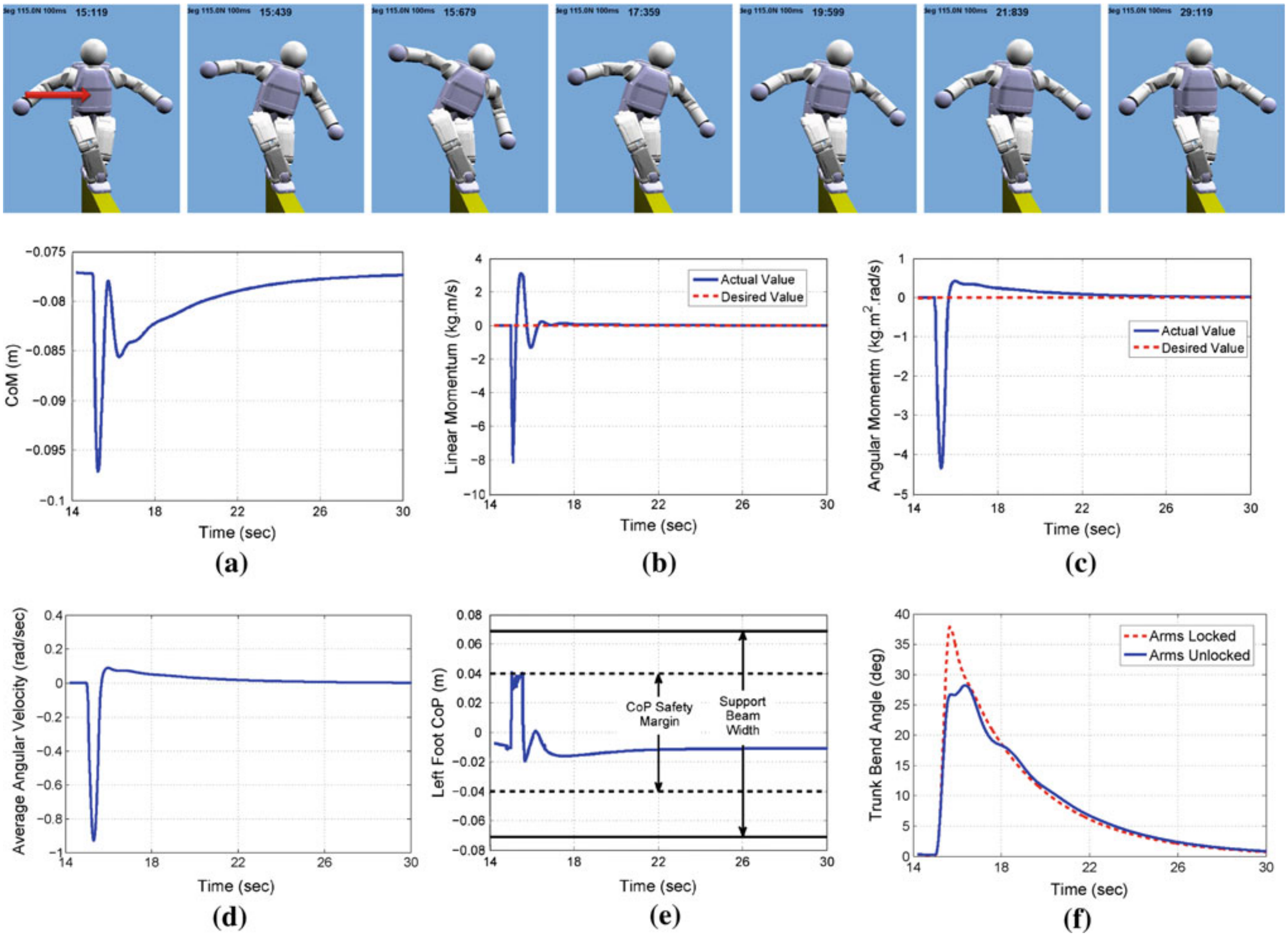

Fig. 7 The robot is standing on a narrow support. *Top row*: Given a lateral push (115 N, 0.1 s) from its right side around 15 s, the postural balance controller controls both linear and angular momentum, by moving the whole body. Especially the robot generates necessary angular momentum by rotating both the trunk and the arms, which is comparable to human's balance control behavior. **a**–**e** Trajectories of important physical properties of the experiment. **f** Trajectories of trunk bend angle when the arms are locked to the trunk (*dotted line*) and when they can rotate (*solid line*). **a** CoM (lateral direction) **b** linear momentum (lateral direction) **c** angular momentum (coronal plane) **d** average angular velocity (coronal plane) **e** left foot CoP (lateral direction) **f** bend angle

10 Conclusions and future work

In this paper we have derived expressions for the centroidal momentum of a humanoid robot using spatial notation. We have studied the structure, properties, and computational schemes of the CMM, which is a local linear function that projects the robot joint rates to the centroidal momentum. We have shown that this matrix is the product of a spatial transformation matrix, an inertia matrix and a Jacobian matrix.

We also introduced the new concept of "average spatial velocity" of the humanoid that encompasses both linear and angular components and results in a novel decomposition of the kinetic energy. It has a number of interesting properties that have been identified which should make it useful in humanoid motion analysis and control.

We have developed the transformation diagram to pictorially summarize the relationships between the velocity and momenta variables of a robot—in the joint space, the CoM space, as well as in the system space. This diagram is also helpful in identifying relationships between the transformation matrices.

We have also developed very efficient $O(N)$ algorithms, expressed in compact form using spatial notation, for computing the CMM, centroidal momentum, centroidal inertia, and average spatial velocity. Finally, as a practical use of centroidal dynamics we described a momentum-based balance controller that directly employs the CMM. The balance controller is capable of balancing the robot on non-level and non-stationary ground, including dissimilar slopes and velocity requirements on the two feet (Lee and Goswami 2012).

For a deeper understanding of the momentum properties of a robot and for practical application, several questions need to be addressed. The average angular velocity of the humanoid has been defined here, and an efficient algorithm has been developed to compute it. As an extension of this work, it may be useful to explore its use in real-time control. Also, while servoing on the position of the CoM can be useful, there is no analogous definition for "average orientation," and this needs to be investigated. One possible approach may be to allow the CoM frame to rotate with the average angular velocity of the humanoid.

A second future direction may be to apply the centroidal dynamics concepts, developed in this paper, to a broader range of mobile robotics systems. Of particular importance will be the availability of torque-controlled robots, as opposed to position-controlled robots, so that the dynamics of these systems can be adequately controlled. In any event, hopefully the concepts and computational tools developed in this paper will provide the foundation for a wide range of systems and applications.

Finally, let us point out that we continue to successfully apply CMM in different humanoid applications. Wensing and Orin (2013) report two examples, dynamic kick and dynamic jump, to show the usefulness of the CMM. In these examples the control of the system's centroidal angular momentum leads to natural-looking emergent whole-body behaviors, such as arm-swing, that are not specified by the designer. Lee and Goswami (2012) present several examples of humanoid control based on the CMM, where the robot can recover from push and can maintain its balance on a non-stationary platform that can translate and rotate.

Acknowledgments The authors gratefully thank Ghassan Bin Hammam for testing the recursive centroidal dynamics algorithm on a PC. Support for this work for David Orin was provided in part by Grant No. CNS-0960061 from NSF with a subaward to The Ohio State University. S.-H. Lee was partly supported by the Global Frontier R&D Program, NRF (NRF-2012M3A6A3055690).

References

Abdallah, M., Goswami, A. (2005). A biomechanically motivated two-phase strategy for biped robot upright balance control. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, Barcelona, pp. 3707–3713.

Bin Hammam, G., Orin, D. E., & Dariush, B. (2010). Whole-body humanoid control from upper-body task specifications. In *Proceedings of IEEE international conference on robotics and automation*, Anchorage, pp. 398–3405.

De Luca, A., Albu-Schaffer, A., Haddadin, S., & Hirzinger, G. (2006). Collision detection and safe reaction with the DLR-III lightweight manipulator arm. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Beijing, pp. 1623–1630.

Essén, H. (1993). Average angular velocity. *European Journal of Physics*, *14*, 201–205.

Fang, A. C., & Pollard, N. (2003). Efficient synthesis of physically valid human motion. *ACM Transactions on Graphics ACM SIGGRAPH Proceedings*, *22*(3), 417–426.

Featherstone, R. (2008). *Rigid body dynamics algorithms*. New York: Springer.

Featherstone, R., & Orin, D. E. (2000). Robot dynamics: equations and algorithms. In *Proceedings of IEEE international conference on robotics and automation*, San Francisco, pp. 826–834.

Featherstone, R., & Orin, D. E. (2008). Dynamics, chapter 2. In B. Siciliano & O. Khatib (Eds.), *Springer handbook of robotics*. New York: Springer.

Goswami, A., & Kallem, V. (2004). Rate of change of angular momentum and balance maintenance of biped robots. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, New Orleans, pp. 3785–3790.

Herr, H., & Popovic, M. B. (2008). Angular momentum in human walking. *The Journal of Experimental Biology*, *211*, 467–481.

Hofmann, A., Popovic, M., & Herr, H. (2009). Exploiting angular momentum to enhance bipedal center-of-mass control. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, Kobe, pp. 4423–4429.

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., & Hirukawa, H. (2003). Resolved momentum control: Humanoid motion planning based on the linear and angular momentum. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Las Vegas, pp. 1644–1650.

Khatib, O. (1987). A unified approach to motion and force control of robot manipulators: The operational space formulation. *IEEE Transactions on Robotics and Automation*, *3*(1), 43–53.

Komura, T., Leung, H., Kudoh, S., & Kuffner, J. (2005). A feedback controller for biped humanoids that can counteract large perturbations during gait. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, Barcelona, pp. 2001–2007.

Lee, S.-H., & Goswami, A. (2007). Reaction mass pendulum (RMP): An explicit model for centroidal angular momentum of humanoid robots. In *Proceedimgs of IEEE international conference on robotics and automation*, Rome, pp. 4667–4672.

Lee, S.-H., & Goswami, A. (November 2012). A momentum-based balance controller for humanoid robots on non-level and non-stationary ground. *Autonomous Robots*, *33*(4), 399–414.

Macchietto, A., Zordan, V., & Shelton, C. R. (2009). Momentum control for balance. *ACM Transactions on Graphics*, *28*(3), 80:1–80:8.

McMillan, S., & Orin, D. E. (1995). Efficient computation of articulated-body inertias using successive axial screws. *IEEE Transactions on Robotics and Automation*, *11*(4), 606–611.

Mistry, M., Righetti, L. (2012). Operational space control of constrained and underactuated systems. In H. Durrant-Whyte (Ed.), *Robotics: Science and systems VII* (pp. 225–232). Cambridge: The MIT Press.

Mitobe, K., Capi, G., & Nasu, Y. (2004). A new control method for walking robots based on angular momentum. *Mechatronics*, *14*, 163–174.

Morita, Y., & Ohnishi, K. (2003). Attitude control of hopping robot using angular momentum. In *Proceedings of IEEE international conference on industrial technology*, Vol. 1, pp. 173–178.

Nenchev, D., Umetani, Y., & Yoshida, K. (1992). Analysis of a redundant free-flying spacecraft/manipulator system. *IEEE Transactions on Robotics and Automation*, *8*(1), 1–6.

Nakamura, Y., & Yamane, K. (2000). Dynamics computation of structure-varying kinematic chains and its application to human figures. *IEEE Transactions on Robotics and Automation*, *16*(2), 124–134.

Naksuk, N., Mei, Y., & Lee, C. S. G. (2004). Humanoid trajectory generation: an iterative approach based on movement and angular momentum criteria. In *Proceedings of the IEEE-RAS international conference on humanoid robots*, Santa Monica, pp. 576–591.

Naksuk, N., Lee, C. S. G., & Rietdyk, S. (2005). Whole-body human to humanoid motion transfer. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Edmonton, pp. 104–109.

Naudet, J. (2005). Forward dynamics of multibody systems: A recursive Hamiltonian approach. PhD Thesis, Vrije Universiteit, Brussels.

Orin, D., & Goswami, A. (2008). Centroidal momentum matrix of a humanoid robot: Structure and properties. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nice, pp. 653–659.

Nishiwaki, K., Kagami, S., Kuniyoshi, Y., Inaba, M., & Inoue, H. (2002). Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired ZMP. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 2684–2689.

Papadopoulos, E. (1990). On the dynamics and control of space manipulators. PhD Thesis, Mechanical Engineering Department, Massachusetts Institute of Technology, Cambridge.

Popovic, M.B., Hofmann, A., & Herr, H. (2004). Zero spin angular momentum control: definition and applicability. In *Proceedings of the IEEE-RAS international conference on humanoid robots,* Santa Monica, pp. 478–493.

Popovic, M. B., Goswami, A., & Herr, H. (2005). Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. *International Journal of Robotics Research*, *24*(12), 1013–1032.

Roberson, R. E., & Schwertassek, R. (1988). *Dynamics of multibody systems*. Berlin: Springer.

Rodriguez, G., Jain, A., & Kreutz-Delgado, K. (1991). A spatial operator algebra for manipulator modelling and control. *International Journal of Robotics Research*, *10*(4), 371–381.

Sano, A., & Furusho, J. (1990). Realization of natural dynamic walking using the angular momentum information. In *Proceedings of IEEE international conference on robotics and automation*, Cincinnati, pp. 1476–1481.

Sciavicco, L., & Siciliano, B. (2005). Modeling and control of robot manipulators. *Advanced textbooks in control and signal processing series*. London: Springer London Limited.

Ugurlu, B., & Kawamura, A. (2010). Eulerian ZMP resolution based bipedal walking: Discussion on the intrinsic angular momentum rate change about center of mass. In *Proceedings of IEEE international conference on robotics and automation*, Alaska, pp. 4218–4223.

Walker, M. W., & Orin, D. (1982). Efficient dynamic computer simulation of robotic mechanisms in ASME. *Journal of Dynamic Systems Measurement and Control*, *104*, 205–211.

Zordan, V. (2010). Angular momentum control in coordinated behaviors. *Third annual international conference on motion in games*. Zeist.

Wieber, P.-B. (2005). *Holonomy and nonholonomy in the dynamics of articulated motion. Fast motions in biomechanics and robotics*. Heidelberg: Springer.

Wieber, P.-B. (2008). Viability and predictive control for safe locomotion. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nice, pp. 1103–1108.

Wensing, P. M., & Orin, D. E. (2013). Generation of dynamic humanoid behaviors through task-space control with conic optimization. In *Proceedings of IEEE international conference on robotics and automation (ICRA)*, Karlsruhe.

David E. Orin received his Ph.D. degree in Electrical Engineering from The Ohio State University in 1976. From 1976 to 1980, he taught at Case Western Reserve University. Since 1981, he has been at The Ohio State University, where he is currently a Professor Emeritus of Electrical and Computer Engineering. He was a sabbatical faculty at Sandia National Laboratories in 1996. Orin's research interests center on humanoid balance and running, dynamic maneuvers in legged locomotion, and robot dynamics. He has over 150 publications. His research has been supported by NSF, Sandia National Laboratories, DARPA, NASA, Cray Research, NRL, Los Alamos National Laboratory, and the Honda Research Institute. He is a Fellow of the IEEE (1993). He is the President of the IEEE Robotics and Automation Society (2012–2013) and received the Distinguished Service Award from the Society in 2004. He serves as a Part Editor on the Editorial Board for the award-winning Springer Handbook of Robotics.

Ambarish Goswami has been with Honda Research Institute in California, USA, for the past eleven years, where he is currently a Principal Scientist. His field is dynamics and control, and his main research is in the application areas of humanoid robots, assistive exoskeletons and vehicles. He received the Bachelor's degree from Jadavpur University, Kolkata, India, the Master's degree from Drexel University, Philadelphia, PA, and the Ph.D. degree from Northwestern University, Evanston, IL, all in Mechanical Engineering. Ambarish Goswami's Ph.D. work, under Prof. Michael Peshkin, was in the area of automated assembly and robot-assisted surgery. For four years following his graduation he worked at the INRIA Laboratory in Grenoble, France, as a member of the permanent scientific staff (Charge de Recherche). He became interested in human walking and in biomechanics while working on "BIP", the first anthropomorphic biped robot in France. This interest in gait study subsequently brought him to Prof. Norm Badler's Center for Human Modeling and Simulation at the University of Pennsylvania, Philadelphia, as an IRCS Fellow, and a three year position as a core animation software developer for 3D Studio Max at Autodesk. Ambarish has held visiting researcher positions at the Ohio State University and the University of Illinois at Urbana-Champaign for short periods. Ambarish has more than 70 publications with a total of more than 3400 Google Scholar citations; he has eleven patents. Ambarish is one of the Editors-in-Chief of the Springer Handbook of Humanoid Robotics (in preparation).

Sung-Hee Lee received the B.S. and the M.S. degree in mechanical engineering from Seoul National University, Korea, in 1996 and 2000, respectively, and the Ph.D. degree in computer science from University of California, Los Angeles, USA, in 2008. He is currently an Assistant Professor with the Graduate School of Culture Technology, KAIST. He was an Assistant Professor at Gwangju Institute of Science and Technology, Korea, from 2010 to 2013, and a Postdoctoral researcher at UCLA and at Honda Research Institute USA from 2008 to 2010. His research interests include humanoid robotics, physics-based computer graphics, biomechanical human modeling, and dynamics simulation.