Integrable Whole-body Orientation Coordinates for Legged Robots

Yu-Ming Chen[1,3], Gabriel Nelson[2,3], Robert Griffin[4], Michael Posa[1] and Jerry Pratt[3,4]

Abstract— Complex multibody legged robots can have complex rotational control challenges. In this paper, we propose a concise way to understand and formulate a *whole-body orientation* that (i) depends on system configuration only and not a history of motion, (ii) can be representative of the orientation of the entire system while not being attached to any specific link, and (iii) has a rate of change that approximates total system angular momentum. We relate this orientation coordinate to past work, and discuss and demonstrate, including on hardware, several different uses for it.

I. INTRODUCTION

Many legged robots are best represented by nontrivial multibody dynamic models. For these systems, reduced-order coordinates have been widely used for model-based planning and control, since these low-dimensional coordinates can capture the bulk of the robot's dynamics [1]–[4]. These models and their coordinates are often derived from physical intuition [3] or computed via optimization [4]. The total system center of mass (CoM) is likely the most well-known of these model-based coordinates. It responds to the total net forces acting on the robot according to Newton's 2nd Law, and gives us an overall translational location for the robot, providing extremely useful information for controlling locomotion.

When we look more closely at complex tasks, such as human-like walking and running, back-flips or aggressive turning, we typically need to also consider the orientation of the robot. In regulating orientation, many researchers have used models based on centroidal angular momentum [5], [6]. Centroidal angular momentum models are usually based on velocity level constraints, and tend not to readily produce an (unique) absolute orientation coordinate for the entire system. Other researchers use a single rigid body (SRB) model [7], where a single SO(3) coordinate represents the entire robot's orientation. The choice for this single body may stem from the morphology and mass distribution of a specific robot. For example, for robots with a heavy torso and light limbs, such as the MIT Mini Cheetah [8], the torso orientation can act as a good proxy for total system orientation. However, for robots with relatively heavy and/or

[1]The authors are with the General Robotics, Automation, Sensing and Perception (GRASP) Laboratory, University of Pennsylvania, Philadelphia, PA 19104, USA {yminchen, posa}@seas.upenn.edu

[2]The author is with the Boston Dynamics Artificial Intelligence Institute, 145 Broadway, Cambridge, MA 02142 gnelson@theaiinstitute.com

[3]The authors are with Boardwalk Robotics, 417 E Zaragoza St, Pensacola, FL 32502 jerry.pratt@boardwalkrobotics.com

[4]The authors are with the Florida Institute for Human and Machine Cognition (IHMC), 40 S Alcaniz St, Pensacola, FL 32502, USA rgriffin@ihmc.org
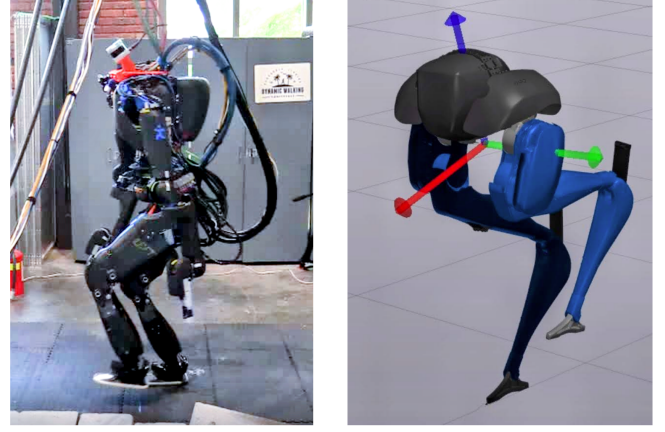
Fig. 1. Left: The humanoid Nadia walking with arm swing and spine yaw rotation induced by tracking WBO. Right: The simulated biped Cassie running and turning with the whole-body orientation (WBO) coordinate visualized at the CoM.

long limbs, where appreciable mass is distributed throughout the system and far from the CoM (e.g IHMC Nadia [9] and Agility Robotics Cassie [10]), the coordinate choice for system orientation is much less clear. Proposing a useful whole-body orientation coordinate for these types of systems, that is not attached to any specific link on the robot, is the focus of this paper.

Unlike the CoM, a weighted averaging of each link's orientation does not produce a consistently meaningful whole-body orientation. To derive this orientation, some have used angular excursion, which is an integral of an angular velocity about the CoM derived from centroidal angular momentum [11]. However, there is not a unique angular excursion value for a defined joint configuration, as the integral is path dependent. Another approach uses the principal axes of the whole-body inertia tensor to derive a whole-body orientation [12]. This approach also relies on the history of the axes in order to produce a meaningful continuous orientation.

Researchers in geometric mechanics have, for about a decade, devised ways of finding history-independent optimal coordinates, called the *minimum perturbation coordinates* [13]–[15]. However, to our knowledge, no literature has yet shown the application of these coordinates to a complex dynamic robot (except for Boston Dynamics' patent [16]), and they remain less prevalent than other approaches.

Similar to [13], our focus is to create an *integrable* (history and path invariant) measure of whole-body orientation for multi-link humanoid and legged robots, where this orientation coordinate (e.g. an angle, or Euler angles,

	Translational	Rotational
Position	CoM	**?**
	Integrable	**Non-integrable** ✗
Velocity	Linear momentum	Angular momentum

Fig. 2. While linear momentum is integrable, angular momentum is *generally* not [17], [18].

or quaternion, etc.) has dynamically analogous behavior to a CoM, but crucially in an angular sense. For example, once formulated, our desire would be that, should there be no external moments acting on the system, this orientation coordinate would remain at rest or rotate at a constant speed. We call this coordinate the integrable whole-body orientation (abbreviated WBO in this paper), though it has been called the *minimum perturbation coordinates* for general coordinates (other than SO(3) coordinates) [13], [14]. As such, we design the WBO to have the same mathematical form as other forward kinematics quantities such as a hand position, or the CoM. Thus, most or all tools and techniques that apply to forward kinematics can be applied to a WBO, such as inverse kinematics, task-space control and planning, etc.

Contributions of this paper

1) Presenting a simple example that clarifies the concept behind the WBO, with clear definitions of how the problem and solution are structured.
2) Providing a concise algorithm that shows how to find an WBO representation for a general multibody robot in 3D.
3) Demonstrating the use of the WBO on hardware (the humanoid robot Nadia) and in simulation (the biped robot Cassie); showing improvements in reducing angular momentum oscillation and foot yaw moment.

The paper is organized as follows. Section II uses simple 2D examples to introduce our WBO. Section III extends this to complex 3D systems and demonstrates the WBO algorithm with Nadia and Cassie. Sections IV and V apply the derived WBO in walking and running controllers. Section VI summarizes this paper and describes future work.

II. WBO OF SIMPLE SYSTEMS

A. Motivation and Problem Definition

Fig. 2 conceptually compares standard centroidal translational and rotational quantities often used in whole-body control. While the total system CoM (a position), and linear and angular momenta are concrete properties of a multibody system, there is in general no unique rotational coordinate (an orientation) corresponding to system CoM. Differentiation of system CoM (scaled by total mass) will arrive at linear momentum. Angular momentum though, for general systems, is *not* integrable [17], highlighting that angular momentum does not represent differential motion along any unique history or path invariant manifold in configuration space.

Thus, as discussed above, the *WBO* is an effort to approximate an angular measurement for the upper-right quadrant in Fig. 2. For general systems, it will be an angle, a set of Euler angles, or a quaternion, etc. Its value is a manufactured quantity (but not without physical relevance). It is the result of a design process that can take various forms, and we present one approach in this paper that we have found useful.

Often this topic involves a larger, typically *formal*, mathematical discussion about differential calculus and geometric mechanics, which we will consider beyond the scope of this paper. Deeper treatments can be found in [13]–[15]. Our goal is rather to provide a concise WBO formulation recipe, and demonstrate its initial use on a few legged robots.

We propose the following benefits from using a WBO for legged robots: (1) As a dynamically relevant WBO for controller tracking: e.g. providing a feedback signal for a proportional term on WBO control; (2) For planning WBO motions or changes; (3) For encouraging low angular momentum behavior for steady-state walking or running [19]–[21]. We believe that a suitable WBO measure for an anthropomorphic robot can aid in producing more natural looking movement, since the whole-body orientation control can be achieved by regulating the WBO directly, rather than controlling the orientation of some specific *base* link (often the pelvis or torso of the robot). This means the *base* link is now free to be treated as just another link on the robot, available for other user-specified objectives: e.g. smoothing system CoM motion by extending leg reach or stride length, stepping while ascending/descending terrain, whole-body reaching motions, etc.

B. The Bar-and-Flywheel Model

We propose the following simple example as an aid in understanding the WBO that we intend to find. The example has simple and complex versions, which are meant to demonstrate what the WBO does and does not represent.

Fig. 3 shows a planar "Bar-and-Flywheel" model: a long solid bar attached, at its CoM, to the axis of a flywheel via a rotary joint. The bar and flywheel have mass properties as indicated, and are free floating with no gravity or external forces acting. θ is the bar orientation in an inertially fixed world-frame, and ϕ is the flywheel orientation relative to the bar. A motor actuates the joint between the bar and flywheel. Thus, the bar represents the *base* of a multibody system, and the flywheel is an outboard body connected to the base by an actuated joint. The angular momentum about the CoM, also called the *centroidal angular momentum* (CAM), is

$$H_{CoM} = (I_B + I_F)\dot{\theta} + I_F\dot{\phi}. \tag{1}$$

In our definitions, we will delineate *base* orientation (θ in this example; in general a SO(3) representation) from *joint* positions or *joint* configuration (ϕ in this example; in general represented by the vector q).

Let the system start at rest with $\phi = 0$ (Fig. 3(a)). If the motor drives the flywheel counter-clockwise, the reaction torque will rotate the bar in the opposite direction. These rotation directions are indicated by the green arrows. Fig.

(a) The starting configuration ($\phi = 0$)

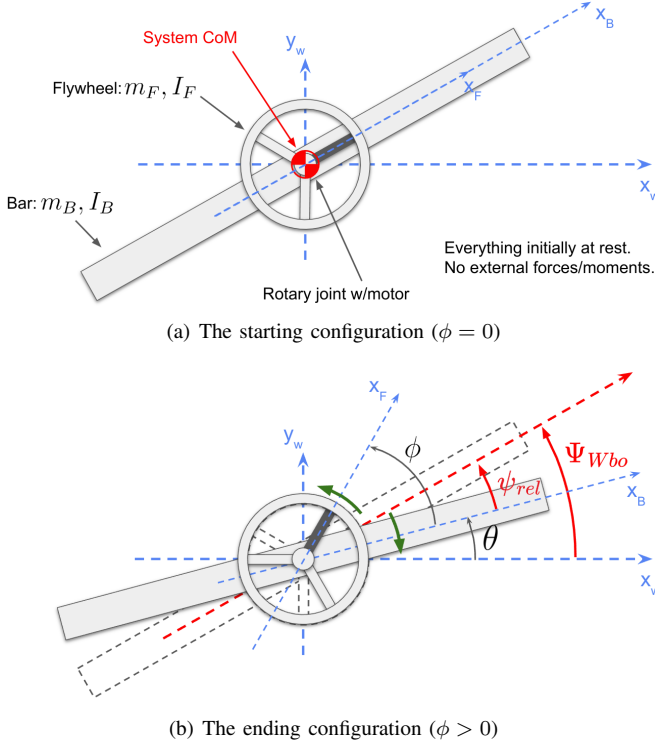(b) The ending configuration ($\phi > 0$)

Fig. 3. Bar-and-Flywheel model (an integrable system). The motor rotates the flywheel counter-clockwise.

3(b) shows both the starting (dashed lines) and ending configurations. We would like to understand how *the bar* moves due to the motion of the flywheel.

An important relationship often used in these problems is the *reconstruction equation*, which includes a *local connection* [14]. The reconstruction equation describes what we have simulated going from Fig. 3(a) to Fig. 3(b): It maps velocities in joint space ($\dot{\phi}$) to the velocity of the base ($\dot{\theta}$), as the base will counter-rotate due to changes in joint space. In this example ($H_{CoM} = 0$), this reconstruction equation is

$$\dot{\theta} = -\frac{I_F}{I_B + I_F}\dot{\phi}, \tag{2}$$

where the coefficient in front of $\dot{\phi}$ (without the negative sign) is the local connection. We can see that Eq. (2) can be integrated directly and also lets us predict the change in θ. Let $\Delta\theta$ and $\Delta\phi$ be the changes in the angles. We define

$$\psi_{rel} \triangleq -\Delta\theta = \frac{I_F}{I_B + I_F}\Delta\phi, \tag{3}$$

and we note that $\Delta\phi = \phi$ since the starting position of ϕ is 0. In Fig. 3(b), we label these various angles. The *initial* orientation of the bar, relative to the world, will be called Ψ_{Wbo}. In a general configuration, we can predict the final orientation of the system using Eq. (3):

$$\Psi_{Wbo} = \theta + \psi_{rel} = \theta + \frac{I_F}{I_B + I_F}\phi. \tag{4}$$

This Ψ_{Wbo} is the WBO of the system. Note that ψ_{rel} is the *relative* orientation of the WBO to the base (bar), such that the final Ψ_{Wbo} will be a base orientation relative to the world plus ψ_{rel}.
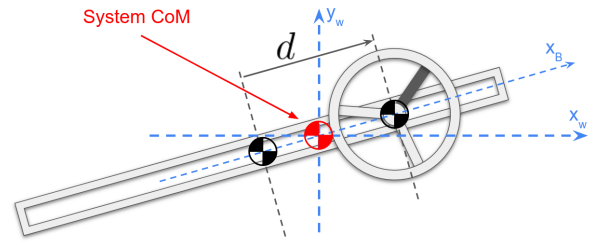

Fig. 4. Bar-and-Flywheel model with an actuated prismatic joint (a non-integrable system). This system can reorient itself without external torques. For example, the entire system can rotate counter-clockwise if the joints (d, ϕ) repeat the following cyclic motion: $(0, 0) \rightarrow (l/2, 0) \rightarrow (l/2, \pi/2) \rightarrow (0, \pi/2) \rightarrow (0, 0)$, where l is the length of the bar.

For this simple system, conservation of angular momentum dictates that Ψ_{Wbo} will never change, regardless of how we actuate the motor. Differentiating Eq. (4), we have (after some rearrangement)

$$(I_B + I_F)\dot{\Psi}_{Wbo} = (I_B + I_F)\dot{\theta} + I_F\dot{\phi}. \tag{5}$$

Note that the right hand sides of Eq. (1) and Eq. (5) are the same. Thus:

$$H_{CoM} = (I_B + I_F)\dot{\Psi}_{Wbo}. \tag{6}$$

Eq. (6) shows that the WBO has behavior analogous to a center of mass, *but in a rotational sense*: it represents an underlying orientation state for the system that can only be changed by external moments. The effective WBO mass moment of inertia (MoI) is, not surprisingly, the sum of the MoI's of the bar and flywheel.

The H_{CoM} above is an integrable differential form, meaning it can be derived from the differentiation of a manifold in configuration space, independent of the joints' history. This manifold can be expressed exactly as Eq. (4).

Also, looking at the left and right-hand sides of Eq. (6), we could think of a system having two representations of angular momentum: the *actual* system angular momentum about the CoM in Eq. (1), and an *approximated* angular momentum based on $\dot{\Psi}_{Wbo}$ in Eq. (6). For this integrable example system, we see that these angular momentum representations are equivalent, though for general systems (usually non-integrable) they are not.

We now extend this model by adding a long slot that allows the flywheel to slide along the length of the bar (Fig. 4). A linear actuator controls this movement, and d is the offset of the flywheel axis from the CoM of the bar. This change increases the size of our joint space (q) to 2 dimensions: $q = [d, \phi]^T$. The new CAM is

$$H_{CoM} = (I_B + I_F + \frac{m_B m_F}{m_B + m_F}d^2)\dot{\theta} + I_F\dot{\phi}. \tag{7}$$

If d is fixed, then this system is much the same as the simpler Bar-and-Flywheel model above. If, though, d is allowed to change, the angular momentum becomes non-integrable, and the goal of our WBO formulation becomes *contriving a differentiable and time independent function for ψ_{rel} that, when differentiated with respect to q, maximally approximates the local connection of the actual system over a*

user-defined region of joint-space. We will call this contrived function $\tilde{\psi}_{rel}(q)$. Like Eq. (2), the reconstruction equation is found by setting $H_{CoM} = 0$ in Eq. (7) and solving for $\dot{\theta}$:

$$\dot{\theta} = - \begin{bmatrix} 0 & \frac{I_F}{I_B + I_F + \frac{m_B m_F}{m_B + m_F} d^2} \end{bmatrix}_{1 \times 2} \begin{bmatrix} \dot{d} \\ \dot{\phi} \end{bmatrix}_{2 \times 1}, \qquad (8)$$

where the 1x2 matrix is the local connection, which is a function of d. In mathematical terms, the goal (stated above) is finding a differentiable function $\tilde{\psi}_{rel}(q)$, such that

$$\begin{bmatrix} \frac{\partial \tilde{\psi}_{rel}(q)}{\partial d} & \frac{\partial \tilde{\psi}_{rel}(q)}{\partial \phi} \end{bmatrix} \approx \begin{bmatrix} 0 & \frac{I_F}{I_B + I_F + \frac{m_B m_F}{m_B + m_F} d^2} \end{bmatrix}. \qquad (9)$$

Coming up with this function is the core design process, bearing in mind that an exact fit is impossible owing to the non-integrability of the physical system. For instance, an example function for $\tilde{\psi}_{rel}(q)$ could be $c_1 d^2 \phi + c_2 d^2 \phi^3 + c_3 \phi + c_4 \phi^3$, where the coefficients c_i are found numerically in order to maximize the approximation implied by Eq. (9) over a user-defined region in joint-space. Our final WBO now becomes (like Eq. (4)):

$$\tilde{\Psi}_{Wbo} \triangleq \theta + \tilde{\psi}_{rel}(q). \qquad (10)$$

Finally we note that, mirroring Eq. (6), we now have *actual* and *approximated* representations for angular momentum (approximation resulting from Eq. (9)):

$$H_{CoM} \approx \tilde{H}_{CoM} \qquad (11)$$

with

$$\tilde{H}_{CoM} = (I_B + I_F + \frac{m_B m_F}{m_B + m_F} d^2) \dot{\tilde{\Psi}}_{Wbo}, \qquad (12)$$

where

$$\dot{\tilde{\Psi}}_{Wbo} = \dot{\theta} + \begin{bmatrix} \frac{\partial \tilde{\psi}_{rel}(q)}{\partial d} & \frac{\partial \tilde{\psi}_{rel}(q)}{\partial \phi} \end{bmatrix} \begin{bmatrix} \dot{d} \\ \dot{\phi} \end{bmatrix}. \qquad (13)$$

The main compromise in our approach is approximating a non-integrable differential system with an integrable one. This "collapses" explicit representation of the nonholonomic motion of the actual physical system [14], [15], [17]. Our WBO will still measure nonholonomic motions, but its resulting dynamics will not correspond with fidelity to the actual externally applied moments.

III. WBO OF COMPLEX ROBOTS

A. Extending WBO to 3D

The goal of this section is to express Eq. (9) in the general 3D case, while the concepts discussed using the Bar-and-Flywheel models remain the same. Table I will be important in translating this structure into 3D. Fig. 5 shows the relevant frames, corresponding conceptually to Fig. 3(b).

We begin by translating Eq. (11) into 3D. The 3D angular momentum is

$$H_{CoM} = M_B \omega_B + M_q \dot{q} = M_B [\omega_B + A \dot{q}], \qquad (14)$$

where M_B and M_q are base and joint space centroidal momentum matrices [5], $A \triangleq M_B^{-1} M_q$ is the local connection and a function of q, ω_B is the angular velocity of the base
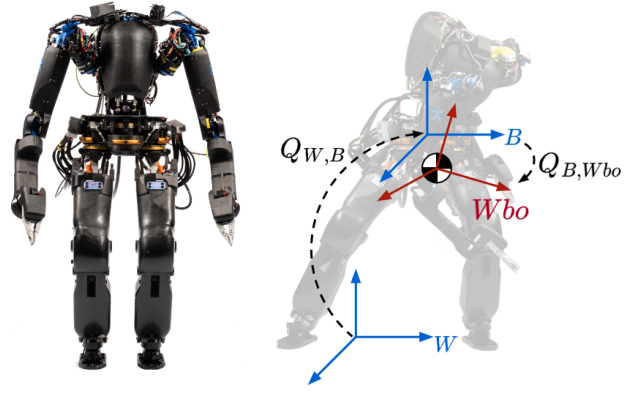

Fig. 5. The humanoid Nadia (left) and frames of interest (right). W, B and Wbo are the world frame, base frame and the WBO frame, respectively. $Q_{W,B}$ is the base orientation in the world, and $Q_{B,Wbo}$ is the orientation of the WBO frame relative to the base. Translationally, we locate the WBO frame at the CoM for convenience.

	Bar-Flywheel	in 3D
joint configuration	$[d, \phi]$	q
base orient. r.t. world	θ	$Q_{W,B}$
WBO r.t. base	$\tilde{\psi}_{rel}$	$Q \triangleq Q_{B,Wbo}$
WBO r.t. world	$\tilde{\Psi}_{Wbo}$	$Q_{W,Wbo}$

r.t. = relative to   Q = Quaternion

TABLE I
NOTATION DEFINITIONS AND CORRESPONDENCES

relative to the world expressed in the base frame, and $\dot{q}$ are the joint velocities. For the WBO, we have an approximated angular momentum

$$\tilde{H}_{CoM} = M_B [\omega_B + \tilde{A} \dot{q}], \qquad (15)$$

where $\tilde{A}$ will be the approximated local connection as discussed in arriving at Eq. (9). In 3D, trying to minimize the differences between these two representations (Eqs. (14) and (15)) means making:

$$A \dot{q} \approx \tilde{A} \dot{q}. \qquad (16)$$

We will keep $\dot{q}$ on both sides of the approximation until we sort out the 3D rotation representation for WBO in Section III-B. We note that $A \dot{q}$ on the left hand side of Eq. (16) is the *relative angular velocity* of the system [21]. On the other hand, $\Omega_{Wbo} \triangleq \tilde{A} \dot{q}$ is the angular velocity of the WBO frame relative to the base, expressed with respect to the base frame.

Paralleling the 2D example above (see Table I), we would like to find a function for $Q \triangleq Q_{B,Wbo}$. This represents the WBO frame orientation relative to the base. Thus, Ω_{Wbo} can be expressed from the quaternion rate $\dot{Q}$ using

$$\Omega_{Wbo} = 2 R_Q E_Q \dot{Q} \qquad (17)$$

where R_Q is the rotation matrix representation of Q, and the matrix $2 \cdot E_Q$ maps a quaternion rate to an angular velocity (details are omitted here for brevity; see [22]).

Algorithm 1 WBO optimization

Input: N random joint configurations $q_i,\ i = 1, ..., N$
Output: Θ^*
1: $\Theta \leftarrow 0$ (initialize to constant identity rotation)
2: **repeat**
3:     Substitute $Q(q_i; \Theta)$ into T_Q in Eq. (20) for $i = 1, ..., N$
4:     $\Theta \leftarrow$ Solve Eq. (20) with given T_Q
5: **until** convergence
6: **return** Θ

B. Parameterization and Optimization Algorithm

Noting that Q has two portions $Q = [Q_s; Q_{x,y,z}]$, we now parameterize $Q_{x,y,z}$ by a vector of basis functions $\lambda(q)$ with dimension n_λ:

$$Q_{x,y,z}(q; \Theta) = \Theta \lambda(q), \tag{18}$$

where $\Theta \in \mathbb{R}^{3 \times n_\lambda}$ is a coefficient matrix. Q_s can be recovered from the unit norm constraint $\|Q\|_2^2 = 1$. Similarly, we take the time derivatives of $Q_{x,y,z}$ and recover $\dot{Q}_s$ from $\frac{d}{dt}\|Q\|_2^2 = 0$. These algebraic manipulations will lead to a final form:

$$\tilde{A}\dot{q} = T_Q \Theta J_\lambda \dot{q} \tag{19}$$

where T_Q and J_λ are functions of q and are respectively defined as

$$T_Q \triangleq 2 R_Q E_Q \begin{bmatrix} -Q_s^{-1} Q_{x,y,z}^T \\ I_{3 \times 3} \end{bmatrix} \in \mathbb{R}^{3 \times 3}, \text{ and}$$

$$J_\lambda \triangleq \frac{\partial \lambda(q)}{\partial q} \in \mathbb{R}^{n_\lambda \times n_q}.$$

Given Eq. (16), our objective is to minimize the difference between A and $\tilde{A}$, both of which are functions of q. Since minimizing the difference over an infinite number of q in a region of joint space is often intractable, we pre-select N number of random configurations (uniformly distributed in the robot's operating joint space) to simplify the problem:

$$\min_\Theta \frac{1}{N} \sum_{i=1}^{N} \|A_i - T_{Q_i} \Theta J_{\lambda i}\|_F^2 \tag{20}$$

where $\|\cdot\|_F$ is the Frobenius norm. Given our choice in Eq. (18), we note that A and J_λ are independent of Θ, while T_Q is nonlinear in Θ.

One can solve Eq. (20) with many nonlinear solvers. In practice, we found that our simple algorithm in Alg. 1 works. The algorithm exploits the structure of the cost function by identifying that Eq. (20) is a least squares problem if T_Q is given. In each iteration, we first substitute the current solution Θ into T_Q to turn (20) into a least squares problem[4], and then solve the problem to get a new solution Θ. We repeat the above steps until Θ converges. This algorithm is similar to the Gauss-Newton method, differing in that it avoids linearizing the objective function at the solution in each iteration.

[4]The Kronecker product identity $vec(XYZ) = Z^T \otimes X vec(Y)$ is useful in vectorizing the matrix Θ in preparation for solving the least squares.
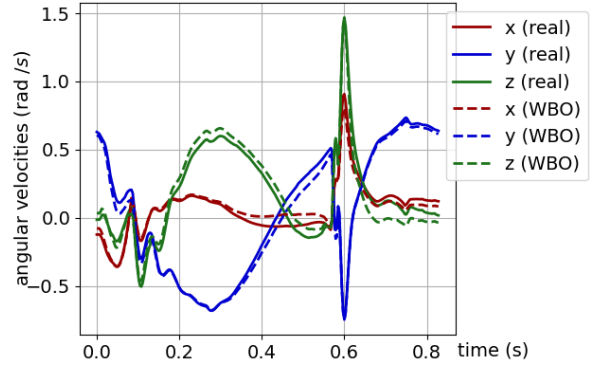
(a) Angular velocities of one step of Nadia robot walking on flat ground. The solid and dashed lines are $A\dot{q}$ and $\tilde{A}\dot{q}$ in Eq. (16), respectively. The spikes around 0.6 seconds are from the swing foot impact event and the feedback reaction of the walking controller.
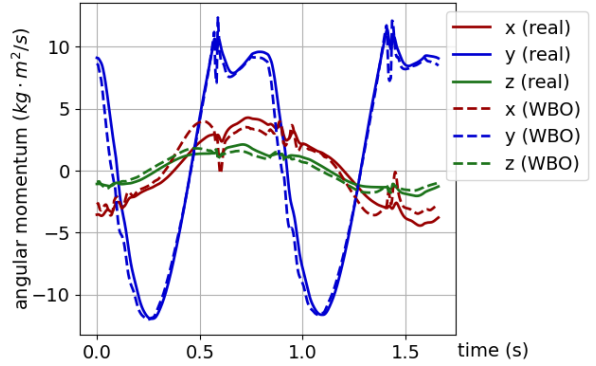
(b) The solid lines are the real CAM H_{CoM} in Eq. (14), and the dashed lines are the approximated CAM by the WBO $\tilde{H}_{CoM}$ in Eq. (15).

Fig. 6. Comparisons between real and approximated quantities. The data is from a simulation where Nadia walked in a straight line at 0.6 m/s.

C. WBO Optimization and Result

We optimized for an WBO function for Nadia (Fig. 5) using Alg. 1. Nadia is a humanoid robot with 31 degrees of freedom (DoF) – 6DoF legs, 7DoF arms, 1DoF grippers and a 3DoF spine. We randomly select 1000 configuration pairs mirrored about the sagittal plane (so $N = 2000$). We also keep the gripper, wrist and ankle joints at neutral positions, because their contribution to the CAM is relatively small. This reduces the configuration space (q) to 19 dimensions. The basis functions are monomials in terms q, with all possible monomials up to 3rd order being used (e.g. q_i, q_i^2, $q_i q_j$, q_i^3, $q_i^2 q_j$, ...), producing 1539 basis functions. The optimization converged smoothly in about 2 minutes or 10 iterations. After the optimization, we dropped terms with coefficients (in Θ) smaller than 1e-8.

In addition to Nadia, we also optimized a WBO function for Cassie running (Fig. 1). Cassie has 16 joints. We ignored the toe and ankle spring joints during optimization, reducing the configuration space to 12 dimensions. The optimization converged within 10 seconds and 7 iterations.

By comparing actual (measured) and approximated quantities, we can evaluate our WBO approximation at different signal scales. For Nadia walking, Fig. 6(a) plots both sides

of Eq. (16). Average angular velocity errors for each axis are about $[0.034, 0.035, 0.061]$ rad/s. Fig. 6(b) plots Eqs. (14) and (15). We see that H_{CoM} and $\tilde{H}_{CoM}$, which are larger signals dominated by base motion, are relatively close: average errors for each axis are about $[0.74, 0.84, 0.32]$ $\text{kg} \cdot \text{m}^2/\text{s}$. Thus, WBO reflects the actual H_{CoM} in a meaningful way.

IV. Walking example

In this section, we design a walking controller for Nadia using the WBO derived in Section III-C, and show that a WBO reference tracking can induce natural upper body motions during walking (Fig. 1).
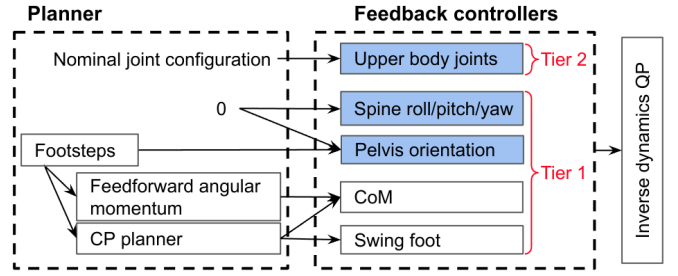
A. Controller

Fig. 7(b) shows our WBO controller, while Fig. 7(a) shows the baseline controller which fixes the desired joint positions for the upper body. Each controller has a planner that generates desired trajectories. These are then converted into acceleration commands by the feedback controllers shown in the diagrams. With the acceleration commands, we use an inverse dynamics whole body controller (the rightmost block in each diagram in Fig. 7) to get the desired actuator commands for the robot [23]. We can roughly separate the controller into leg and upper body parts. The leg part handles tracking the desired path and heading of the robot, while the rest of the controller handles the upper body motion.

1) Legs: This part of the controller is the same between the baseline and WBO controllers. We use Capture Point (CP) control for the locomotion task [24], [25]. The footsteps are generated given desired velocity commands from a higher level controller. The planner outputs a reference Centroidal Moment Pivot trajectory that is converted into a linear momentum rate command in the feedback controller, which is sent to the inverse dynamics controller.

2) Upper body: Our short-term goal was getting more natural arm swing and spine yaw rotation by servoing just WBO yaw[5]. In Fig. 7(b), we servo the WBO yaw axis relative to the world frame, while both the pelvis and the upper body tasks reside in the null space of the WBO task. In simulation, we achieved straight-line walking with this controller. When moving to hardware, we temporarily focused on demonstrating arm swing and spine yaw rotation. To do so, we servoed the pelvis orientation relative to the world and regulated the WBO yaw angle relative to the pelvis to 0. More complex motions have been left to future work, where we would like to take full advantage of our WBO.

We use a task hierarchy [26] in our whole body controller, shown in Fig. 7 as "Tiers". In experiments, we noticed that the inverse dynamics QP solver would trade swing foot orientation tracking performance for WBO tracking performance. This happened when the robot could not regulate WBO yaw to 0 with only the upper body. Thus, in order to prevent the

---

[5]In our experiments, we found that arm swing and spine yaw rotation were mostly induced by servoing the WBO yaw angle to zero. Additionally, Miyata et al. [21] only used the yaw part of angular momentum to generate the arm swing.

(a) Controller with constant desired joint positions of the upper body.
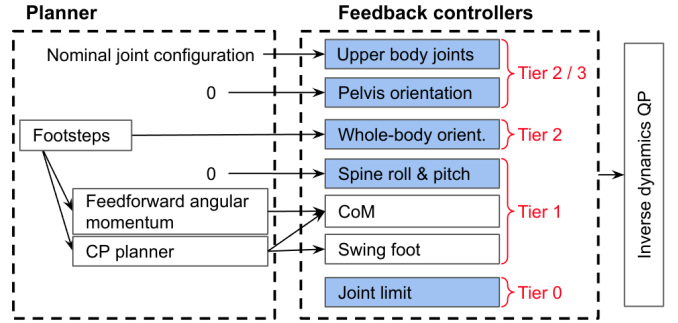
(b) Controller with WBO-induced upper body motions.

Fig. 7. Controller diagrams. Each diagram is composed of a high-level planner, low-level feedback controllers and an inverse-dynamics quadratic program (QP). Blue color highlights the difference between the two controllers. Red color is used for indicating the task priorities. Tier 0 is implemented as a constraint in the QP, while other Tiers are implemented via cost functions in the QP. Additionally, Tier n has higher priority than Tier $n+1$ for $n > 0$. We use the null-space projection technique to prioritize tasks [26]. The zeros and nominal joint configuration in the planner are constant trajectory sources.

WBO task from impairing the leg tasks, we set the WBO task to a lower priority than the leg tasks.

Besides the above task objectives, we also add nominal joint configuration tracking to handle the system's redundancy. This task can exist in the null space of the WBO task or at the same level as WBO. The parameters for this upper body joint controller can be used to sculpt the desired motion. For example, increasing the cost weight on the spine joint achieves more arm swing and less spine rotation.

3) Joint limits: The joint limit controller takes current joint positions and ranges of motion, and outputs limits on joint accelerations for the whole body controller. These limits are used for self-collision avoidance and aesthetics. Because the legs on Nadia are much heavier than the arms, when regulating the WBO yaw to 0, the robot can generate excessive arm swing or spine rotation. Thus, self-collision avoidance helps contain these motions, and therefore affects WBO tracking and overall appearance. We note that conventional momentum approaches [20], [21] would also exhibit this same behavior on Nadia.

B. Experiment Result

In both simulation and hardware experiments, we saw natural upper body motion induced by tracking a constant WBO. Additionally, although the controller for the upper body motion was designed for straight-line walking, we
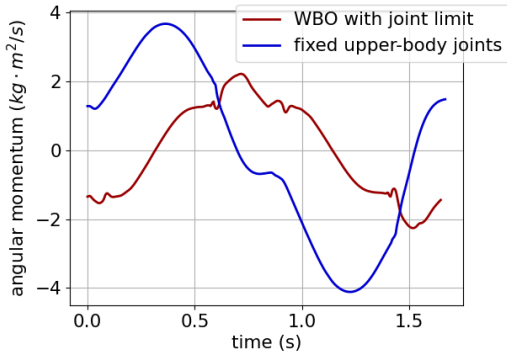
(a) Simulation (average walking speed ≈ 0.6 m/s)
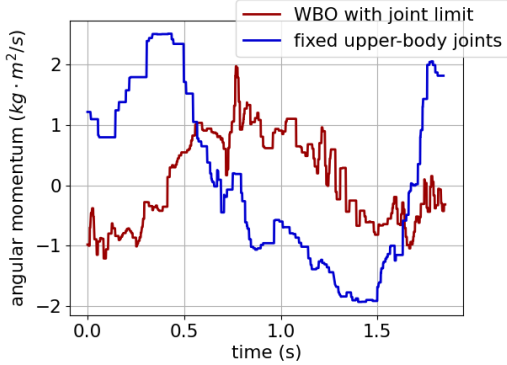
(b) Hardware (average walking speed ≈ 0.37 m/s)

Fig. 8. The CAM about the z axis when Nadia walked in a straight line. We note that there were issues with Nadia's leg actuator at the time of hardware experiment and the update rate of the control loop was not fast. These issues partially caused the non-smoothness in the hardware plot.

found that, in simulation, the robot was also able to walk forward, backward, sideways, and turn. The video clips can be found in the accompanying video for this paper.

Fig. 8 shows the z-component of CAM of straight-line walking for both the WBO and the baseline controller. We see that the angular momentum profiles look similar between the simulation and hardware, and that the CAM is 50% smaller when the desired WBO yaw is set to 0. Joint limit constraints prevent the CAM from tracking closer to zero. Additionally, we observed in simulation that the WBO controller reduces foot yaw moment against the ground. These are some of the advantages of using the upper body to counter moments generated by the legs during walking [21].

The conventional approach to generating natural upper body motions is directly minimizing the CAM [20], [21]. The downside of this approach is that the CAM controller is a feedback controller based on mutually constrained velocities rather than positions. Thus the upper body configuration could gradually drift away from a neutral target unless care is taken. To address this, a competing control objective is typically introduced that servos the robot, or some specific link on the robot, back to a desired orientation relative to the world. In contrast, a control law based on the WBO provides a single desired orientation for the entire robot, and thus need not employ competing objectives.
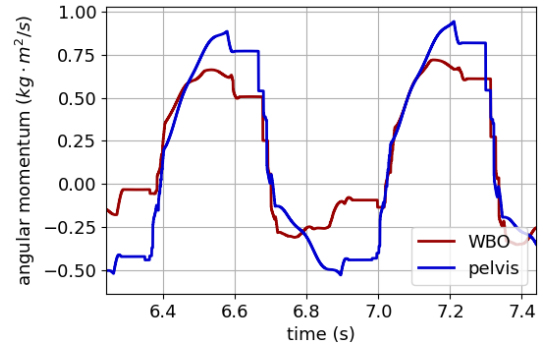

Fig. 9. The CAM about the z axis when Cassie runs at 2.7 m/s and follows a desired yaw trajectory that goes from 0 to $\pi/2$ rad in 10 seconds.

V. RUNNING EXAMPLE

Besides the walking example, we also want to test our WBO on a slightly more agile motion. For this, we implemented two running controllers on Cassie.

A. Controller

The baseline running controller uses a finite state machine with four states – left stance, left flight, right stance and right flight. The state transitions are triggered by foot touch-down and lift-off events. In the left/right stance state, the stance leg behaves like a vertical virtual spring, the pelvis pitch and roll angles are regulated to 0, and pelvis yaw follows a desired trajectory. The swing leg uses a Raibert-style control law [27], while the leg length is determined by the nominal leg length at touchdown. In the flight state, the controller continues to track the desired orientation of the pelvis and the desired positions of the leading swing foot. The desired pelvis orientation is relative to the world frame.

The second (preliminary) running controller is the same as the baseline controller, except that we replace the pelvis yaw with WBO yaw.

B. Experiment Result

In our experiments, Cassie is commanded to run at 2.7 m/s in the Drake simulator [28]. We also set a desired yaw trajectory, which goes from 0 to $\pi/2$ rad linearly in 10 seconds. The baseline controller tracks this desired yaw with the pelvis, while the WBO controller tracks it with WBO. Fig. 9 shows the CAM of Cassie. We can see that the momentum oscillates less with the WBO controller (more than a 26% reduction). This reduction is due to the WBO representing the orientation of the entire system, and the total momentum is approximated by its time derivative via Eq. (15). Also, the WBO controller is able to adjust the desired pelvis orientation when the legs move. In contrast, the baseline controller considers the pelvis motion only and ignores the contributions from the legs. One could, of course, regulate the CAM while tracking the desired orientation of the pelvis, but these two objectives could conflict since the pelvis alone does not represent the entire system well. The advantage of the WBO approach here is the consistency between the orientation-tracking and momentum objectives.

VI. CONCLUSION AND FUTURE WORK

We introduced the integrable whole-body orientation (WBO) with simple examples and clear problem motivation, so it is more accessible to a general robotic audience. A formulation of the WBO problem was provided, including an algorithm that solves the problem quickly by exploiting its structure. WBO functions were synthesized offline for the Nadia and Cassie robots, and were then used to induce arm swing and spine yaw rotation in a walking example and to turn the robot's global orientation in a running example.

The WBO enables us to servo the orientation of the entire system. Thus, it can free up the base link (e.g. pelvis) to achieve high-level goals such as natural walking with natural pelvis motion and stepping up/down terrain. In this paper, we mostly demonstrated more natural arm swing and spine rotation. Future work will utilize the WBO to achieve more complex behaviors, such as whole-body natural walking. Another area of future research involves incorporating high-level planning for the WBO trajectories (e.g. using the SRB model in planning), which could potentially enable more agile motions for the robots. Lastly, this paper does not explore the impact of WBO on system stability, leaving it for further investigation. However, a prior study [29] found larger regions of attraction for balance and step recovery by moving from a point-mass to SRB model. We believe substituting WBO for the SRB model could also improve stability for the above systems.

VII. ACKNOWLEDGMENT

REFERENCES

[1] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. Del Prete, "Optimization-based control for dynamic legged robots," *arXiv preprint arXiv:2211.11644*, 2022.

[2] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, "Optimization-based locomotion planning, estimation, and control design for Atlas," *Autonomous Robots*, vol. 40, no. 3, pp. 429–455, 2016.

[3] S. Kajita and K. Tani, "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pp. 1405–1406, IEEE Computer Society, 1991.

[4] Y.-M. Chen and M. Posa, "Optimal reduced-order modeling of bipedal locomotion," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8753–8760, IEEE, 2020.

[5] D. E. Orin and A. Goswami, "Centroidal momentum matrix of a humanoid robot: Structure and properties," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 653–659, IEEE, 2008.

[6] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body Motion Planning with Centroidal Dynamics and Full Kinematics," *IEEE-RAS International Conference on Humanoid Robots*, 2014.

[7] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2245–2252, IEEE, 2018.

[8] B. Katz, J. Di Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6295–6301, IEEE, 2019.

[9] "Nadia humanoid." https://www.ihmc.us/nadia-humanoid/. Accessed: 2022-09-05.

[10] R. Batke, F. Yu, J. Dao, J. Hurst, R. L. Hatton, A. Fern, and K. Green, "Optimizing bipedal maneuvers of single rigid-body models for reinforcement learning," *arXiv preprint arXiv:2207.04163*, 2022.

[11] V. Zordan, D. Brown, A. Macchietto, and K. Yin, "Control of rotational dynamics for ground and aerial behavior," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 10, pp. 1356–1366, 2014.

[12] W. Du, Z. Wang, E. Moullet, and F. Benamar, "Meaningful centroidal frame orientation of multi-body floating locomotion systems," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3061–3067, IEEE, 2021.

[13] M. Travers, R. Hatton, and H. Choset, "Minimum perturbation coordinates on so (3)," in *2013 American Control Conference*, pp. 2006–2012, IEEE, 2013.

[14] R. L. Hatton and H. Choset, "Geometric motion planning: The local connection, stokes' theorem, and the importance of coordinate choice," *The International Journal of Robotics Research*, vol. 30, no. 8, pp. 988–1014, 2011.

[15] R. L. Hatton and H. Choset, "Nonconservativity and noncommutativity in locomotion: geometric mechanics in minimum-perturbation coordinates," *The European Physical Journal Special Topics*, vol. 224, no. 17-18, pp. 3141–3174, 2015.

[16] A. Khripin and A. A. Rizzi, "Natural pitch and roll," Dec. 13 2016. US Patent 9,517,561.

[17] Y. Nakamura and R. Mukherjee, "Nonholonomic path planning of space robots via bi-directional approach," in *Proceedings., IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1764–1769, IEEE, 1990.

[18] A. Saccon, S. Traversaro, F. Nori, and H. Nijmeijer, "On centroidal dynamics and integrability of average angular velocity," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 943–950, 2017.

[19] M. B. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *The International Journal of Robotics Research*, vol. 24, no. 12, pp. 1013–1032, 2005.

[20] T. Erez and E. Todorov, "Trajectory optimization for domains with contacts using inverse dynamics," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4914–4919, IEEE, 2012.

[21] A. Miyata, S. Miyahara, and D. N. Nenchev, "Walking with arm swinging and pelvis rotation generated with the relative angular acceleration," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 151–158, 2019.

[22] B. Wie and P. M. Barba, "Quaternion feedback for spacecraft large angle maneuvers," *Journal of Guidance, Control, and Dynamics*, vol. 8, no. 3, pp. 360–365, 1985.

[23] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot Atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 1, 2016.

[24] T. Koolen, T. De Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models," *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.

[25] T. Seyde, A. Shrivastava, J. Englsberger, S. Bertrand, J. Pratt, and R. J. Griffin, "Inclusion of angular momentum during planning for capture point based walking," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1791–1798, IEEE, 2018.

[26] M. Hutter, M. A. Hoepflinger, C. Gehring, M. Bloesch, C. D. Remy, and R. Siegwart, "Hybrid operational space control for compliant legged systems," *Robotics*, p. 129, 2013.

[27] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.

[28] R. Tedrake and the Drake Team Development, "Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems," 2016.

[29] M. A. Posa, T. Koolen, and R. L. Tedrake, "Balancing and step recovery capturability via sums-of-squares optimization," 2017.