Trajectory Generation for Legged Robots Based on a Closed-form Solution of Centroidal Dynamics

Yuichi Tazaki[1]

***Abstract*—This paper proposes a novel reduced-order model of rigid body dynamics that can be used for versatile motion generation of legged robots. It can express both linear CoM movement and rotation of the base link, and it is applicable to general multi-contact setup including flight. Moreover, its closed-form solution enables optimization of long trajectories consisting of more than 20 contact phases in less than 100ms. Various trajectory optimization examples of biped and quadruped models together with trajectory tracking simulation using whole-body MPC are shown to demonstrate the flexibility and practical applicability of the proposed model.**

***Index Terms*—Legged Robots, Trajectory Generation, Centroidal Dynamics, Closed-form Solution**

I. Introduction

A. Importance of Compact and Expressive Models for Trajectory Generation

Recent years have seen rapid development of humanoid and legged robots and their application in various fields such as logistics, security, and social robots. Since these types of robots possess many degrees of freedom, generating motion in real time using a full-DoF dynamics model requires large computational resource. Therefore, various reduced-order models (also known as template models) have been widely used in combination with full models to reduce the computational cost of trajectory optimization and model predictive control methods [1, 2, 3].

B. Review of Reduced-Order Models for Trajectory Generation

The centroidal dynamics (CD) [4], which expresses the relationship between the linear and angular momenta and total contact wrench, is low-dimensional dynamics naturally embedded in rigid body systems. Various reduced-order models of legged robots known in the literature are derived by simplifying the CD in different ways. The linear inverted pendulum mode (LIPM) [5] expresses the motion of the center-of-mass (CoM) driven by the zero-moment point (ZMP). It was originally limited to the horizontal movement but later generalized to a 3D model named the Variable-Height Inverted Pendulum (VHIP) [6] and to include angular momentum in roll and pitch directions [7, 8, 9]. In [10], planning of CoM movement based on the VHIP was solved by decomposing it into the vertical and horizontal components. The spring-loaded inverted pendulum (SLIP) model [11] can express the vertical motion of the CoM required for running. Because of its nonlinearity, however, its closed-form solution is not known to date. Combination of the LIPM and the SLIP was discussed in [12, 13]. Reduced-order models are integrated into trajectory optimization (TO) problems in various ways. Existing methods for TO of CoM motion include: using the LIPM as the state equation [14, 15], defining desired ZMP as a cost [16], and defining a ZMP-based stability criterion as an output constraint [17, 18, 19]. Existing approaches to optimization of trajectories combining both CoM motion and base link rotation include: using the centroidal dynamics as a constraint or a state equation [20, 21, 22, 23, 24, 25], and imposing a stability condition based on the ZMP support region or the centroidal wrench cone (CWC) [26, 8]. Numerical integration and collocation methods require small time step to achieve acceptable accuracy, which generally results in a large number of decision variables. Moreover, feasibility is guaranteed only on discrete keypoints, although studies on feasibility between sample points exist [27, 28]. If a closed-form solution of the dynamical system model is available, rigorous solution trajectories can be expressed by minimum number of decision variables, and moreover, as long as constraints on control inputs (e.g., ZMP and contact wrench) are satisfied, the feasibility of the trajectory at arbitrary time instant is guaranteed. Some attempts to linearize or analytically integrate the CD have been made in the past studies. It has been shown in [29] that CD can be linearized by ignoring the rotational dynamics along the vertical axis and assuming constant CoM height. Although these restrictions may be acceptable in many practical scenes, they prohibit application of this approach to planning of multi-contact and acrobatic motion. One commonly adopted way to analytically integrate the centroidal dynamics over a fixed time interval is to apply the zero-order hold directly to the contact wrench. However, piecewise-constant contact wrench inevitably induces variation of angular momentum, which is undesirable in many types of motion such as normal walking. To limit variation within an acceptable range, the integration time step must be set small enough. The multi-contact (mc-) LIPM proposed in our previous study [30] expressed contact force acting on each contact point in terms of stiffness and the displacement between the contact point and the CoM. Thanks to this formulation, the integration time step could be set as large as a single contact phase without causing variation of angular momentum. However, the model had limitation in the

Manuscript received: May 22, 2024; Accepted August 30, 2024.

This paper was recommended for publication by Editor Abderrahmane Kheddar upon evaluation of the Associate Editor and Reviewers ' comments. This work was supported by JSPS KAKENHI Grant Number 24K03017.

[1]Yuichi Tazaki is with Graduate School of Engineering, Department of Mechanical Engineering, Kobe University tazaki@mech.kobe-u.ac.jp

Digital Object Identifier (DOI): see top of this page.

expressiveness of the rotational dynamics.

C. Contribution of This Letter

The main contributions of this letter is summarized as follows.

- The stiffness-based centroidal dynamics (SBCD) is proposed as a generalization of the mc-LIPM that can express both translational and rotational dynamics. Moreover, the SBCD is shown to include existing linear models as special cases.
- Closed-form solutions of the centroidal dynamics based on two types of parametrization of contact wrench, stiffness-based and direct, are derived to be utilized in trajectory optimization.
- Various trajectory optimization examples of bipedal and quadruped locomotion are presented to demonstrate the flexibility of the proposed model. Moreover, trajectories generated with the stiffness-based and direct formulation are compared.
- To verify that reference trajectories generated with the proposed model can be tracked under realistic kino-dynamic constraints, trajectory tracking using whole-body MPC is implemented and tested in rigid-body simulation.

The organization of this letter is as follows. In Section II, the derivation of the SBCD and its relationship with conventional models is shown. In Section III, an optimal control problem for trajectory generation is formulated and its solution strategy is discussed. In Section IV, several trajectory generation results are presented together with results of whole-body trajectory tracking control in simulation. Concluding remarks are given in Section V.

II. Centroidal Dynamics

A. Stiffness-based Centroidal Dynamics

Let us start from the well-known centroidal dynamics [31] equation, which describes the dynamical relationship between the linear and angular momenta and the total external wrench.

$$m\ddot{\boldsymbol{p}} = \boldsymbol{f} - m\boldsymbol{g}, \tag{1a}$$
$$\dot{\boldsymbol{L}} = \boldsymbol{\eta} \tag{1b}$$

Here, $\boldsymbol{p}$ is the position of the CoM, $\boldsymbol{L}$ is the angular momentum around the CoM, and $\boldsymbol{f}$ and $\boldsymbol{\eta}$ are the translational and rotational component of the total external wrench, respectively. Moreover, m is the total mass and $\boldsymbol{g}$ is the gravitational acceleration. We assume that all external forces are contact forces acting between the robot and the environment. The robot can make contact with the environment with n_e *ends*. We use the term *end* to refer to a part of the robot that makes contact with the environment. Let us denote the translational and rotational components of the contact wrench applied to the l-th end ($l = 1, 2, \ldots, n_\mathrm{e}$) by $\boldsymbol{f}_l$ and $\boldsymbol{\eta}_l$, respectively. Then the total contact wrench is expressed as follows.

$$\boldsymbol{f} = \textstyle\sum_l \boldsymbol{f}_l, \quad \boldsymbol{\eta} = \sum_l \left[(\boldsymbol{p}_l - \boldsymbol{p}) \times \boldsymbol{f}_l + \boldsymbol{\eta}_l\right] \tag{2}$$

Here, $\boldsymbol{p}_l$ denotes the position of the l-th end.

The centroidal dynamics fall into the class of *bilinear systems* because of the cross product included in (2). One way to derive a closed-form solution of the centroidal dynamics is to apply the zero-order hold to contact wrenches, as is commonly done in the literature. In this letter, we propose another approach, which is to parametrize contact wrenches in a different form and apply the zero-order hold to the parameters. To this aim, let us express the contact wrenches in the following form.

$$\boldsymbol{f}_l = m\lambda_l^2(\boldsymbol{p} - (\boldsymbol{p}_l + \boldsymbol{r}_l)), \quad \boldsymbol{\eta}_l = m\lambda_l^2\hat{\boldsymbol{\eta}}_l \tag{3}$$

Here, λ_l is the stiffness and it expresses the strenth of the contact wrench. Since contact forces are repulsive, the stiffness must satisfy $\lambda_l \geq 0$. Moreover, $\boldsymbol{r}_l$ expresses a shift in the direction of the contact force; the contact force points from $\boldsymbol{p}_l$ towards $\boldsymbol{p} - \boldsymbol{r}_l$. It is conceptually similar to the CMP (centroidal moment pivot) [32] and eCMP [33] ; the difference is that it is defined separately for each end. Hereafter we call $\boldsymbol{r}_l$ the CMP offset of the l-th end. As a result, we obtain the following equations of motion, which we call the *s*tiffness-based centroidal dynamics (see Appendix for step-by-step derivation).

$$\ddot{\boldsymbol{p}} = \bar{\lambda}^2(\boldsymbol{p} - (\bar{\boldsymbol{p}} + \bar{\boldsymbol{r}})), \tag{4a}$$
$$\dot{\boldsymbol{L}} = m(\ddot{\boldsymbol{p}} \times \bar{\boldsymbol{r}} + \bar{\boldsymbol{\eta}}) \tag{4b}$$

where

$$\bar{\lambda} = \sqrt{\textstyle\sum_l \lambda_l^2 + \epsilon^2}, \;\; \bar{\boldsymbol{p}} = \frac{\sum_l \lambda_l^2 \boldsymbol{p}_l + \boldsymbol{g}}{\bar{\lambda}^2}, \;\; \bar{\boldsymbol{r}} = \frac{\sum_l \lambda_l^2 \boldsymbol{r}_l}{\bar{\lambda}^2}, \\ \bar{\boldsymbol{\eta}} = \bar{\lambda}^2 \bar{\boldsymbol{p}} \times \bar{\boldsymbol{r}} + \textstyle\sum_l \lambda_l^2(\hat{\boldsymbol{\eta}}_l - \boldsymbol{p}_l \times \boldsymbol{r}_l). \tag{5}$$

Here, $\epsilon \geq 0$ is a small constant inserted to avoid dividing by zero in the case of a flight-phase (i.e., $\lambda_l = 0 \quad \forall l$). If flight phases need not be considered, then ϵ can be set as 0 and (4a) and (4b) hold precisely. Otherwise the equalities hold approximately.

Remark 1. Similar stiffness-based (or force-to-point [33]) parametrization has been explored in the literature. The main difference is that our model directly parametrizes the contact wrench of each end, whereas existing models apply similar parametrization to the total contact wrench. The stiffness, CoP (center-of-pressure), and eCMP in the conventional sense appear as weighted averages: $\bar{\lambda}$, $\bar{\boldsymbol{p}}$, and $\bar{\boldsymbol{p}} + \bar{\boldsymbol{r}}$, respectively.

Remark 2. If all ends are off-contact, the movement of the CoM becomes ballistic ($\ddot{\boldsymbol{p}} = -\boldsymbol{g}$) and the angular momentum is conserved ($\dot{\boldsymbol{L}} = \boldsymbol{0}$). In fact, by setting $\lambda_l = 0 \; \forall l$, we obtain $\ddot{\boldsymbol{p}} = \epsilon^2\boldsymbol{p} - \boldsymbol{g} \approx -\boldsymbol{g}$, $\dot{\boldsymbol{L}} = \boldsymbol{0}$. Therefore, (4a) and (4b) express centroidal dynamics in general multi-contact configurations including flight.

Remark 3. By setting $\boldsymbol{r}_l = \boldsymbol{0} \; \forall l$, we obtain $\ddot{\boldsymbol{p}} = \bar{\lambda}^2(\boldsymbol{p} - \bar{\boldsymbol{p}})$, $\dot{\boldsymbol{L}} = \boldsymbol{\eta}$ and thus the linear and rotational dynamics are decoupled. In this case, the linear dynamics is equivalent to the VHIP [6].

Remark 4. By assuming $p_{l,\mathrm{z}} = r_{l,\mathrm{z}} = 0\ \forall l$, eliminating $\bar{\boldsymbol{r}}$, ignoring the z-component and rearranging the x, y components, we obtain

$$\ddot{p}_{\mathrm{x}} = \bar{\lambda}^2 \left(p_{\mathrm{x}} - \bar{p}_{\mathrm{x}} - \frac{\dot{L}_{\mathrm{y}} - \eta_{\mathrm{y}}}{m(\ddot{p}_{\mathrm{z}} + g)} \right)$$
$$\ddot{p}_{\mathrm{y}} = \bar{\lambda}^2 \left(p_{\mathrm{y}} - \bar{p}_{\mathrm{y}} + \frac{\dot{L}_{\mathrm{x}} - \eta_{\mathrm{x}}}{m(\ddot{p}_{\mathrm{z}} + g)} \right)$$

which is essentially the same as the VIP [7] and the NIPFM [8].

B. Closed-form Solutions and Discrete-time Equations

Consider a finite interval of time $[0, T]$ which is further split into N sub-intervals. The k-th interval is given by $[t_k, t_{k+1}]$ ($t_{k+1} = t_k + \tau_k$). During each interval, the contact state of each end does not change. In other words, contact states may change only at discrete time instants t_k.

1) Direct Formulation: By applying the zero-order hold to $\boldsymbol{f}_l$ and $\boldsymbol{\eta}_l$, we obtain the following discrete-time equations.

$$\boldsymbol{p}(t) = \boldsymbol{p}_k + (t - t_k)\boldsymbol{v}_k + \frac{(t - t_k)^2}{2} \left(\frac{1}{m} \textstyle\sum_l \boldsymbol{f}_{l,k} - \boldsymbol{g} \right) \tag{6a}$$

$$\boldsymbol{v}(t) = \boldsymbol{v}_k + (t - t_k) \left(\frac{1}{m} \textstyle\sum_l \boldsymbol{f}_{l,k} - \boldsymbol{g} \right) \tag{6b}$$

$$\begin{aligned} \boldsymbol{L}(t) = \boldsymbol{L}_k &+ (t - t_k) \textstyle\sum_l \left[(\boldsymbol{p}_l - \boldsymbol{p}_k) \times \boldsymbol{f}_{l,k} + \boldsymbol{\eta}_{l,k} \right] \\ &+ \left(\textstyle\sum_l \boldsymbol{f}_{l,k} \right) \times \left(\frac{(t - t_k)^2}{2} \boldsymbol{v}_k - \frac{(t - t_k)^3}{6} \boldsymbol{g} \right) \end{aligned} \tag{6c}$$

Although this method is close to those used in [1][34], there is a difference in the treatment of the moment arm $\boldsymbol{p} - \boldsymbol{p}_l =: \boldsymbol{r}_l$; namely, existing methods approximate $\boldsymbol{r}_l$ to be fixed during the integration interval, whereas our method integrates $\boldsymbol{p}$ without approximation.

2) Stiffness-based Formulation: In the stiffness-based formulation, the zero-order hold is applied to λ_l, $\boldsymbol{r}_l$, and $\hat{\boldsymbol{\eta}}_l$ to obtain the following.

$$\begin{aligned} \boldsymbol{p}(t) = \bar{\boldsymbol{p}}_k + \bar{\boldsymbol{r}}_k + C_k(t - t_k) \left(\boldsymbol{p}_k - (\bar{\boldsymbol{p}}_k + \bar{\boldsymbol{r}}_k) \right) \\ + \frac{S_k(t - t_k)}{\bar{\lambda}_k} \boldsymbol{v}_k \end{aligned} \tag{7a}$$

$$\boldsymbol{v}(t) = \bar{\lambda}_k S_k(t - t_k)(\boldsymbol{p}_k - (\bar{\boldsymbol{p}}_k + \bar{\boldsymbol{r}}_k)) + C_k(t - t_k)\boldsymbol{v}_k \tag{7b}$$

$$\boldsymbol{L}(t) = \boldsymbol{L}_k + m((\boldsymbol{v}(t) - \boldsymbol{v}_k) \times \bar{\boldsymbol{r}} + (t - t_k)\bar{\boldsymbol{\eta}}_k) \tag{7c}$$

where

$$C_k(t) = \cosh\left(\bar{\lambda}_k t\right),\ S_k(t) = \sinh\left(\bar{\lambda}_k t\right)$$

and $\bar{\lambda}_k$, $\bar{\boldsymbol{r}}_k$, and $\bar{\boldsymbol{\eta}}_k$ are given by substituting $\{\lambda_{l,k}, \boldsymbol{p}_{l,k}, \boldsymbol{r}_{l,k}, \hat{\boldsymbol{\eta}}_{l,k}\}$ into (5).

C. Integration of Base Link Rotation

The discrete-time dynamics derived in the previous section does not express rotation explicitly in terms of orientation and angular velocity. This could be inconvenient if desired rotational movement is to be specified in trajectory optimization. In a general multi-body system, the relationship between the total angular momentum and the angular velocity of the base link is expressed as follows.

$$\boldsymbol{L} = (RIR^{\mathsf{T}})\boldsymbol{\omega} + R\hat{\boldsymbol{L}} \tag{8}$$

where R is the rotation of the base link, RIR^{T} is the composite rigid body inertia of the system and $\hat{\boldsymbol{L}}$ is angular momentum genereted by motion relative to the base link. Note that (8) is essentially the same as the rotational component of (7)-(9) in [35]. Because I and $\hat{\boldsymbol{L}}$ depend on the whole-body kinematics and mass distribution of the robot, some form of approximation is necessary in trajectory planning based solely on the centroidal dynamics. In the following, we assume that reference values of these quantities, denoted by I_{ref} and $\hat{\boldsymbol{L}}_{\mathrm{ref}}$, are given. Based on this information, the angular velocity of the base link is given by

$$\boldsymbol{\omega} = RI_{\mathrm{ref}}^{-1}(R^{\mathsf{T}}\boldsymbol{L} - \hat{\boldsymbol{L}}_{\mathrm{ref}}) \tag{9}$$

Note that this formulation includes the commonly adopted single rigid-body approximation, in which case $\hat{\boldsymbol{L}}_{\mathrm{ref}} \equiv \boldsymbol{0}$ and I_{ref} is a constant matrix. If reference whole-body motion is known in advance, this information can be reflected to I_{ref} and $\hat{\boldsymbol{L}}_{\mathrm{ref}}$ to generate centroidal trajectories that are more compatible with the whole-body dynamics. Analytical integration of 3D rotation is difficult in general, except for a special case in which the axis of rotation is fixed during the integration interval. Thus, we resort to approximate stepping scheme such as Euler stepping, following the line of [36]. Let us subdivide the interval $[t_k, t_{k+1}]$ into n_{div} even subintervals $[t'_i, t'_{i+1}]$ ($i = 0, 1, \ldots, n_{\mathrm{div}}$), where $t'_i = t_k + \tau'_k i$, $\tau'_k = \frac{\tau_k}{n_{\mathrm{div}}}$. The update of orientation is expresses as follows.

$$\boldsymbol{q}_{k+1} = q\left(\boldsymbol{\omega}(t'_{n_{\mathrm{div}}-1})\tau'_k\right) \cdot \ \ldots\ \cdot q\left(\boldsymbol{\omega}(t'_0)\tau'_k\right) \cdot \boldsymbol{q}_k \tag{10}$$

Here, $\boldsymbol{\omega}(t)$ is given by (9). The resolution of rotation n_{div} must be selected by the designer. Note that the value of n_{div} does not affect the number of decision variables of the trajectory optimization problem described in the next section. Selecting greater n_{div}, however, will increase the cost for computing the gradient of (10) used inside the optimization algorithm.

III. Formulation of Trajectory Optimization Problem

1) State Equation: Let us define the state variable as

$$\boldsymbol{x}_k = \begin{bmatrix} \boldsymbol{p}_k \\ \boldsymbol{q}_k \\ \boldsymbol{v}_k \\ \boldsymbol{L}_k \\ t_k \\ \{\boldsymbol{p}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{q}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \end{bmatrix} \tag{11}$$

and the control input (depending on the type of parametrization used) as

$$\boldsymbol{u}_k = \begin{bmatrix} \tau_k \\ \{\boldsymbol{v}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{\omega}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{f}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{\eta}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \end{bmatrix} \quad \boldsymbol{u}_k = \begin{bmatrix} \tau_k \\ \{\boldsymbol{v}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{\omega}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\lambda_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\boldsymbol{r}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \\ \{\hat{\boldsymbol{\eta}}_{l,k}\}_{l \in 1,\ldots,n_{\mathrm{e}}} \end{bmatrix} \tag{12}$$

The reason why we define the end velocities as control inputs and not end poses is that in this way we can impose contact complementarity by simply assigning large cost to the velocity of the in-contact ends, as described later. Moreover, t_k denotes the absolute time instant at the beginning of the k-th contact phase and τ_k denotes the duration of the k-th phase, for which

$$t_{k+1} = t_k + \tau_k \tag{13}$$

holds. The movement of each end is expressed as

$$\boldsymbol{p}_{l,k+1} = \boldsymbol{p}_{l,k} + \boldsymbol{v}_{l,k}\tau_k, \quad \boldsymbol{q}_{l,k+1} = q(\boldsymbol{\omega}_{l,k}\tau_k) \cdot \boldsymbol{q}_k \tag{14}$$

where $\boldsymbol{p}_{l,k}$ and $\boldsymbol{q}_{l,k}$ denote the position and orientation of the l-the end with respect to the global coordinate frame, while $\boldsymbol{v}_{l,k}$ and $\boldsymbol{\omega}_{l,k}$ denote its linear and angular velocity.

The equations defined above are integrated into the following state transition equation.

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k) \tag{15}$$

Here, $\boldsymbol{v}(t)$ that appears in the right hand side of (7c) is eliminated by substituting (7b).

A. Task-related Costs

The following task-related cost function is defined.

$$L_{\mathrm{task},k} = \frac{1}{2}\|W_k^x(\boldsymbol{x}_k - \boldsymbol{x}_k^{\mathrm{ref}})\|^2 + \frac{1}{2}\|W_k^u(\boldsymbol{u}_k - \boldsymbol{u}_k^{\mathrm{ref}})\|^2 \tag{16}$$

Here, $(*)^{\mathrm{ref}}$ are desired values. We consider a waypoint-tracking task where a series of intermediate configurations (waypoints) are specified for the CoM, the base link, and the ends. Desired position and velocity values are given by values along spline curves connecting these waypoints. The desired stiffness values are determined by analytically solving the following least squares problem for each k:

$$\min \ \left\|\textstyle\sum_l \lambda_{l,k}^2\right\|^2 \ \text{ subject to } \ \textstyle\sum_l \lambda_{l,k}^2(\boldsymbol{p}_k^{\mathrm{ref}} - \boldsymbol{p}_{l,k}^{\mathrm{ref}}) = \boldsymbol{g}$$

This subproblem gives the stiffness distribution that supports the CoM against gravity. The desired values of the CMP offset and the moment of each end are both set as $\mathbf{0}$ by default, but they could be assigned non-zero values to generate certain characteristic motion. The weight parameters of each term, w_*, must be specified by the designer. For variables defined as quaternions, the difference between the variable and the reference value is defined as $\boldsymbol{q} - \boldsymbol{q}^{\mathrm{ref}} := \omega(\boldsymbol{q}^{\mathrm{ref}^{-1}} \cdot \boldsymbol{q})$, where $\omega(\cdot)$ transforms a unit quaternion into an equivalent angle-axis vector.

B. Inequality Constraints

In Subsections III-B and III-C, the subscript k is omitted for simplicity of notation. The following box constraint is imposed on the position of each end relative to the CoM and the base link:

$$\boldsymbol{p}_{l,\min} \le \boldsymbol{q}^{-1}(\boldsymbol{p}_l - \boldsymbol{p}) \le \boldsymbol{p}_{l,\max} \ \ \forall l \tag{17}$$

In addition, simple range constraints are imposed on the duration and the stiffness of each end.

$$\tau_{\min} \le \tau \le \tau_{\max} \tag{18}$$

$$0 \le \lambda_l \le \lambda_{\max} \ \ \forall l \tag{19}$$

The contact wrench that acts on each end must satisfy the non-slip condition and the limitation of moment. First, the non-slip condition is expressed as

$$\sqrt{f_{l,\mathrm{x}}^2 + f_{l,\mathrm{y}}^2} \le \mu f_{l,\mathrm{z}} \tag{20}$$

where μ is the static friction coefficient. Moreover, constraints on the contact moment are defined as follows.

$$-c_{\max,\mathrm{x}} f_{l,\mathrm{z}} \le \eta_{l,\mathrm{x}} \le -c_{\min,\mathrm{x}} f_{l,\mathrm{z}} \tag{21a}$$

$$c_{\min,\mathrm{y}} f_{l,\mathrm{z}} \le \eta_{l,\mathrm{y}} \le c_{\max,\mathrm{y}} f_{l,\mathrm{z}} \tag{21b}$$

$$-\mu_{\mathrm{z}} f_{l,\mathrm{z}} \le \eta_{l,\mathrm{z}} \le \mu_{\mathrm{z}} f_{l,\mathrm{z}} \tag{21c}$$

Here, $\boldsymbol{c}_{\min}$ and $\boldsymbol{c}_{\max}$ define rectangular admissible range of the center-of-pressure, and μ_{z} is the coefficient of friction torque.

Inequality constraints defined above can be expressed in a general form $g(\boldsymbol{x}_k, \boldsymbol{u}_k) \ge \mathbf{0}$, where g is a differentiable vector-valued function and $\ge$ is evaluated componentwise. This inequality constraint is integrated into the overall optimization problem as the following log-barrier cost.

$$L_{\mathrm{limit}}(\boldsymbol{x}_k, \boldsymbol{u}_k) = \textstyle\sum_{i=1}^{n_g} - \log\max(\epsilon, g_i(\boldsymbol{x}_k, \boldsymbol{u}_k))$$

where n_g is the dimension of g. Clearly, L_{limit} becomes greater as g_i approaches 0. Taking $\max$ with a small constant ϵ ensures that the infeasibility of $\log$ is avoided if g_i becomes negative during optimization.

C. Contact-dependent Costs

Let $\sigma_{l,k}$ denote the contact state of the l-th end at the k-th step; namely, $\sigma_{l,k} = i$ if the l-th end is in contact with the i-th contact face during $[t_k, t_{k+1}]$, and $\sigma_{l,k} = \varnothing$ if it is off contact. The following cost is used to impose the complementarity of the end velocity and contact wrench.

$$\begin{aligned} J_{\mathrm{compl},k} = w_{\mathrm{compl}}^2 \textstyle\sum_l \Big[\sum_i \big(\delta[\sigma_{l,k} = i](\boldsymbol{\eta}_i^{\mathsf{T}}(\boldsymbol{p}_{l,k} - \boldsymbol{o}_i))^2\big) \\ + \delta[\sigma_{l,k} \ne \varnothing](\|\boldsymbol{v}_{l,k}\|^2 + \|\boldsymbol{\omega}_{l,k}\|^2) + \delta[\sigma_{l,k} = \varnothing]\lambda_{l,k}^2 \Big] \end{aligned} \tag{22}$$

Here, $\delta[*]$ takes 1 if the condition inside the bracket is true, and 0 otherwise. The first term requires that the distance between the l-th end and the i-th contact face in the normal direction must be zero if they are in contact. Here, $\boldsymbol{o}_i$ and $\boldsymbol{\eta}_i$ are the origin and the normal of the i-th contact face, respectively. The second term requires that the end velocity must be zero if it is in contact with one of the contact faces. The third term requires that the end stiffness must be zero if it is not in contact with any of the contact faces. Note that constraining the stiffness to zero enforces the corresponding contact wrench to zero by definition. The weight w_{compl} must set large enough in order to make the error of the complementarity conditions acceptably small after optimization.

D. Formulation of Optimal Control Problem and Solution Method

The overall cost function is defined as

$$J[\boldsymbol{\sigma}] = \sum_k \left[L_{\mathrm{task},k} + L_{\mathrm{limit},k} + L_{\mathrm{compl},k}[\boldsymbol{\sigma}_k]\right] \tag{23}$$

and the planning problem is formulated as the following optimal control problem:

$$\begin{aligned} &\text{find } \boldsymbol{x}, \boldsymbol{u} \text{ that minimizes } J[\boldsymbol{\sigma}](\boldsymbol{x}, \boldsymbol{u}) \\ &\text{subject to } \boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k) \end{aligned} \tag{24}$$

Because the inequality constraints are integrated in to the cost function as a barrier function as described earlier, the overall optimal control problem is unconstrained and therefore differential dynamic programming (DDP) can be used for computing its locally optimal solution. Methods proposed in [37, 38] could be used for more strict treatment of inequality constraints. At initialization, the decision variables are reset to their desired values described in Section III-A. Here, the initial trajectory does not necessarily satisfy the state equation. A technique proposed in F-DDP [39] was used to correct the error of the state equation iteratively in the course of optimization. The error contraction rate was set as 0.5. The gradient of the dynamical model and the cost function was derived analytically and implemented in the program code. Automatic differentiation techniques could be used to reduce the manual effort of this step.

The variables are scaled based on their physical dimensions so that all variables fall within the range of the same order of magnitude. As a result of proper scaling of variables, the convergence of iterative optimization is improved, and moreover, setting all weights of the cost fuction to one becomes a good starting point for weight adjustment. In fact, for examples shown in the next section, all diagonal elements of the weight matrice W_x and W_u are set as 1 unless otherwise noted. The only exception is the complementarity weight w_{compl}, which was set as 1000.

IV. Trajectory Optimization and Tracking Examples

A. Common Setup

For both planning and simulation, a laptop computer with Intel Core-i7 1280P CPU was used. The proposed trajectory optimization method was implemented in custom C++ code without using any off-the-shelf solvers and run in a single thread.

B. Trajectory Optimization Results

Trajectory optimization results of bipedal and quadrupedal locomotion with the stiffness-based and direct parametrization are shown in Fig. 1. Contact patterns used to generate these motions are listed in Table I. Each contact pattern is shown as a matrix whose (l, k) component indicates the contact state of the l-th end at the k-th step. A numeral indicates the contact face index whereas the symbol '-' indicates the off-contact state. The biped model has two ends: the right foot (1) and the left foot (2), while the quadruped model has four ends: the right front, the left front, the rear right, and the rear left (IDs are assigned in this order). The continuous trajectory of each end while it is in the off-contact state is defined as a cycloid-like curve. It is for visualization and for providing desired values to the trajectory-tracking controller described later.

TABLE I
Contact sequences

Task	N	contact sequence
Walk	24	`000-000-000-000-000-000-0` `0-000-000-000-000-000-000`
Run	24	`00---0---0---0---0---0--0` `0--0---0---0---0---0---00`
Long-jump	6	`000-000` `000-000`
Back-flip	6	`000-111` `000-111`

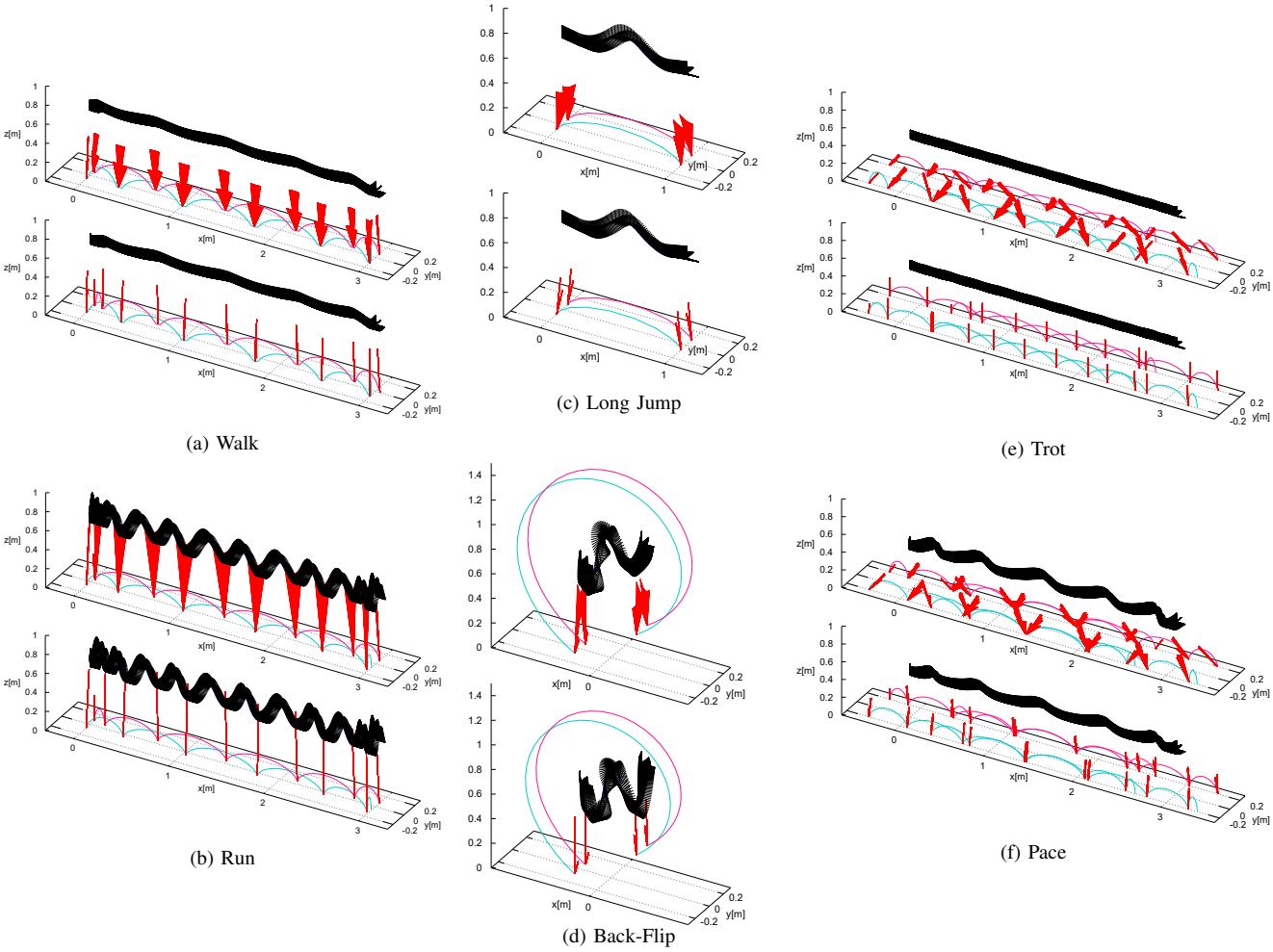
Trot	24	`000-000-000-000-000-000-0`
		`0-000-000-000-000-000-000`
		`0-000-000-000-000-000-000`
		`000-000-000-000-000-000-0`

Pace	24	`000-000-000-000-000-000-0`
		`0-000-000-000-000-000-000`
		`000-000-000-000-000-000-0`
		`0-000-000-000-000-000-000`

For **Walk** and **Run**, the weights on the CoM velocity and the angular momentum were raised to 10. As demonstrated in **Run**, the SBCD can express flight phases together with other contact phases seamlessly, and it can generate vertical CoM movement by varying the end stiffness. In **Long Jump**, to realize small launch angle under the friction constraint, the CMP offset was utilized to shift the direction of the contact force away from the CoM. As a result, a linear CoM trajectory with long horizontal jump distance (1.0m) with low vertical jump height (0.13m) together with rotational movement about the y (pitch) axis consistent with the dynamics was generated. The CMP offset was also utilized in **Trot** and **Pace** to satisfy the friction constraint. In **Back-flip**, desired orientation (-45deg and $+45$deg resp.) and angular velocity (-10rad/s) of the base link at lift-off and landing were specified to create rotation exceeding 180 degrees. Note that these reference values are used mainly for initialization of the trajectory, and optimized values were different. Similar to **Long Jump**, the CMP offset was utilized to generate required angular momentum for back-flip. The landing position was set 0.5m behind and 0.3m below the lift-off position.

C. Comparison of Contact Wrench Parametrization

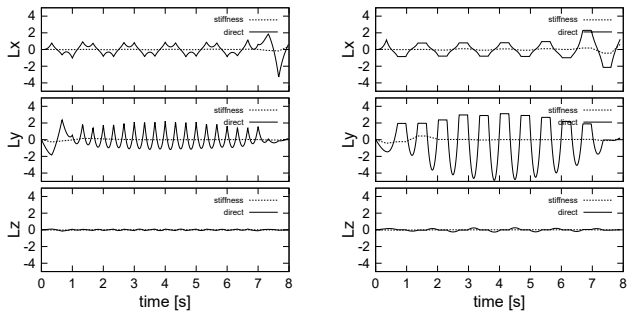
Overall, both formulations (stiffness-based and direct) could be used to generate feasible trajectories of all demonstrated examples. However, trajectories generated with the direct parametrization showed oscillation of angular momentum as shown in Fig. 2. This oscillation is inevitable in the direct parametrization because the contact wrench is fixed throughout each integration step. The magnitude of oscillation may be reduced by subdividing each contact phase into smaller intervals, but this will increase the computation cost. Similar oscillation was observed in the **Run** and **Pace** examples too.



(a) Walk (c) Long Jump (e) Trot

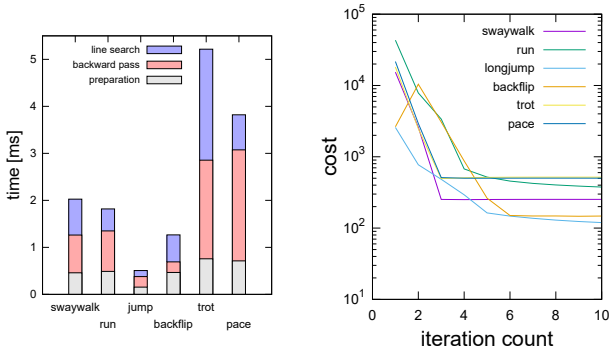
(b) Run (d) Back-Flip (f) Pace

Fig. 1. Trajectory optimization results. Each figure shows stiffness-based (top) and direct (bottom). The CoM trajectory is depicted by a black curve with sticks indicating the base link rotation. The swing-foot trajectories are depicted by cyan (right) and magenta (left) curves. The ground reaction force is depicted by red sticks.



Walk Run

Fig. 2. Angular momentum profile of walk and run , stiffness-based (dashed) and direct (solid).



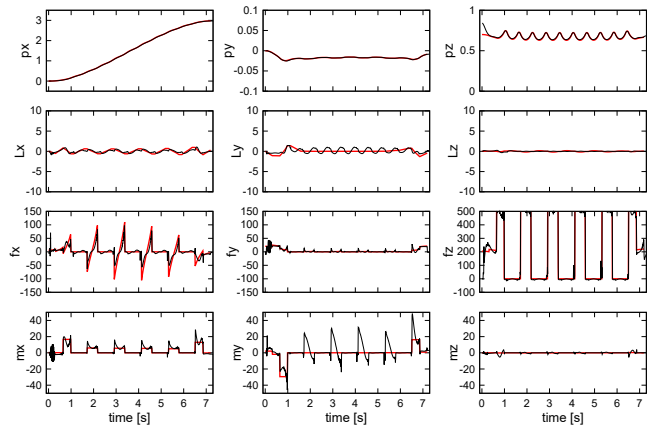
(a) Average computation time per iteration (b) Cost curve

Fig. 3. Computational characteristics

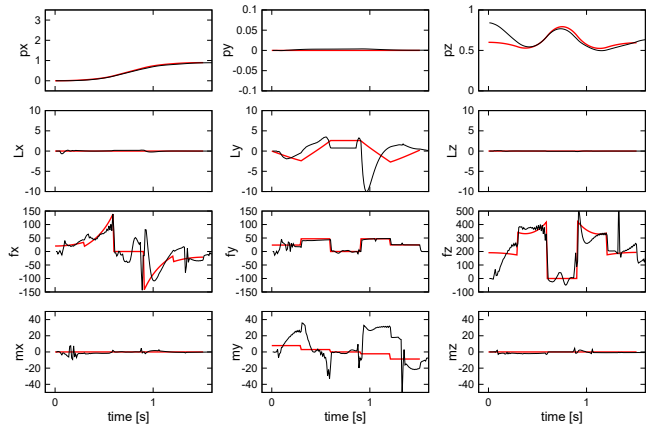
From this we consider that the stiffness-based formulation is more useful for trajectory generation with long time steps.

D. Computational Characteristics

Computation time for one iteration (average of 100 iterations) is summarized in Fig. 3(a). The total computation time is further broken down into preparation, backward pass of DDP, and line search. The preparation phase mainly consists of computation of the gradient of the state equation and the gradient and Hessian of the cost function. The backward pass of DDP consists of computation of the gradient and Hessian



(a) Run



(b) Long-jump

Fig. 4. Comparison of reference trajectory (red) and simulation (black). From top to bottom: CoM position, angular momentum, contact force, and contact moment of the right foot.

of the value function. Line search is needed to ensure stable convergence and it includes the forward pass of DDP. The computation cost of DDP is cubic to the state dimension n_x and the input dimension n_u, and proportional to the number of steps N. The computation time of the quadruped examples were greater than the biped ones because the quadruped model has greater state and input dimensions that the biped model. Direct comparison of computation time with other existing studies is difficult because of differences in n_x, n_u, and N. Generally speaking, existing studies that reported small computation time used models with small dimensions. Similar dimensionality reduction may also be done in the present formulation by assuming point contact instead of surface contact. Fig. 3(b) shows curves of optimized cost against number of iterations. For all examples, the cost decreased very close to a local minimum in less than five iterations. Some examples showed slower convergence speed in the vicinity of a local minimum, and took up to 10 to 20 iterations for convergence. The total computation cost is well below 100ms.

E. Trajectory Tracking Using Whole-body MPC

To test if reference trajectories generated by using the SBCD can be tracked under realistic kino-dynamic constraints, trajectory tracking control using a full-DoF model was performed in rigid-body simulation. Whole-body model predictive control was used to implement reference trajectory tracking. The detail of the whole-body MPC controller used here is not described in detail due to limited space; it is based on the centroidal dynamics and full kinematics formulation similar to [40]. Choreonoid with the AIST Simulator was used for simulation environment. The humanoid robot model used for simulation has 30 joints and its total mass is 44[kg]. To work around the difficulty of whole-body state estimation, the full state of the robot was directly obtained from the simulator. Figures 4(a)(b) show plots of the time-series of representative variables for **Run** and Long Jump. The robot could successfully perform both motions. See the attached video for more simulation results. For both examples, the reference CoM trajectory as well as the reference ground reaction force could be tracked accurately. For **Run**, weak oscillation of L_{y} was observed. This is because variation in angular momentum caused by leg swinging could not be compensated completely by whole-body motion such as arm swinging. For **Long Jump**, large error in L_{y} immediately after landing was observed. This was caused by large deviation of the lateral component of the ground reaction force (f_{x}) after landing. Although tracking performance could be improved by implementing better reaction force control, it is beyond the scope of this letter. In both cases, variation of contact moment to reduce the error of angular momentum was observed.

V. Conclusion

This paper proposed a trajectory generation method for legged robots based on a closed-form solution of the linear centroidal dynamics model. It has been demonstrated in several examples that the proposed method can be used to generate trajectories that involve both linear and rotational movement of centroidal dynamics. Our future work includes application of the proposed model to capturability analysis including angular momentum and synthesis of fall avoidance controllers.

[1] G. Bledt, P. M. Wensing, and S. Kim. Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the MIT Cheetah. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4102–4109, 2017.

[2] H. Li, R. J. Frei, and P. M. Wensing. Model hierarchy predictive control of robotic systems. *IEEE Robotics and Automation Letters*, 6(2):3373–3380, 2021.

[3] Z. Zhou, B. Wingo, N. Boyd, S. Hutchinson, and Y. Zhao. Momentum-aware trajectory optimization and control for agile quadrupedal locomotion. *IEEE Robotics and Automation Letters*, 7(3):7755–7762, 2022.

[4] D. E. Orin, A. Goswami, and S. H. Lee. Centroidal dynamics of a humanoid robot. *Autonomous Robots*, 35(2-3):161–176, 2013.

[5] S. Kajita and K. Tani. Study of dynamic biped locomotion on rugged terrain - derivation and application of the linear inverted pendulum mode -. In *IEEE International Conference on Robotics and Automation*, pages 1405–1411, 1991.

[6] S. Caron. Biped stabilization by linear feedback of the variable-height inverted pendulum model. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9782–9788, 2020.

[7] K. Guan, K. Yamamoto, and Y. Nakamura. Virtual-mass-ellipsoid inverted pendulum model and its applications to 3d bipedal locomotion on uneven terrains. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1401–1406, 2019.

[8] J. Ding, C. Zhou, S. Xin, X. Xiao, and N. Tsagarakis. Nonlinear model predictive control for robust bipedal locomotion: exploring angular momentum and com height changes. *Advanced Robotics*, 35(18):1079–1097, 2021.

[9] G. Gibson, O. Dosunmu-Ogunbi, Y. Gong, and J. Grizzle. Terrain-adaptive, alip-based bipedal locomotion controller via model predictive control and virtual constraints. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6724–6731, 2022.

[10] F. M. Smaldone, N. Scianca, L. Lanari, and G. Oriolo. From walking to running: 3d humanoid gait generation via mpc. *Frontier in Robotics and AI*, 9, 2022.

[11] I. Poulakakis and J. W. Grizzle. The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper. *IEEE Transactions on Automatic Control*, 54(8):1779–1793, 2009.

[12] I. Mordatch, M. de Lasa, and A. Herzmann. Robust physics based locomotion using low-dimensional planning. *ACM Trans. Graphics, Proc. ACM SIGGRAPH 2010*, 29(4), 2010.

[13] T. Apgar, P. Clary, K. Green, A. Fern, and J. Hurst. Fast online trajectory optimization for the bipedal robot cassie. In *Robotics: Science and Systems*, 2018.

[14] J. Englsberger, G. Mesesan, and C. Ott. Smooth trajectory generation and push-recovery based on divergent component of motion. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4560–4567, 2017.

[15] T. Kamioka, H. Kaneko, T. Takenaka, and T. Yoshiike. Simultaneous optimization of zmp and footsteps based on the analytical solution of divergent component of motion. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1763–1770, 2018.

[16] R. Tedrake, S. Kuindersma, R. Deits, and K. Miura. A closed-form solution for real-time zmp gait generation and feedback stabilization. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 936–940, 2015.

[17] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *IEEE Int. Conf. Robotics and Automation*, 2003.

[18] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl. Online walking motion generation with automatic footstep placement. *Advanced Robotics*, 24(5-6):719–737, 2010.

[19] M. Murooka, M. Morisawa, and F. Kanehiro. Centroidal trajectory generation and stabilization based on preview control for humanoid multi-contact motion. *IEEE Robotics and Automation Letters*, 7(3):8225–8232, 2022.

[20] 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 the atlas humanoid robot. *Autonomous Robots*, 40(3):429–455, 2016.

[21] J. Carpentier and N. Mansard. Multicontact locomotion of legged robots. *IEEE Transactions on Robotics*, 34(6):1441–1460, 2018.

[22] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters*, 3(3):1560–1567, 2018.

[23] T. Kwon, Y. Lee, and M. Van De Panne. Fast and flexible multilegged locomotion using learned centroidal dynamics. *ACM Trans. Graph.*, 39(4), 2020. ISSN: 0730-0301.

[24] Y. Ding, C. Khazoom, M. Chignoli, and S. Kim. Orientation-aware model predictive control with footstep adaptation for dynamic humanoid walking. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 299–305, 2022.

[25] L. Rossini, E. M. Hoffman, S. H. Bang, L. Sentis, and N. G. Tsagarakis. A real-time approach for humanoid robot walking including dynamic obstacles avoidance. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pages 1–8, 2023.

[26] H. Dai and R. Tedrake. Planning robust walking motion on uneven terrain via convex optimization. In *IEEE-RAS International Conference on Humanoid Robots*, pages 579–586, 2016.

[27] P. Fernbach, S. Tonneau, O. Stasse, J. Carpentier, and M. Taïx. C-croc: continuous and convex resolution of centroidal dynamic trajectories for legged robots in multicontact scenarios. *IEEE Transactions on Robotics*, 36(3):676–691, 2020.

[28] L. Clemente, O. Villarreal, A. Bratta, M. Focchi, V. Barasuol, G. G. Muscolo, and C. Semini. Foothold evaluation criterion for dynamic transition feasibility for quadruped robots. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4679–4685, 2022.

[29] H. Audren, J. Vaillant, A. Kheddar, A. Escande, K. Kaneko, and E. Yoshida. Model preview control in multi-contact motion-application to a humanoid robot. In *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 4030–4035, 2014.

[30] Y. Tazaki. Fast multi-contact motion planning based on best-neighbor search of contact sequences. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 277–284, 2022.

[31] 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*, pages 653–659, 2008.

[32] 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*, 24(12):1013–1032, 2005.

[33] J. Englsberger, C. Ott, and A. Albu-Schäffer. Three-dimensional bipedal walking control based on divergent component of motion. *IEEE Trans. Robotics*, 31(2):355–368, 2015.

[34] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim. Dynamic locomotion in the MIT Cheetah 3 through convex model-predictive control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1–9, 2018.

[35] S.-H. Lee and A. Goswami. Reaction mass pendulum (rmp): an explicit model for centroidal angular momentum of humanoid robots. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 4667–4672, 2007.

[36] J. Shen and D. Hong. Convex model predictive control of single rigid body model on so(3) for versatile dynamic legged motions. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6586–6592, 2022.

[37] A. Jordana, S. Kleff, A. Meduri, J. Carpentier, N. Mansard, and L. Righetti. Stagewise implementations of sequential quadratic programming for model-predictive control. hal-04330251, 2023.

[38] W. Jallet, A. Bambade, E. Arlaud, S. El-Kazdadi, N. Mansard, and J. Carpentier. Proxddp: proximal constrained trajectory optimization. hal-04332348, 2023.

[39] C. Mastalli, R. Budhiraja, W. Merkt, G. Saurel, B. Hammoud, M. Naveau, J. Carpentier, L. Righetti, S. Vijayakumar, and N. Mansard. Crocoddyl: an efficient and versatile framework for multi-contact optimal control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2536–2542, 2020.

[40] J. Sleiman, F. Farshidian, M. Minniti, and M. Hutter. A unified mpc framework for whole-body dynamic locomotion and manipulation. *IEEE Robotics and Automation Letters*, 6(3):4688–4695, 2021.

A. Derivation of Stiffness-based Centroidal Dynamics

By substituting (2)(3) into (1a) and assuming that ϵ is small enough, we obtain

$$\begin{aligned}\ddot{\boldsymbol{p}} &= \textstyle\sum_l \lambda_l^2(\boldsymbol{p} - (\boldsymbol{p}_l + \boldsymbol{r}_l)) - \boldsymbol{g} \\ &\approx \textstyle\sum_l \lambda_l^2(\boldsymbol{p} - (\boldsymbol{p}_l + \boldsymbol{r}_l)) - \boldsymbol{g} + \epsilon^2\boldsymbol{p} \\ &= \left(\textstyle\sum_l \lambda_l^2 + \epsilon^2\right)\boldsymbol{p} - \left(\textstyle\sum_l \lambda_l^2(\boldsymbol{p}_l + \boldsymbol{r}_l) + \boldsymbol{g}\right) \\ &= \bar{\lambda}^2(\boldsymbol{p} - \bar{\boldsymbol{p}} - \bar{\boldsymbol{r}})\end{aligned}$$

The rotational dynamics is derived as follows. Substituting (2)(3) into (1b) yields

$$\begin{aligned}\dot{\boldsymbol{L}} &= \textstyle\sum_l((\boldsymbol{p}_l - \boldsymbol{p}) \times m\lambda_l^2(\boldsymbol{p} - \boldsymbol{p}_l - \boldsymbol{r}_l) + m\lambda_l^2\hat{\boldsymbol{\eta}}_l) \\ &= \textstyle\sum_l(\boldsymbol{p} - \boldsymbol{p}_l) \times m\lambda_l^2\boldsymbol{r}_l + \sum_l m\lambda_l^2\hat{\boldsymbol{\eta}}_l \\ &= \boldsymbol{p} \times m\bar{\lambda}^2\bar{\boldsymbol{r}} + \textstyle\sum_l m\lambda_l^2(\hat{\boldsymbol{\eta}}_l - \boldsymbol{p}_l \times \boldsymbol{r}_l) \\ &\approx (\ddot{\boldsymbol{p}} + \bar{\lambda}^2(\bar{\boldsymbol{p}} + \bar{\boldsymbol{r}})) \times m\bar{\boldsymbol{r}} + \textstyle\sum_l m\lambda_l^2(\hat{\boldsymbol{\eta}}_l - \boldsymbol{p}_l \times \boldsymbol{r}_l) \\ &= m(\ddot{\boldsymbol{p}} \times \bar{\boldsymbol{r}} + \bar{\boldsymbol{\eta}})\end{aligned}$$