



Centroidal-momentum-based trajectory generation for legged locomotion

Chuanzheng Li^a, Yanran Ding^a, Hae-Won Park^{b,*}

^a University of Illinois at Urbana-Champaign, Champaign, IL 61801 USA

^b Korea Advanced Institute of Science and Technology, Korea

ARTICLE INFO

Keywords:

Centroidal momentum
Trajectory optimization
Legged locomotion

ABSTRACT

This paper presents a trajectory optimization framework for planning dynamic legged locomotion based on a robot's centroidal momentum (CM), which is the aggregation of all the links' momenta at the robot's Center of Mass (CoM). This new framework is built around CM dynamic model driven by Ground Reaction Forces (GRFs) parameterized with Bézier polynomials. Due to the simple form of CM dynamics, the closed-form solution of the robot's CM can be obtained by directly integrating the Bézier polynomials of GRFs. The CM can be also calculated from the robot's generalized coordinates and velocities using Centroidal Momentum Matrices (CMM). For dynamically feasible motions, these CM values should match, thereby providing equality constraints for the proposed trajectory optimization framework. Direct collocation methods are utilized to obtain feasible GRFs and joint trajectories simultaneously under kinematic and dynamic constraint. With the closed-form solutions of CM due to the parameterization of GRFs in the formulation, numerical error induced by collocation methods in the solution of trajectory optimization can be reduced, which is crucial for reliable tracking control when applied to real robotic systems. Using the proposed framework, jumping trajectories of legged robots are obtained in the simulation. Experimental validation of the algorithm is performed on a planar robot testbed, proving the effectiveness of the proposed method in generating dynamic motions of the legged robots.

1. Introduction

Agile quadrupeds such as cats and squirrels are capable of performing highly dynamic maneuvers over a variety of terrains. When navigating in challenging environments, they can plan trajectories that fully utilize their physical capabilities and inherent dynamics. Such abilities, if successfully implemented on robots, possess profound potential in scenarios such as hazardous environment reconnaissance, disaster response and rescue.

Recent advances in quadruped robots has shown promising results in matching the dynamic capabilities of their natural counterparts: robust walking gaits have been performed by the BigDog [1], Spot and SpotMini from Boston Dynamics, though details of their control algorithms are yet to be published. ANYmal [2] is able to perform stable locomotion such as walking and trotting. Park et al. [3] displayed robust bounding motion on the MIT Cheetah 2 for a wide range of speeds in untethered 3D tests. Similar maneuverability has also been achieved on HyQ2Max [4] in creating trotting and self-righting while using hydraulic actuation. As the performance of the quadrupedal robots keeps advancing, so is the need for accurate algorithms capable of planning trajectories that fully utilize a robot's dynamical capabilities to create

agile motions, or to navigate challenging environments while respecting the robot's physical restrictions.

The impediment to fast trajectory planning is partly rooted in the high degrees of freedom (DoF) present in robots. Methods based on a robot's full-body dynamics have proven effectiveness in producing expressive motions, as shown in [5–8], but would often suffer from long computational time. As a work-around, simplified models including the point-mass model [9] and other ones stemming from the linear inverted pendulum (LIP) model introduced by [10] have been used, such as the Reaction Mass Pendulum (RMP) [11] which augment the LIP models with a reaction-mass ellipsoid to capture the change in centroidal momenta for tasks such as balancing. A modified Spring Loaded Inverted Pendulum (SLIP) model is used by [12] to achieve running long jumps. Though less restrictive, these models still require task-specific modifications to its dynamic models to complement the full-body motions, or part of the dynamics will be left undecided or uncontrolled.

In recent years, the centroidal momentum (CM) of a robot, which is the aggregation of all links' momenta at the robot's Center of Mass (CoM) [13], has gained increasing attention as a reduced-order dynamic model in planning trajectories for robots with complex dynamics [14–17]. A robot's CM enjoys simple dynamics driven by the ground reaction forces

* Corresponding author.

E-mail addresses: cli67@illinois.edu (C. Li), yding35@illinois.edu (Y. Ding), haewonpark@kaist.ac.kr (H.-W. Park).

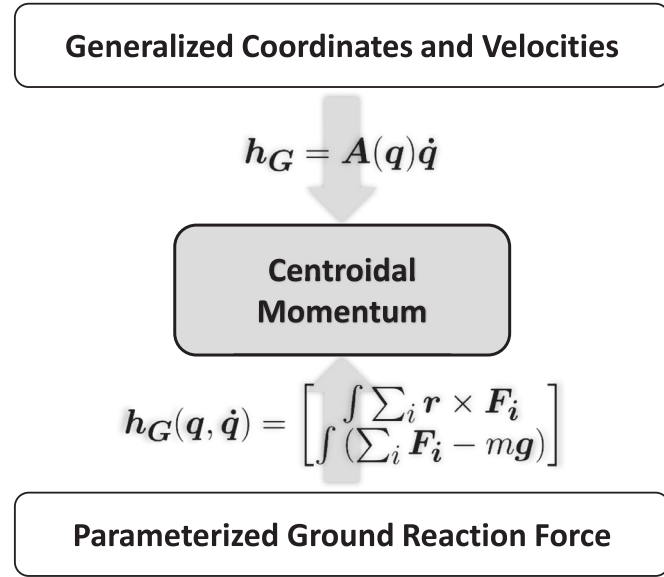


Fig. 1. A robot's centroidal momentum $\mathbf{h}_G$ can be calculated from its Centroidal Momentum Matrix $\mathbf{A}(\mathbf{q})$ and the generalized velocities $\dot{\mathbf{q}}$ (top), or through direct integration of its external wrenches created by GRFs the ground reaction forces $\mathbf{F}_i$ (bottom). For dynamically feasible trajectories, the two calculation results should match.

(GRF) in legged locomotion. By including the GRF as optimization variables, a robot's CM can be obtained through numerical integration, from which the corresponding joint angles can be recovered from kinematic equations without solving the complete second-order equation of motions. However, the use of CM dynamics in trajectory optimization for robots are often limited to a subsection of the optimization: either to validate transition feasibility between consecutive contact states [15], or to generate contact wrenches, which tends to rely on pre-defined contact locations and also requires additional process for joint trajectories [14]. Two notable exceptions come from the work of Dai et al. [17] and Fernbach et al. [15], where the contact states and forces are solved simultaneously as a hard linear complementarity problem. This method is shown to produce expressive motions and allows automatic gait scheduling, but often at the cost of long computation time when incorporating the full joint-space rigid body dynamics, or loss of generality.

In this paper, we propose to augment the CM-dynamic-based trajectory optimization scheme by parameterizing the GRF and swing leg trajectories with Bézier polynomials. The optimizer simultaneously selects the Bézier polynomial coefficients and the contact leg's joint trajectories, under an equality constraint which unifies the CM directly integrated from GRF through equations of motion, to the CM calculated from the robot's generalized states and velocities using Centroidal Momentum Matrix [13], as shown in Fig. 1. This ensures that the obtained CM always respect the robot's full-body dynamics, even between optimization nodes, unlike other collocation-based methods which approximate the solution of the model using polynomials. A similar approach can be found in [14], but is restricted to using a basic power basis polynomial for the CoM and GRF only, and relies on additional processing for the joint trajectories which requires prior knowledge of the exact contact locations.

The remainder of this paper is organized as follows: Section 2.1 describes the planar robot model used in formulating the proposed trajectory optimization algorithm. Section 2.2 presents two ways of obtaining a robot's centroidal momentum with ground contact. Detailed formulation of the proposed multi-phase optimization framework, including transition map between different motion phases is presented in Section 3. Experimental validations are presented in Section 4 with reference trajectories obtained from the proposed method.

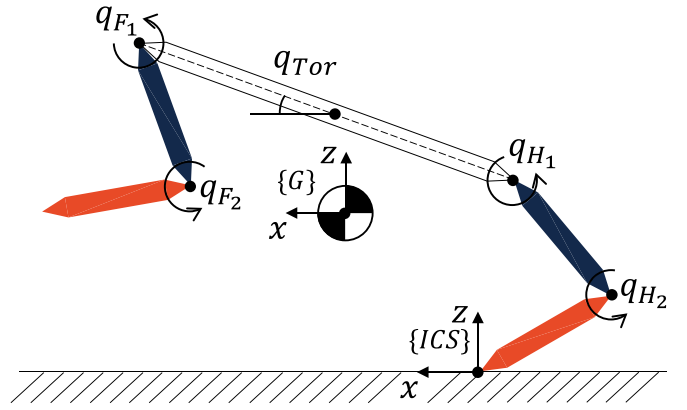


Fig. 2. The five-link planar model used in this paper. The inertia coordinate system (ICS) is earth fixed at the model's contacting leg. The model's CoM frame {G} locates at its CoM, axes parallel to that of {ICS}.

Table 1

The physical parameters of the planar model used in trajectory optimization. All the link values are expressed in terms of a set of reference values: total mass m_0 , torso length l_0 , and the moment of inertia J_0 obtained by lumping all the leg mass onto the torso link.

Ref. values	$m_0 = 1.9 \text{ kg}, l_0 = 0.3 \text{ m}, J_0 = 1/12 m_0 l_0^2$		
Link	$M_{\text{tor}} = 2/3 m_0$	$M_1 = 1/6 m_0$	$M_2 = 1/60 m_0$
Values	$L_{\text{tor}} = l_0$	$L_1 = 1/2 l_0$	$L_2 = 1/2 l_0$
	$J_{\text{tor}} = 2/3 J_0$	$J_1 = 10^{-4} J_0$	$J_2 = 10^{-4} J_0$

2. Centroidal dynamics

2.1. Simplified planar model

In this paper, a planar model shown in Fig. 2 is adopted to study a quadruped's behavior in its sagittal plane, where the most of dynamic motions happen. The model consists of five massive links, with the torso link in the middle connecting two identical two-link legs through ideal revolute joints. The physical parameters of the planar model specified in Table 1.

The configuration of this five-link body can be described with its *body angles*:

$$\mathbf{q}_b = (q_{\text{Tor}}, q_{F1}, q_{F2}, q_{H1}, q_{H2})^T.$$

where subscripts *F* and *H* stands for front and hind, 1 and 2 stand for the hip and knee joints respectively.

When unconstrained, the configuration space Q_e of the model's dynamics is seven-dimensional. To fully describe the model in the inertia coordinate system (ICS), the Cartesian position of its floating base is needed. Unlike the common practice in floating-body dynamics where the base is chosen as the torso, here we use the model's CoM position denoted as $(x_{\text{CoM}}, z_{\text{CoM}})$. The resulting *general coordinate* of the model is:

$$\mathbf{q} = (x_{\text{CoM}}, z_{\text{CoM}}, \mathbf{q}_b^T)^T \in Q_e$$

Here Q_e can be decomposed as:

$$Q_e = G \times S^4 \quad (1)$$

where G is a Lie group formed by $(x_{\text{CoM}}, z_{\text{CoM}}, q_{\text{Tor}})$, S^4 is the *shape space* where the four joint angles evolve [18].

This choice of floating-base facilitates the direct use of CoM trajectory information and is more natural to the proposed centroidal-momentum-based optimization framework, since the linear centroidal momentum of the model is directly related to its CoM trajectory, whereas the centroidal angular momentum can also be recovered from the *body angles* $\mathbf{q}_b$ and the joint velocities, more details in Section 2.3. This is especially conveniently during the aerial phase where the model

is not in contact with the ground, the CoM of which follows a ballistic trajectory with only gravity acting it. However, special treatment is required when the aerial phase ends with a ground impact.

2.2. Calculation of centroidal momentum

With the model and its generalized coordinate systems defined above, the full-body dynamics of the planar model can be readily derived through the standard Euler-Lagrange method. However, the resulting dynamics are highly coupled and nonlinear with up to fourteen states. As an alternative, we resort to studying the change of the model's centroidal momentum (CM), which enjoys a simple dynamics evolving under the wrenches created by the ground reaction forces (GRF) and its gravity. The CM can be also mapped from the joint velocities of the model using its Centroidal Momentum Matrix [19].

2.2.1. Centroidal momentum dynamics

The centroidal momentum of the planar model in Fig. 2 evolves under a net external wrench induced by gravity and the contact forces exerted by its environment:

$$\dot{k} = \sum_i (c_i - r) \times F_i \quad (2a)$$

$$\dot{l} = m\ddot{r} = \sum_i F_i - mg \quad (2b)$$

where l and k are the centroidal linear and angular momentum, the later is a scalar in the planar case. m is the total mass. g is the gravitational acceleration. F_i is GRF on the i th legs. r and c_i are the Cartesian position of the CoM and the contact point respectively. Note that $c_1 = 0$ as the ICS is defined at the first contact foot. The other contact locations can be calculated from joint angles. Due to its simple form, k and l can be easily obtained by integrating Eq. (2) given F_i .

We propose to parameterize F_i with M th order Bézier polynomials defined on interval $[0, T]$:

$$C(t) = \sum_{i=0}^M \alpha_i B_i^M(t), \quad t \in [0, T] \quad (3)$$

where α_i is the coefficients of the i th Bernstein basis $B_i^M(t)$, the derivative of which can be calculated using:

$$\frac{d}{dt} B_i^M(s) = \frac{M}{T} (B_{i-1}^{M-1}(s) - B_i^{M-1}(s)), \quad s \in [0, T] \quad (4)$$

[20]. This property will be used to solve for centroidal momentum using parameterized GRFs.

The benefits of using Bézier polynomials are two folds: 1) their start and end values only depend on the first/last coefficients, 2) the derivatives of Bézier polynomials at the start/end only depend on the first/last two coefficients. These properties are ideal for linking different dynamic phases as explained in Sections 3. They also allow easily increasing/decreasing the polynomials' order without modifying the start/end values and derivative values.

It can be shown that by parameterizing the GRF as Bézier polynomials, the corresponding CM also takes the form of Bézier polynomials, their coefficient in linear and bi-linear forms of the GRF coefficients and the model's initial conditions [21]. For simplicity, the following derivation assumes only one contact, but can be extended to multiple contacts as the effect of external wrenches can be combined linearly.

Centroidal linear momentum The horizontal and vertical GRF: F^x and F^z are parameterized using Bézier polynomials with coefficients $\alpha_{F_x}, \alpha_{F_z} \in \mathbb{R}^{M+1}$ respectively, the resulting linear momentum $l(\alpha_{l_x}, \alpha_{l_z})$ takes the form of two $(M+1)$ th order Bézier polynomials along the x and z direction. Given initial velocities $\dot{x}_0, \dot{z}_0$, the coefficients $\alpha_{l_x}, \alpha_{l_z}$ can be calculated using Eq. (4):

$$\Phi_{M+2,T} \alpha_{l_x} = \begin{bmatrix} \alpha_{F_x} \\ m\dot{x}_0 \end{bmatrix}, \quad \Phi_{M+2,T} \alpha_{l_z} = \begin{bmatrix} \alpha_{F_z} - mg \\ m\dot{z}_0 \end{bmatrix} \quad (5)$$

where $\Phi_{M+2,T} \in \mathbb{R}^{(M+2) \times (M+2)}$ is a coefficient matrix condensed from Eq. (4), its elements defined as:

$$\Phi_{M+2,T}(i, j) := \begin{cases} -M/T, & i = j = 1, 2, \dots, M+1 \\ M/T, & i = j-1 = 1, 2, \dots, M+1 \\ 1, & i = M+2, j = 1 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

The detailed derivation of $\Phi_{M+2,T}$ can be found in Appendix A. This matrix is always full rank thus invertible. Solving Eq. (5) yields the coefficients of the centroidal linear momentum as $\alpha_{l_x}(\alpha_{F_x}, \alpha_{F_z}), \alpha_{l_z}(\alpha_{F_x}, \alpha_{F_z})$, which are of $(M+1)$ th order. This lead to the closed-form expression of the centroidal linear momentum:

$$l(t, \alpha_{F_x}, \alpha_{F_z}), \quad t \in [0, T]. \quad (7)$$

Following a similar process, one can also obtain the CoM trajectory which are $(M+2)$ th order Bézier polynomials with $\alpha_x, \alpha_z \in \mathbb{R}^{M+3}$ being their coefficients along x and z direction respectively. Please refer to Appendix A for details on the derivation.

Centroidal Angular Momentum Similarly, the centroidal angular momentum k and its derivative $\dot{k}$ also take the form of a Bézier polynomial. The calculation for k starts from deriving $\dot{k}$ using Eq. (2a), which can be written into Bézier polynomial multiplication as:

$$\dot{k} = F^z(x_1 - x_{CoM}) - F^x(z_1 - z_{CoM}). \quad (8)$$

where $(x_1, z_1)^T$ is the contact location and can be calculated from the model's joint angles.

In Eq. (8), the first term of $F^z(x_1 - x_{CoM})$ is also a Bézier polynomial. Its coefficients, denoted as $\alpha_1 \in \mathbb{R}^{2M+3}$, can be obtained with the Bézier polynomial multiplication rule:

$$\alpha_1(i) = \sum_{j=\max(0, i-M-2)}^{\min(M, i)} C_{i,j} \alpha_{F_z}(i)(x_1 - \alpha_x(i-j)) \quad (9)$$

the coefficients $C_{i,j}$ are given by:

$$C_{i,j} = \frac{\binom{M}{j} \binom{M+2}{i-j}}{\binom{2M+2}{i}}, \quad (10)$$

where $\binom{M}{j} = \frac{M!}{j!(M-j)!}$. Please refer to Appendix A for the details on calculating CoM trajectory. The same calculation applies to the second term $F^x(z_1 - z_{CoM})$. Summing up the two terms yields the Bézier coefficients of $\dot{k}$, from which the Bézier polynomial of the centroidal angular momentum k can then be obtained following a similar procedure as Eq. (5). The centroidal angular momentum k is of order $(2M+3)$, its coefficients $\alpha_k(\alpha_{F_x}, \alpha_{F_z}) \in \mathbb{R}^{2M+4}$:

$$k(t, \alpha_{F_x}, \alpha_{F_z}), \quad t \in [0, T]. \quad (11)$$

In conclusion, by parameterizing the GRF as Bézier curves, we are able to obtain the closed-form solution of a robot's centroidal linear and angular momentum in the form of $(M+1)$ th-order and $(2M+3)$ th-order Bézier polynomials respectively, whose coefficients are in linear and bi-linear forms of the GRF coefficients and the model's initial conditions:

$$h_{G_a} = \begin{bmatrix} k(t, \alpha_{F_x}, \alpha_{F_z}) \\ l(t, \alpha_{F_x}, \alpha_{F_z}) \end{bmatrix}, \quad t \in [0, T]. \quad (12)$$

The subscript G_a indicates that its value is calculated from a set of coefficients.

Remark 1. Though this formulation is developed for sagittal motions, its extension to 3D scenarios does not involve fundamental changes, but rather replacing Eq. (2a) with a 3×1 vector, from which the centroidal angular momentum k (now a vector) can still be solved analytically with parameterized ground reaction forces F_i .

Table 2

Categories of system states for the four different dynamic configurations for the planar model. Hind stance is when the hind leg is in contact with the ground while its front leg swinging in the air, as shown in Fig. 2. The Swing states $\mathbf{q}_{sw}$ are chosen as the swing leg joints CoM trajectory (x_{CoM}, z_{CoM}) is chosen as the constrained states hence left out. Front stance is similar except that the swing and stance legs are swapped. Double stance is with both feet on the ground. Aerial phase is when non of the legs are in contact with the ground.

Type	Front Stance	Hind Stance	Double Stance*	Aerial**
$\mathbf{q}_c$ Constrained	x_{CoM}, z_{CoM}	x_{CoM}, z_{CoM}	$x_{CoM}, z_{CoM}, q_{H1}, q_{H2}$	—
$\mathbf{q}_s$ Swing states $\mathbf{q}_{sw}$	q_{H1}, q_{H2}	q_{F1}, q_{F2}	—	$q_{F1}, q_{F2}, q_{H1}, q_{H2}$
Stance states $\mathbf{q}_{st}$	q_{Tor}, q_{F1}, q_{F2}	q_{Tor}, q_{F1}, q_{F2}	q_{Tor}, q_{F1}, q_{F2}	$x_{CoM}, z_{CoM}, q_{Tor}$

*There are no Swing states during double stance phase. The choice of Stance states $\mathbf{q}_{st}$ can be either $(q_{Tor}, q_{F1}, q_{F2})$ or $(q_{Tor}, q_{H1}, q_{H2})$, and the Constrained states $\mathbf{q}_c$ should change accordingly. **During the aerial phase, the two legs all swing freely while the centroidal angular momentum remains the same. Note that the term "Stance states" is loosely used here to describe part of the system states that can not be parameterized, but are instead time-discretized using methods like forward Euler.

2.2.2. Centroidal momentum from joint velocities

By definition, the centroidal momentum of a robot can be calculated by summing up all the links' momenta with respect to the CoM. The result is a linear mapping from the joint velocities to the CM, as presented by [19] using Plücker coordinates. Here we briefly present its planar variation under the presence of contact constraints, and how the swing leg trajectories can also be parameterized with Bézier polynomials to further simplify the trajectory optimization problem.

The centroidal momentum of the planar model is a 3×1 vector given by:

$$\mathbf{h}_{G_q} = \begin{bmatrix} k(\mathbf{q}, \dot{\mathbf{q}}) \\ l(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix} = \mathbf{A}_G(\mathbf{q})\dot{\mathbf{q}} \quad (13)$$

where $\mathbf{A}_G(\mathbf{q}) \in \mathbb{R}^{3 \times 7}$ is the *centroidal momentum matrix* which linearly maps $\dot{\mathbf{q}}$ to $\mathbf{h}_{G_q}$. The subscript G_q indicates that its value comes from the generalized states.

Contact Constraints When a robot contacts with its environment at one or multiple locations, a set of constraints are enforced on the contacting feet:

$$\mathbf{J}\dot{\mathbf{q}} = \mathbf{0} \quad (14)$$

where $\mathbf{J} \in \mathbb{R}^{2c \times 7}$ is the contact feet's Jacobian, c is the number of independent contacts. The introduction of contact constraints reduces the dimension of the robot's configuration space, allowing to compute $\mathbf{h}_{G_q}$ with a reduced set of the general velocity $\dot{\mathbf{q}}$. To see this, one can partition Eq. (14) into:

$$\begin{bmatrix} \mathbf{J}_c & \mathbf{J}_s \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_c \\ \dot{\mathbf{q}}_s \end{bmatrix} = \mathbf{J}_c \dot{\mathbf{q}}_c + \mathbf{J}_s \dot{\mathbf{q}}_s = \mathbf{0} \quad (15)$$

where $\dot{\mathbf{q}}_c \in \mathbb{R}^{2c}$, denoted as *constrained velocities*, is a set of velocities related to the contact constraint, the corresponding states are referred to as the *constrained states* $\mathbf{q}_c$ often include the CoM position. $\dot{\mathbf{q}}_s \in \mathbb{R}^{7-2c}$ consists of the rest of the velocities, the corresponding states $\mathbf{q}_s$ includes the swing leg joints that are fully actuated and move freely in the air. $\mathbf{J}_c \in \mathbb{R}^{2c \times 2c}$ consists of the rows of $\mathbf{J}$ corresponding to $\dot{\mathbf{q}}_c$. Applying the same partition to Eq. (13) yields:

$$\mathbf{h}_{G_q} = [\mathbf{A}_{G_c} \quad \mathbf{A}_{G_s}] \begin{bmatrix} \dot{\mathbf{q}}_c \\ \dot{\mathbf{q}}_s \end{bmatrix} = \mathbf{A}_{G_c} \dot{\mathbf{q}}_c + \mathbf{A}_{G_s} \dot{\mathbf{q}}_s. \quad (16)$$

which can be combined with Eq. (15) to obtain:

$$\mathbf{h}_{G_q} = \mathbf{A}_{G_s} \dot{\mathbf{q}}_s \quad (17)$$

where:

$$\mathbf{A}_{G_s} = (\mathbf{A}_{G_s} - \mathbf{A}_{G_c} \mathbf{J}_c^{-1} \mathbf{J}_s)$$

This expression of $\mathbf{h}_{G_q}$ only depends on a subset of $\dot{\mathbf{q}}$, as a result of encoding the contact constraints into the centroidal momentum matrix $\mathbf{A}_{G_s}$.

Constrained States As shown in Eq. (15), the choice of the *constrained velocities* $\dot{\mathbf{q}}_c$ is dependent on the contact configuration of the planar model, for which there are four possible scenarios:

1. Front/Hind Stance: when the model is contacting the ground through its front or hind leg, the contact constraint is of size 2. A common choice of $\dot{\mathbf{q}}_c$ is the CoM velocity of the model.
2. Double Stance: when the two feet of the model are on the ground, two more constraints are introduced compared to the single stance case. This corresponds to the addition of two velocities to $\dot{\mathbf{q}}_c$ on top of the CoM velocity. One can choose the two joint velocities of any leg, such as $\dot{\mathbf{q}}_c = (\dot{x}_{CoM}, \dot{z}_{CoM}, \dot{q}_{H1}, \dot{q}_{H2})$.
3. Aerial: for the aerial phase where the size of contact constraint is zero, so is the size of $\dot{\mathbf{q}}_c$.

Following the choice of $\dot{\mathbf{q}}_c$, the *constrained states* $\mathbf{q}_c$ can be calculated using forward Euler:

$$\mathbf{q}_c[i+1] = \mathbf{q}_c[i] + h\dot{\mathbf{q}}_c[i] \quad (18a)$$

$$= \mathbf{q}_c[i] + h\mathbf{J}_c^{-1} \mathbf{J}_s \dot{\mathbf{q}}_s[i] \quad (18b)$$

where h is the time step between two neighbouring optimization nodes. The use of this integration scheme fits naturally into the trajectory optimization framework later presented in Section 3. Note that except for the aerial phase, the *constrained states* $\mathbf{q}_c$ of the model can also be calculated from geometric approaches.

Using the single-stance configuration in Fig. 2 as an example, the joint angles q_{F1} and q_{F2} can be parameterized with two Bézier polynomials, reducing the trajectory optimization into choosing the stance states $\mathbf{q}_{st} = (q_{Tor}, q_{F1}, q_{F2})^T$, stance velocities $\dot{\mathbf{q}}_{st}$, and the swing joints' Bézier coefficients α_{sw} . Table 2 categorizes all the system states under different dynamic phases, the transition between which will be discussed in the following section.

Swing Leg Parameterization In the presence of a swing leg, Eq. (17) can be further decomposed by parameterizing the joint trajectories using Bézier polynomials. The resulting optimization is thus reduced to choosing the swing leg Bézier coefficients α_{sw} and the rest of the states and velocities related to stance leg, denoted as $\mathbf{q}_{st}$ and $\dot{\mathbf{q}}_{st}$.

2.3. Transition map with CM states

Typically, the legged locomotion of a robot consists of a series of phases connected by discrete transitions, some of which would introduce discontinuity in system states, such as the transition between flight and stance phase marked with the robot impacting the ground.

During legged locomotion, the planar model in Fig. 2 alternates between four different dynamics based on their contact configurations as follows:

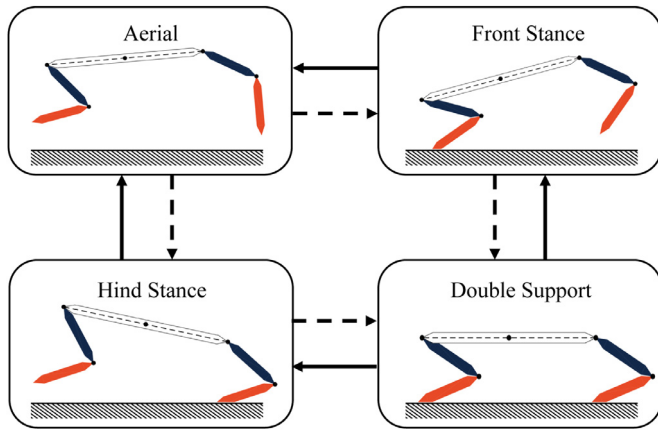


Fig. 3. The four basic dynamic phases that the planar model alternates between in its legged locomotion (the front side is to the left). State transition can only happen along the connected edges. Transitions along the dashed edges will introduce discontinuity in joint velocities caused by impact, whereas those that follow the solid edges will not.

1. Double support: the model is in contact with the ground with both its legs, its dynamics is driven by the GRFs applied on both feet,
2. Hind stance: the front leg of the model lifts off and start to swing, while the hind leg remains in contact with the ground applying forces,
3. Aerial: this happens when both the legs lose contact with the ground and swing in the air until landing,
4. Front stance: similar to the hind stance phase, except that the contact foot is the front leg while the hind leg is swinging

For example, a simple four-phase jumping would excite all the four dynamics sequentially: the model first start with a double-support posture where both its legs exert force to the ground. It transits into the hind stance configuration as its front leg lifts off while the hind leg still pushing the ground. Next the model enters the aerial phase when the hind leg loses contact, in this phase the CoM of the model follows a ballistic trajectory with only gravity acting it. The end of the aerial phase is marked with its front leg impacts the ground, entering the front stance phase. Here the impact between a swing motion and the following stance motion are not considered as an individual phase, but a mapping that connects the two phases. Ignoring the rare case that the two legs touches the ground at exactly the same time, we arrive at a state transition map shown in Fig. 3.

2.3.1. Discontinuous transition: swing to stance

When either of the legs touches the ground marking the end of its swing motion, an impact takes place. Here we assume the process to be inelastic, i.e. velocity of the contacting foot is driven to zero instantaneously [22]. Such effect introduces discontinuity to velocities of both inter-segment joints and the fictitious joint connecting Inertial Coordinate System (ICS) frame with the floating body, i.e. the CoM, whereas values of the model's generalized coordinates remain intact.

Assuming the ground reaction force during the process is an impulse with intensity δF_{ext} . The impact model can be written as:

$$D(\dot{q}^+ - \dot{q}^-) = J^T F_{ext} \quad (19)$$

where D is the inertia matrix which appears in the system's equation of motion. J is the contacting foot's Jacobian.

We further assume that the contacting foot serves as an ideal pivot after impact, i.e. it has no actuation, no slip, and no bounce, its velocity thus remains zero, same as the linear constraints in Eq. (14). Combining with Eq. (19), we obtain the impact map as:

$$\begin{bmatrix} D(q) & -J^T \\ J & 0 \end{bmatrix} \begin{bmatrix} \dot{q}^+ \\ F_{ext} \end{bmatrix} = \begin{bmatrix} \dot{q}^- \\ 0 \end{bmatrix} \quad (20)$$

where $\dot{q}^- = [\dot{x}_{CoM}^-, \dot{z}_{CoM}^-, \dot{q}_{Tor}^-, \dot{q}_{F1}^-, \dot{q}_{F2}^-, \dot{q}_{H1}^-, \dot{q}_{H2}^-]^T$.

Due to the simple centroidal momentum dynamics derived in Section 2.2.1, the CoM velocity terms in *generalized velocity* $\dot{q}^-$ above is readily available from the prescribed contact GRF, or follows ballistic motion if the dynamics phase prior to impact is an aerial phase. However, the angular velocity of the torso link $\dot{q}_{Tor}^-$, unlike the centroidal angular momentum k which can be integrated from GRF, or simply remains constant during aerial phases, is not directly tied to the GRF thus requires additional computation. To further simplify the calculation, we adopt a change of velocity coordinates from $\dot{q}$ to:

$$\dot{q}_G = (k^-, I^-, \dot{q}_{Tor}^-, \dot{q}_{F1}^-, \dot{q}_{F2}^-, \dot{q}_{H1}^-, \dot{q}_{H2}^-)^T \quad (21)$$

Note that Eq. (13) can be partitioned into:

$$\begin{bmatrix} k \\ I \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \end{bmatrix} \begin{bmatrix} I/m \\ \dot{q}_{Tor} \\ \dot{q}_b \end{bmatrix} \quad (22)$$

where $A_{1,1}, A_{1,2}, A_{1,3}, A_{2,1}, A_{2,2}, A_{2,3}$ are scalars or matrices obtained from grouping the corresponding entries of the centroidal momentum matrix A_G .

$$k = \frac{A_{1,1}}{m} I + A_{1,2} \dot{q}_{Tor} + A_{1,3} \dot{q}_b \quad (23)$$

The mapping between the *generalized velocity* $\dot{q}$ and $\dot{q}_G$ can thus be obtained from Eq. (23) as:

$$\dot{q} = \Delta_G \cdot \dot{q}_G \quad (24)$$

$$\text{where: } \Delta_G = \begin{bmatrix} \mathbf{0}_{2 \times 1} & \frac{I_{2 \times 2}}{m} & \mathbf{0}_{2 \times 4} \\ \frac{1}{A_{1,1}} & -\frac{A_{1,2}}{A_{1,1}} & -\frac{A_{1,3}}{A_{1,1}} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ \mathbf{0}_{2 \times 1} & \mathbf{0}_{2 \times 2} & \mathbf{1}_{4 \times 4} \end{bmatrix} \quad (25)$$

The resulting transition map thus becomes:

$$\begin{bmatrix} D(q) & -J^T \\ J & 0 \end{bmatrix} \begin{bmatrix} \Delta_G \cdot \dot{q}_G^+ \\ F_{ext} \end{bmatrix} = \begin{bmatrix} \Delta_G \cdot \dot{q}_G^- \\ \mathbf{0}_{2 \times 2} \end{bmatrix} \quad (26)$$

This particular form of impact map allows the direct use of the centroidal momentum, which evolves under the simple dynamics derived in Section 2.2.1 and can be directly integrated from the prescribed GRF.

2.3.2. Continuous transition: stance to swing

A continuous transition happens when a contacting foot of the model lifts off from the ground, its features include:

- The vertical position of this previously contacting foot is still zero at the moment of lifting off,
- The velocity of the previously contacting foot is non-zero, i.e. Eq. (14) no longer holds.
- The ground reaction force once applied on the system through this previously contacting foot becomes zero.

Upon removing the corresponding contact constraint, the system's dynamics changes accordingly causing a change in the generalized coordinates depending on the previous contact configuration (double support or front/hind stance). This transition is continuous both in the generalized coordinates q and in the generalized velocity $\dot{q}$. For consistency to the case of non-continuous transition, the mapping here is also written with respect to the generalized velocity defined in Eq. (21):

$$\dot{q}_G^+ = \dot{q}_G^- \quad (27)$$

With the application of $\dot{q}_G$ in both continuous and discontinuous transitions, the centroidal momentum of the model is carried across all four dynamic phases, which further simplifies the use of the model's centroidal dynamics. With all the dynamic phases defined and the transition maps between them derived, the trajectory planning for the model can be formulated as a sequence of dynamic phases connected by their corresponding transitions.

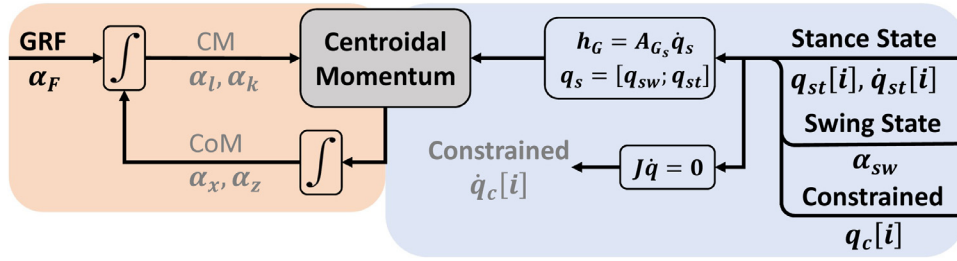


Fig. 4. This trajectory optimization framework is based on matching two centroidal momentum values, one to the left where the ground reaction forces, parameterized with Bézier coefficients α_f , are integrated once to obtain the centroidal linear momentum coefficient α_l , then again to obtain CoM trajectory required to calculate the centroidal angular momentum coefficient α_k . The path to the right in blue maps system states consisting of swing states (if any) parameterized with Bézier coefficients α_{sw} , stance states and velocities $q_{st}[i], \dot{q}_{st}[i]$,

and any constrained states $q_c[i]$ to centroidal momentum. $[i]$ indicates that its value is taken at optimization node i . To get dynamically feasible solutions, the two centroidal momentum values should agree at any node $[i]$. The bold variables in the figure are optimization variables, as shown in Eq. (28). The gray variables are dependent on the optimization variables and are used internally. The calculation for CM, CoM values are shown in details in Appendix A. For calculating $\dot{q}_c[i]$ refer to Eq. (15). Note that inequality constraints such as joint limit, friction cone are not included in this figure. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3. Trajectory optimization

The values of a robot's centroidal momentum derived in the section above should agree for any dynamically feasible trajectories. An equality constraint that unifies the two calculated values forms the basis of the proposed motion planning framework. This section presents the setup of this multi-phase trajectory optimization framework, followed by a test example to compare it with two methods

3.1. Optimization formulation

Due to the hybrid nature of legged locomotion, a robot's motion is planned in a series of phases subject to simultaneous kinematic and dynamic constraints, including the equality constraint that unifies the centroidal momentum values calculated from the GRF and from joint states. Fig. 4 list out the structure of this proposed optimization framework. The decision variables include the GRF's Bézier coefficients $\alpha_f = (\alpha_{f_x}, \alpha_{f_z})$, the Bézier coefficients of any swing leg α_{sw} , the stance states q_{st} , the stance velocities $\dot{q}_{st}$, and any constrained states q_c .

Cost function:

$$\min_{q_c[i^{(p)}], q_{st}[i^{(p)}], \dot{q}_{st}[i^{(p)}], \alpha_f^{(p)}, \alpha_{sw}^{(p)}} \sum_{p=1}^P \mathcal{L}^{(p)} \quad (28)$$

p is the phase index, P is the number of phases. The index $i^{(p)}$ indicates the values are taken at optimization node $i^{(p)}$ in phase p . $\mathcal{L}^{(p)}$ is the task-specific cost function. *Constraints for the trajectory optimization:* For any dynamic phase p , a set of optimization constraints should be applied to shape the desired trajectory (the phase index p is omitted for the rest of this section):

1. The optimization is first subject to the following dynamic constraints:

$$q_c[i+1] = q_c[i] + h\mathcal{J}(q_c[i], \alpha_{sw}, q_{st}[i], \dot{q}_{st}[i]) \quad (29a)$$

$$q_{st}[i+1] = q_{st}[i] + h\dot{q}_{st}[i] \quad (29b)$$

$$h_{G_\alpha}(i, \alpha_f) = h_{G_\alpha}(i, \alpha_{sw}, q_c[i], q_{st}[i], \dot{q}_{st}[i]) \quad (29c)$$

Equation set (29) ensure the obtained states always respect the system dynamics. Eq. (29a) is a compact form of Eq. (18) where $\mathcal{J} = \mathcal{J}_c^{-1} \mathcal{J}_s \dot{q}_s$. With Eq. (29b), the two equations describe the change in non-swing joint states in the form of forward-Euler integration. Eq. (29c) unifies the CM calculated from the GRF coefficients with the CM from the generalized coordinates and velocities, its left hand side is from Eqs. (5) to (12), and the right hand side from Eq. (17).

2. Joint range of motion, joint velocities, and on ground reaction forces such as friction cone:

$$\underline{q} \leq q \leq \bar{q} \quad (30a)$$

$$\underline{v} \leq \dot{q} \leq \bar{v} \quad (30b)$$

$$\underline{F} \leq F(i, \alpha_f) \leq \bar{F} \quad (30c)$$

where $\underline{q}$, $\underline{v}$, $\underline{F}$ denote the lower bound of the states, velocities, and GRF, while $\bar{q}$, $\bar{v}$, $\bar{F}$ are the upper bounds.

Remark 2. This formulation does not include direct constraint on joint torques, as it would involve the second-order dynamics of the system hence defying the purpose of using the simply centroidal dynamics. However, we are able to indirectly discourage the use of large joint torques by limiting the magnitude of ground reaction forces.

3. *Kinematic constraints* such as no ground penetration:

$$g(i, \alpha_{sw}^{(p)}, q_{st}[i^{(p)}]) \geq 0 \quad (31)$$

where g is the forward kinematics of major body locations such as knees.

4. *Phase transition.* To ensure the continuity between two connected phases, the following *linkage constraint* is introduced based on the *Transition Maps* described in Section 2.3:

Impact Map: Swing to Stance

$$\begin{bmatrix} D(q) & -J^T \\ J & 0 \end{bmatrix} \begin{bmatrix} \Delta_G \cdot \dot{q}_G^+ \\ F_{ext} \end{bmatrix} = \begin{bmatrix} \Delta_G \cdot \dot{q}_G^- \\ 0_{2 \times 2} \end{bmatrix} \quad (32a)$$

Continuous Map: Stance to Swing

$$\dot{q}_G^+ = \dot{q}_G^- \quad (32b)$$

which are taken from Eqs. (26) and (27).

Compared to other optimization frameworks that rely on the second-order full-body dynamics, this proposed formulation only depends on the simple CM dynamics described in Eq. (2) and the reduced first-order dynamics shown in Eq. (17). Further more, unlike other direct-collocation-based methods which discretize the equation of motion and solve the dynamic constraints at each optimization grid, thus more susceptible to numerical integration error when the number of grid points is small, this proposed method utilizes the closed-form solutions of the GRFs and swing trajectories parameterized by Bézier polynomials, as a result, the integration is done analytically hence its numerical accuracy is less sensitive to large mesh size. This permits fast optimization by using coarse time steps and low-order Bézier polynomials under mild terrain conditions, or the other way around for more refined trajectories if computation time is not a concern.

Another implicit benefit of this trajectory optimization lies in the fact that its outputs – the GRF profiles and swing leg joint trajectories, are automatically smooth and can be readily applied to the tracking controllers on real robotic systems in the form of polynomial coefficients, whereas other collocation-based methods only provide a dictionary of reference

Table 3

The constrains for the one-phase double-support trajectory optimization. The model needs to start from a given posture specified by $\mathbf{q}_b(t_0)$ and rise to a given final CoM height.

$t_0 = 0$ s, $t_f \in [0.5$ s, $+\infty)$	
$\mathbf{q}_b(t_0) = [0^\circ, -20^\circ, -140^\circ, -20^\circ, -140^\circ]$	
$\dot{\mathbf{q}}_b(t_0) = 0$, $\dot{\mathbf{q}}_b(t_f) = 0$	
$\mathbf{q}_b(t) \in \mathcal{Q}$, $ \dot{\mathbf{q}}_b(t) \leq 30$ rad/s	
$F_{ix}(t) < 0.7F_{iz}(t)$	$i \in \{F, H\}$
$F_{ix}(t) \in [-30$ N, 30 N]	
$F_{ix}(t) \in [0$ N, 30 N]	
$x_{CoM}(t_f) = x_{CoM}(t_0)$	
$z_{CoM}(t_f) = 0.18$ m	

points which are not necessarily smooth. Though the stance coordinates $\mathbf{q}_{st}$ and velocities $\dot{\mathbf{q}}_{st}$ are still picked independently at each nodes to meet various dynamic and kinematic constraints, as Eq. (29) implies, their values are not directly used in controller designs since the stance leg is under force control which does not rely on the joint trajectories but rather on the closed-form solution of the GRF.

3.2. Comparison with other formulations

The discontinuity in joint velocities inherent to the legged locomotion limits the use of numerical integration up to first order [23,24]. As a result, direct collocation methods that do not utilize closed-form GRFs are prone to loss of accuracy when force to run with coarse time steps, whereas this proposed method would not receive less impact. A simple trajectory optimization example is tested below to verify the performance of our proposed method.

Test Example The planar model in Fig. 2 is commanded to rise to a given CoM height starting from a double-support contact posture. The motion is subject to constraints on GRF, joint angle, and joint velocity, detailed optimization constraints over its states and controls are specified in Table 3.

The cost function is chosen as 0 to find a feasible trajectory:

$$\min_{\mathbf{q}[i], \dot{\mathbf{q}}[i], \mathbf{F}[i]} \mathcal{L} = 0 \quad (33)$$

here the $\mathbf{F}[i]$ is the value of the GRF vector taken at optimization node $[i]$.

Reference Methods Two optimization formulations based on direct collocation are chosen as references. Similar to the proposed formulation, they also include GRFs as decision variables, but are chosen independently at each optimization grids without seeking their closed-form

solution. The first method uses the same CM dynamics as Section 2.2.2, subject to the following dynamic constraints other than the ones already include in Table 3:

$$\mathbf{q}[i+1] = \mathbf{q}[i] + h\dot{\mathbf{q}}[i] \quad (34a)$$

$$\mathbf{h}_{G_a}(i, \mathbf{F}[i]) = \mathbf{h}_{G_a}(i, \mathbf{q}[i], \dot{\mathbf{q}}[i]) \quad (34b)$$

The second formulation uses the model's full-body dynamics, resulting in a different set of dynamic constraints:

$$\mathbf{q}[i+1] = \mathbf{q}[i] + h\dot{\mathbf{q}}[i] \quad (35a)$$

$$\dot{\mathbf{q}}[i+1] = \mathbf{f}(i, \mathbf{q}[i], \dot{\mathbf{q}}[i], \mathbf{F}[i]) \quad (35b)$$

where $\mathbf{f}$ is derived from the standard joint-space EoM (equation of motion) of the model. All optimizations would start from zero initial conditions, are formulated using GPOPS package [25], and solved using SNOPT [26] on a laptop with i7-8550U processor.

Result Comparison With the cost function defined in Eq. (33), all three optimization formulations were able to find a simple double-stance standing-up motions, though the computational time varied: the CM-dynamics-based reference method took 0.7203 s to converge, and the full-body-dynamic-based method spent 0.7904 s on finding a feasible solution. In comparison, the proposed method was able to converge in 0.5005 s.

To assess the quality of the optimization outputs, all the output trajectories are compared with the solutions from integrating the full-body dynamics with high-accuracy Runge-Kutta algorithm, starting with the controls and initial conditions taken from the corresponding optimization outputs. The comparison results are shown in Fig. 5.

When looking at Fig. 5-A1 and Fig. 5-B1, it is evident that the GRF curves provided by the two reference optimization formulations are highly non-smooth. Whereas our formulation automatically generates smooth GRF profiles shown in Fig. 5-C1. Though the two reference methods were able to lift the CoM close to the desired height (0.18 m), as demonstrated in Fig. 5-A2 and Fig. 5-B2, the accuracy of the resulting CoM trajectories were noticeably compromised due to the numerical integration of coarse GRF.

To rule out the possibility that the inaccuracy of the two reference methods is purely caused by the non-ideal GRFs, not inherent to the two formulations themselves, the following cost function is applied to all three methods to regulate the GRFs.

$$\min_{\mathbf{q}[i], \dot{\mathbf{q}}[i], \mathbf{F}[i]} \mathcal{L} = \sum_i \mathbf{w} \mathbf{F}[i]^2, \quad (36)$$

the weights $\mathbf{w}$ are chosen such that the contribution from the L2 norms of the horizontal and vertical forces are at the same order of magnitude,

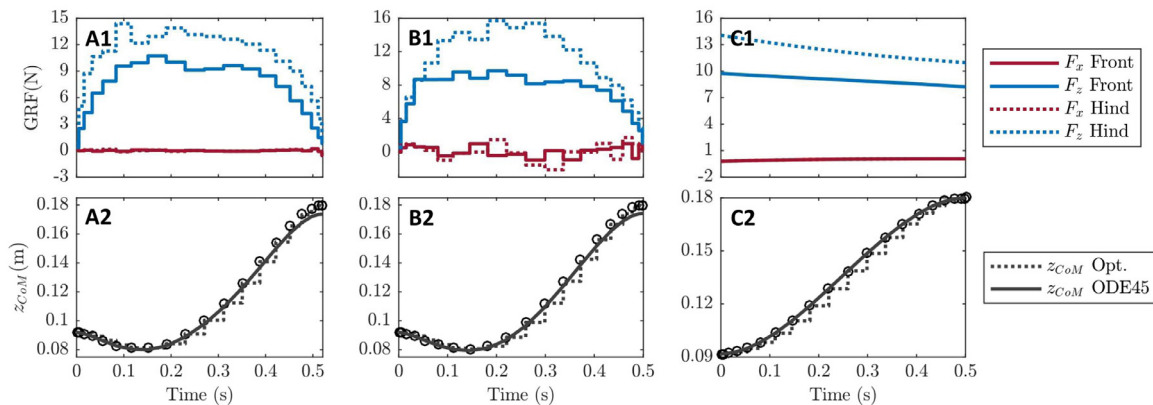


Fig. 5. The outputs from the three trajectory optimization formulations for the same test example. From left to right: CM-dynamic-based, non-closed-form GRFs (column A), full-body-dynamic-based, non-closed-form GRFs (column B), and CM-dynamic-based, closed-form GRFs (column C, the proposed method). The first two methods generated non-smooth force profiles, as shown in A1 and B1, while the proposed method naturally generated smooth GRF in C1. Using these GRF curves, high-order integration based on the ODE45 solver is performed in A2, B2, and C2 to calculate the corresponding CoM height trajectories (solid line) and compare with the CoM height trajectories from optimizations (dotted lines). The proposed optimization results in noticeably more accurate CoM trajectories.

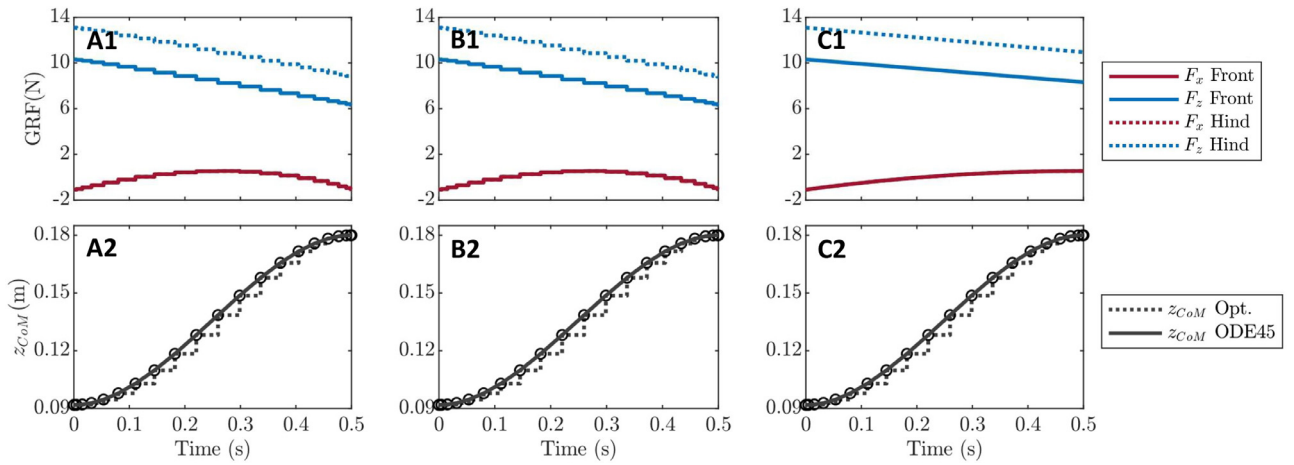


Fig. 6. The regulated GRF outputs and the CoM height trajectories from the CM-dynamic-based optimization with non-closed-form GRF (column A), the full-body-dynamic-based optimization (column B), and the proposed method (column C). It is evident that the CoM height trajectory predicted by the optimization is closer to that from ODE45 compared to Fig. 5 for column A and B.

else the cost term would be dominated by the vertical forces based on the results in Fig. 5. In our test case, the weights for the horizontal and vertical forces are chosen as 1 and 0.6 respectively. This cost function steers the optimizer away from choosing large GRF values, effectively smoothing out the GRF profiles.

Note that the added terms in optimization would likely lead to an increase in computation time, in this case the CM-dynamic-based method ended in 3.3498 s, and the full-body-dynamic-based method converged after 21.5129 s. Our proposed method also saw a rise in computation time to 5.7028 s.

It is worth noting that the cost function was added to the proposed method only to make a fair comparison, the method itself does not require such an addition and was able to generate smooth GRF profiles on its own, as suggested by Fig. 5.

The updated outputs are shown in Fig. 6. With the GRF curves smoothed out seen in Fig. 6-A1 and Fig. 6-B1, the accuracy of the CoM trajectories from the two reference methods sees a noticeable improvement shown in Fig. 6-A2 and Fig. 6-B2. Unsurprisingly, the outputs from our proposed method does not experience a significant change, as its previous GRF outputs were already smooth. However, taking a closer look at the absolute errors of the three outputs in Fig. 7, our method is able to consistently provide more accurate position and torso angle (and other joint angles) before and after the addition of the extra cost term.

The energy properties of the three formulations are also compared to see if their outputs are energy efficient in achieving the same motion, by examining the cost of transport (CoT) [27] of each solutions calculated as:

$$C(t) = \int_{t_0}^t \frac{\sum_{i,j} \max(\tau_{ij}(t)\dot{q}_{ij}(t), 0)}{m_0 g \Delta z_{CoM}} dt, t \in [t_0, t_f] \quad (37)$$

where the index $i \in \{F, H\}$ denotes the location of the joint: on front or hind leg, $j \in \{1, 2\}$ stands for hip ($j = 1$) or knee ($j = 2$). $\Delta z_{CoM} = z_{CoM}(t_f) - z_{CoM}(t_0)$ is the total change of CoM height. In this particular setup, since the model starts and ends with zero joint velocities, its CoM ends at the same horizontal position, the change of its total mechanical energy equals the change in potential energy $m_0 g \Delta z_{CoM}$.

The CoT of the three methods are shown in Table 4. The motions from the two reference methods are less energy efficient before smoothing out the GRF curves, as seen with larger CoT values. It is worth noting that smoothing out the GRF caused the CoT of the two reference methods to match that of the proposed method. This is possibly due to the fact that the smooth GRF curves from the proposed method is manually discretized when solving for the joint mechanical work, resulting in less-smooth profiles.

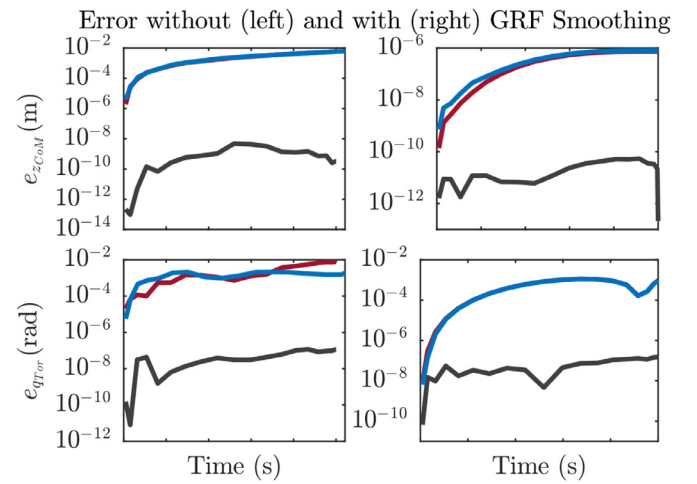


Fig. 7. The absolute error of the proposed method (black) compare with two other reference methods: one based on CM dynamics (red), the other uses the full-body dynamics (blue). The proposed method has the least numerical errors in both CoM height (top) and torso angle (bottom).

Table 4

The CoT (Cost of Transportation) of the motions created by the proposed method (CM dynamics, closed-form GRFs) and the two reference methods: one based on CM dynamics, the other on full-body dynamics, both with non-closed-form GRFs.

Formulation	CoT	
	Coarse GRF	Smooth GRF
Proposed Method	1.101	1.087
Ref. 1: CM Dynamics	1.287	1.087
Ref. 2: Full-body Dynamics	1.363	1.087

4. Experimental validation

This section introduces the planar testbed used in validating the proposed trajectory optimization framework, together with the experiment result of a 3-phase forward jumping motion. The optimization was based

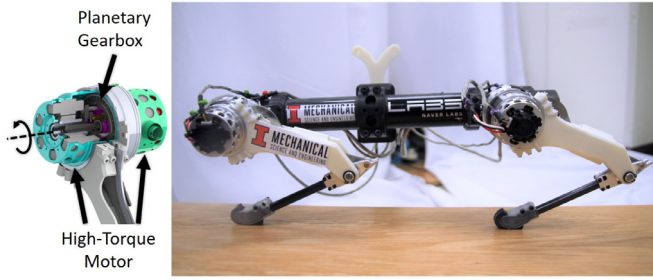


Fig. 8. The planar robot testbed used for experimental validation of the proposed algorithm. Its two quasi-direct-drive legs are capable of high-bandwidth, high-fidelity force control.

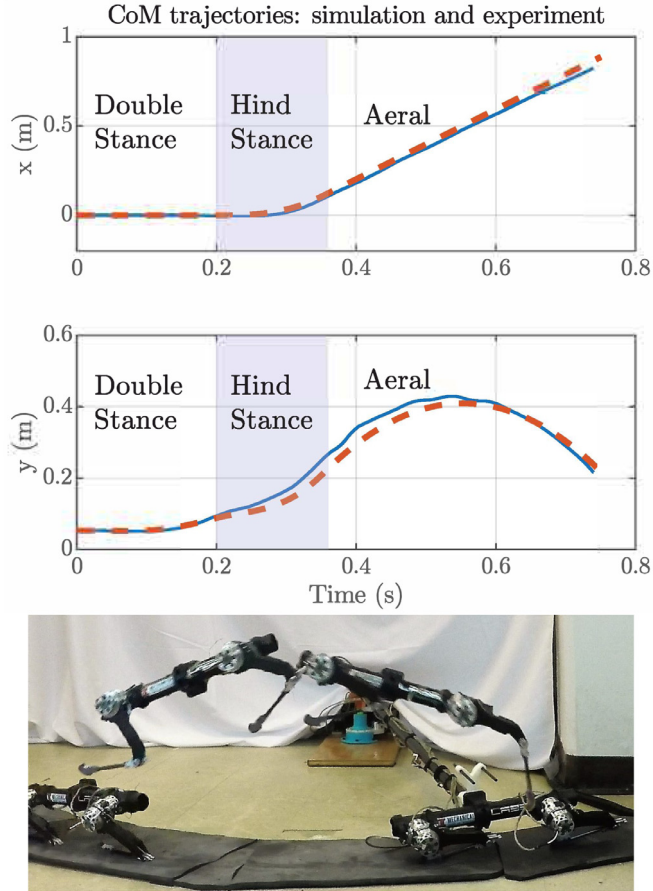


Fig. 9. The planar robot testbed executing a 3-phase leaping motion following the output from the proposed trajectory optimization framework. The *double stance* phase and *flight* phase are separated by the *hind stance* phase (shaded). Top: the CoM trajectory from the optimization output (dashed orange) and the experiment measurement taken from the boom encoders (blue). Bottom: the testbed jumped from right to left. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

on the planar floating-base model described in Section 2.1 and formulated with the same structure proposed in Section 3.

4.1. Test hardware

A two-legged robot testbed shown in Fig. 8 is used to validate the trajectories generated by the proposed method. The testbed is driven by two customized BLDC motor with high torque density and a low gear ratio of 23:1, allowing high-fidelity control of GRF [28]. The two legs both consist of two 15 cm links, one 3D-printed (white) and the other cut from carbon fiber tube (black). The legs are attached to a 30cm

carbon fiber tube acting as a torso. The robot testbed is supported by a boom system to constrain its motion in a plane. Most of the robot's electronic components are separated from its body, including four Elmo G-TWI motor drivers and a Intel Core i7 Single-Board-Computer (SBC) which runs a control loop at 4kHz in Simulink Real-Time, leaving only four magnetic encoders mounted to each motor's rotation axis providing joint angle readings. The entire experimental setup weighs 1.9 kg. The experiment video can be found at <https://youtu.be/3rcmn4K3ZeY>.

4.2. Experimental results

In this experiment, the robot is commanded to perform a forward leap following the trajectory produced using the same setup as Section 3.1 and Table 3. The cost function is set to maximize the travel distance.

The leaping motion consisted of the following three phases: the robot started from a *double support* phase with both of its legs under force control. It entered the *hind stance* phase after 0.2 s, with the feedforward force on its front leg set to zero while the hind leg keep pushing against the ground for another 0.16s. The last phase – *flight phase* started after the hind leg stops applying force, ended with the robot landing on its front leg. Since there was no direct way of measuring the testbed's CoM position, it was approximated as the center of the torso link, as shown in Fig. 9.

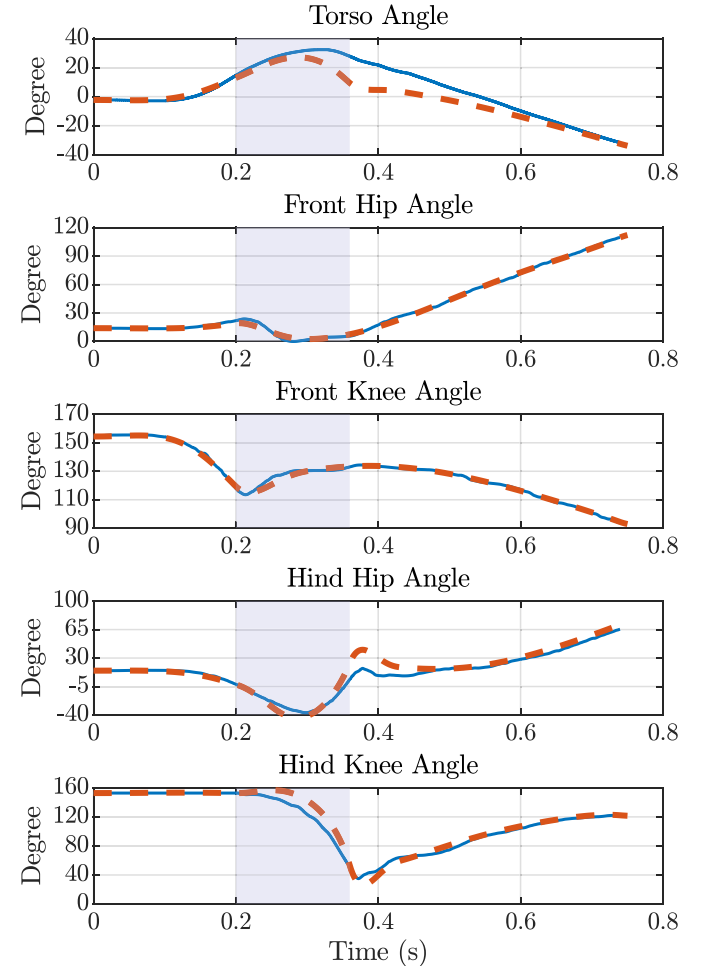


Fig. 10. Comparing the torso and joint trajectories generated by the proposed algorithm (dashed orange) with the corresponding angle readings recorded when performing the motion on the robot testbed (blue). All figures are separated into three stages by two black dashed lines, which are 1) double support, 2) hind stance, and 3) aerial phase from left to right. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5
The controller gains used in each phase.

Phase	Kp(N/m)		Kd(N/ms ⁻¹)	
	Front	Hind	Front	Hind
Double Support	100	100	0.5	0.5
Hind Stance	500	100	4	0.5
Aerial	500	500	4	4

The swing leg was under impedance control to track the desired trajectory given by as a set of Bézier polynomials. Another impedance controller with small stiffness and damping was added on top of the feedforward force to help regulate the stance leg position, with its max control effort capped at 15% of the GRF value to limit its effect on the robot's motion. The gains for each phase are listed in Table 5. Joint trajectories are shown in Fig. 10, which closely matched the output trajectories from the optimization.

5. Conclusion and future work

In this paper we present a trajectory optimization framework for planning legged locomotion based on a robot's centroidal momentum, which enjoys simple dynamics dominated by the GRFs and the gravity. By parameterizing the GRFs as Bézier polynomials, the centroidal momentum of the robot can be solved analytically instead of through numerical integration. This avoids interpolating or curve-fitting the output trajectories between collocation points, which may result in sub-optimal trajectories or violation of constraints. The same parameterization is also applied to any swing leg joints. The accuracy of the CoM and swing joints trajectories provided by this framework hence is independent of mesh size. As an added benefit, its outputs are automatically smooth, and can be readily applied to a tracking controller in real robotic systems as sets of polynomial coefficients.

The optimization framework was able to produce a jumping motion which was validated by a planar robot platform with reasonable accuracy. Though the proposed method has been derived and validated in planar cases, thanks to the simplicity of the centroidal momentum dynamics of a robot, the extension from planar to 3D would not necessarily induce substantial revision to the existing formulation, where Eq. (2b) remains the same while Eq. (2a) is now a 3×1 vector. Such a change would still allow the use of parameterized ground reaction forces and the close-form solution of centroidal momentum, with the rest of the derivations remain largely unchanged. The authors believe that the results presented in this paper could encourage further implementation of this method on more complex locomotion such as multi-step walking and running.

Declaration of Competing Interest

The authors declare that they do not have any financial or nonfinancial conflict of interests

CRediT authorship contribution statement

Chuanzheng Li: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing - original draft. **Yanran Ding:** Investigation, Resources, Validation, Software. **Hae-Won Park:** Conceptualization, Methodology, Supervision, Project administration, Funding acquisition, Writing - review & editing.

Acknowledgment

This project is supported by NAVER LABS Corp. under grant 087387, Air Force Office of Scientific Research under grant FA2386-17-1-4665, and National Science Foundation under grant 1752262.

Appendix A. Bézier calculation for centroidal momentum and CoM trajectory

The centroidal momentum of the planar model in Fig. 2 evolves under a net external wrench induced by gravity and the contact forces exerted by its environment:

$$\dot{\mathbf{l}} = m\dot{\mathbf{r}} = \sum_i \mathbf{F}_i - m\mathbf{g} \quad (\text{A.1a})$$

$$\dot{\mathbf{k}} = \sum_i (\mathbf{c}_i - \mathbf{r}) \times \mathbf{F}_i \quad (\text{A.1b})$$

Note that the centroidal linear momentum $\mathbf{l}$ in Eq. (A.1a) only linearly depend on the ground reaction forces $\mathbf{F}$, we can first solve $\mathbf{l}$ by integrating $\mathbf{F}$ one time, then again for the CoM trajectory, then Eq. (A.1b) can be integrated to get the centroidal angular momentum $\mathbf{k}$.

Given the horizontal and vertical GRF profiles parameterized with M th order Bézier curves with coefficients

$$\alpha_{F_x} = [\alpha_{F_x,0}, \alpha_{F_x,1}, \dots, \alpha_{F_x,M}] \in \mathbb{R}^{M+1}$$

$$\alpha_{F_z} = [\alpha_{F_z,0}, \alpha_{F_z,1}, \dots, \alpha_{F_z,M}] \in \mathbb{R}^{M+1}$$

respectively, the corresponding centroidal linear momentum can be calculated from Bézier polynomials differentiation:

$$\frac{d}{ds} B_{i,M}(s) = \frac{M}{T} (B_{i-1,M-1}(s) - B_{i,M-1}(s)) \quad (\text{A.2})$$

where $B_{i,M}(s)$ is the i th coefficient of a M th order Bézier curve, $i = 0, 1, \dots, M$, $s \in [0, T]$.

Given the initial CoM velocities $\dot{x}_0, \dot{z}_0$, from the GRF coefficients $\alpha_{F_x}, \alpha_{F_z}$, the calculation for the centroidal momentum coefficients $\alpha_{l_x}, \alpha_{l_z} \in \mathbb{R}^{M+2}$ based on Eq. (A.2) can be written as:

$$\begin{bmatrix} -\frac{T}{M} & \frac{T}{M} & 0 & \dots & 0 & 0 \\ 0 & -\frac{T}{M} & \frac{T}{M} & \dots & 0 & 0 \\ & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -\frac{T}{M} & \frac{T}{M} \\ 1 & 0 & \dots & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha_{l_x,0} \\ \alpha_{l_x,1} \\ \vdots \\ \alpha_{l_x,M} \\ \alpha_{l_x,M+1} \end{bmatrix} = \begin{bmatrix} \alpha_{F_x,0} \\ \alpha_{F_x,1} \\ \vdots \\ \alpha_{F_x,M} \\ m\dot{x}_0 \end{bmatrix}$$

$$\begin{bmatrix} -\frac{T}{M} & \frac{T}{M} & 0 & \dots & 0 & 0 \\ 0 & -\frac{T}{M} & \frac{T}{M} & \dots & 0 & 0 \\ & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -\frac{T}{M} & \frac{T}{M} \\ 1 & 0 & \dots & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha_{l_z,0} \\ \alpha_{l_z,1} \\ \vdots \\ \alpha_{l_z,M} \\ \alpha_{l_z,M+1} \end{bmatrix} = \begin{bmatrix} \alpha_{F_z,0} - mg \\ \alpha_{F_z,1} - mg \\ \vdots \\ \alpha_{F_z,M} - mg \\ m\dot{z}_0 \end{bmatrix}$$

where m is the robot's mass, g is the gravitational acceleration. The two equations can be condensed into the following form:

$$\Phi_{M+2,T} \alpha_{l_x} = \begin{bmatrix} \alpha_{F_x} \\ m\dot{x}_0 \end{bmatrix}, \quad \Phi_{M+2,T} \alpha_{l_z} = \begin{bmatrix} \alpha_{F_z} - mg \\ m\dot{z}_0 \end{bmatrix} \quad (\text{A.3})$$

$\Phi_{M+2,T} \in \mathbb{R}^{(M+2) \times (M+2)}$ is the corresponding coefficient matrix, its elements defined as:

$$\Phi_{M+2,T}(i, j) := \begin{cases} -M/T, & i = j = 1, 2, \dots, M+1 \\ M/T, & i = j-1 = 1, 2, \dots, M+1 \\ 1, & i = M+2, j = 1 \\ 0, & \text{otherwise} \end{cases}$$

$\Phi_{M+2,T}$ is by its structure full rank hence always invertible.

Similarly, the Bézier coefficients $\alpha_x, \alpha_z \in \mathbb{R}^{(M+3)}$ of the CoM trajectory can be retrieved given the initial CoM position x_0, z_0 as follows:

$$\Phi_{M+3,T} \alpha_x = \begin{bmatrix} \alpha_{l_x} \\ m \\ x_0 \end{bmatrix}, \quad \Phi_{M+3,T} \alpha_z = \begin{bmatrix} \alpha_{l_z} \\ m \\ z_0 \end{bmatrix}$$

The calculation for centroidal angular momentum $\mathbf{k}$ starts from calculating $\dot{\mathbf{k}}$ from Eq. (A.1b), which can be written into Bézier polynomial multiplication as:

$$\dot{\mathbf{k}} = F^z(x_1 - x_{CoM}) - F^x(z_1 - z_{CoM}).$$

Note that the multiplication of two Bézier polynomials yields another Bézier polynomial, the order of which is the sum of the two multiplicands. Denote the Bézier coefficient of k as $\alpha_k \in \mathbb{R}^{(2M+3)}$. Due to the space limit, we first calculate the two terms $F^z(x_1 - x_{CoM})$ and $F^x(z_1 - z_{CoM})$ separately. The two terms also take the form of Bézier polynomials. Their coefficients, denoted as $\alpha_1, \alpha_2 \in \mathbb{R}^{2M+3}$, can be obtained with the Bézier polynomial multiplication rule:

$$\alpha_1(i) = \sum_{j=\max(0, i-M-2)}^{\min(M, i)} C_{i,j} \alpha_{F_z}(i)(x_1 - \alpha_x(i-j))$$

$$\alpha_2(i) = \sum_{j=\max(0, i-M-2)}^{\min(M, i)} C_{i,j} \alpha_{F_x}(i)(z_1 - \alpha_z(i-j))$$

the coefficients $C_{i,j}$ are given by:

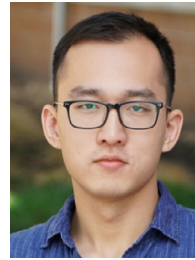
$$C_{i,j} = \frac{\binom{M}{j} \binom{M+2}{i-j}}{\binom{2M+2}{i}}, \text{ where } \binom{M}{j} = \frac{M!}{j!(M-j)!}$$

Then from $\alpha_k = \alpha_1 + \alpha_2$ and given the initial value k_0 , the centroidal angular momentum k can be calculated following the same principle as Eq. (A.3):

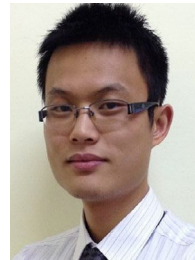
$$\Phi_{2M+4,T} \alpha_k = \begin{bmatrix} \alpha_k \\ k_0 \end{bmatrix}$$

References

- [1] Raibert M, Blankespoor K, Nelson G, Playter R. BigDog, the rough-terrain quadruped robot. IFAC Proc Vol 2008;41(2):10822–5. doi:10.3182/20080706-5-KR-1001.01833.
- [2] Hutter M, Gehring C, Jud D, Lauber A, Bellicoso CD, Tsounis V, et al. AnyMal - a highly mobile and dynamic quadrupedal robot. In: 2016 IEEE/RSJ International conference on intelligent robots and systems; 2016. p. 38–44. doi:10.1109/IROS.2016.7758092. ISBN 2153-0866
- [3] Park H-W, Wensing PM, Kim S. High-speed bounding with the mit cheetah 2: control design and experiments. Int J Rob Res 2017;36(2):167–92. doi:10.1177/0278364917694244.
- [4] Semini C, Barasuol V, Goldsmith J, Frigerio M, Focchi M, Gao Y, et al. Design of the hydraulically actuated, torque-controlled quadruped robot HyQ2Max. IEEE/ASME Trans Mechatron 2017;22(2):635–46. doi:10.1109/TMECH.2016.2616284.
- [5] Mombaur K. Using optimization to create self-stable human-like running. Robotica 2009;27(3):321–30. doi:10.1017/S0263574708004724.
- [6] Tassa Y, Erez T, Todorov E. Synthesis and stabilization of complex behaviors through online trajectory optimization. In: 2012 IEEE/RSJ International conference on intelligent robots and systems; 2012. p. 4906–13. doi:10.1109/IROS.2012.6386025. ISBN 2153-0866
- [7] Posa M, Cantu C, Tedrake R. A direct method for trajectory optimization of rigid bodies through contact. Int J Rob Res 2014;33(1):69–81. doi:10.1177/0278364913506757.
- [8] Neunert M, Farshidian F, Winkler AW, Buchli J. Trajectory optimization through contacts and automatic gait discovery for quadrupeds. IEEE Rob Autom Lett 2017;2(3):1502–9. doi:10.1109/LRA.2017.2665685.
- [9] Ding Y, Li C, Park H-W. Single leg dynamic motion planning with mixed-integer convex optimization. In: 2018 IEEE/RSJ International conference on intelligent robots and systems; 2018. p. 1–6. doi:10.1109/IROS.2018.8594161. ISBN 2153-0866
- [10] Kajita S, Kanehiro F, Kaneko K, Yokoi K, Hirukawa H. The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In: 2001 IEEE/RSJ International conference on intelligent robots and systems, vol. 1; 2001. p. 239–46. doi:10.1109/IROS.2001.973365.
- [11] Lee SH, Goswami A. Reaction mass pendulum (RMP): an explicit model for centroidal angular momentum of humanoid robots. In: 2007 IEEE/RSJ International conference on robotics and automation; 2007. p. 4667–72. doi:10.1109/ROBOT.2007.364198. ISBN 1050-4729
- [12] Wensing PM, Orin DE. Development of high-span running long jumps for humanoids. In: 2014 IEEE International conference on robotics and automation; 2014. p. 222–7. doi:10.1109/ICRA.2014.6906613. ISBN 1050-4729
- [13] Orin DE, Goswami A. Centroidal momentum matrix of a humanoid robot: Structure and properties. In: 2008 IEEE/RSJ International conference on intelligent robots and systems; 2008. p. 653–9. doi:10.1109/IROS.2008.4650772.
- [14] Herzog A, Rotella N, Schaaf S, Righetti L. Trajectory generation for multi-contact momentum control. In: 2015 IEEE-RAS 15th International conference on humanoid robots; 2015. p. 874–80. doi:10.1109/HUMANOIDS.2015.7363464.
- [15] Fernbach P, Tonneau S, Tax M. CROC: Convex resolution of centroidal dynamics trajectories to provide a feasibility criterion for the multi contact planning problem. In: 2018 IEEE/RSJ International conference on intelligent robots and systems; 2018. p. 1–9. doi:10.1109/IROS.2018.8593888. ISBN 2153-0866
- [16] Winkler AW, Bellicoso CD, Hutter M, Buchli J. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. IEEE Rob Autom Lett 2018;3(3):1560–7. doi:10.1109/LRA.2018.2798285.
- [17] Dai H, Valenzuela A, Tedrake R. Whole-body motion planning with centroidal dynamics and full kinematics. In: 2014 IEEE-RAS International conference on humanoid robots; 2014. p. 295–302. doi:10.1109/HUMANOIDS.2014.7041375.
- [18] Ostrowski JP. Computing reduced equations for robotic systems with constraints and symmetries. IEEE Trans Rob Autom 1999;15(1):111–23. doi:10.1109/70.744607.
- [19] Orin DE, Goswami A, Lee S-H. Centroidal dynamics of a humanoid robot. Auton Robots 2013;35(2):161–76. doi:10.1007/s10514-013-9341-4.
- [20] Farouki RT, Rajan VT. Algorithms for polynomials in Bernstein form. Comput Aided Geom Des 1988;5(1):1–26. doi:10.1016/0167-8396(88)90016-7.
- [21] Park H-W, Wensing PM, Kim S. Online planning for autonomous running jumps over obstacles in high-speed quadrupeds. Robotics: science and systems XI; 2015. doi:10.15607/RSS.2015.XI.047.
- [22] Acoustin Y, Chevallereau C, Formal'sky A. Numerical and experimental study of the virtual quadrupedal walking robot-semiquad. Multibody Syst Dyn 2006;16(1):1–20. doi:10.1007/s11044-006-9015-5.
- [23] Xi W, Remy CD. Optimal gaits and motions for legged robots. In: 2014 IEEE/RSJ International conference on intelligent robots and systems; 2014. p. 3259–65. doi:10.1109/IROS.2014.6943015. ISBN 2153-0858
- [24] Anitescu M. A fixed time-step approach for multibody dynamics with contact and friction. In: Proceedings 2003 IEEE/RSJ international conference on intelligent robots and systems (IROS 2003) (Cat. No.03CH37453), 4; 2003. p. 3725–3731 vol.3. doi:10.1109/IROS.2003.1249734.
- [25] Rao AV, Benson DA, Darby C, Patterson MA, Francolin C, Sanders I, et al. Algorithm 902: Gpops, a matlab software for solving multiple-phase optimal control problems using the gauss pseudospectral method. ACM Trans Math Softw 2010;37(2):1–39. doi:10.1145/1731022.1731032.
- [26] Gill PE, Murray W, Saunders MA. SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM Rev 2005;47(1):99–131. doi:10.1137/s0036144504446096.
- [27] Collins S, Ruina A, Tedrake R, Wisse M. Efficient bipedal robots based on passive-dynamic walkers. Science 2005;307(5712):1082–5. doi:10.1126/science.1107799.
- [28] Ding Y, Park H-W. Design and experimental implementation of a quasi-direct-drive leg for optimized jumping. In: 2017 IEEE/RSJ International conference on intelligent robots and systems; 2017. p. 300–5. doi:10.1109/IROS.2017.8202172. ISBN 2153-0866



Chuanyang Li received his B.S. degree in Mechatronics from Zhejiang University, Hangzhou, China in 2014, and the M.S. degree from the Mechanical Science and Engineering Department, University of Illinois at Urbana-Champaign, Champaign, IL, USA in 2017. He is currently in the Ph.D. program at University of Illinois at Urbana-Champaign supervised by Dr. Hae-Won Park, working primarily on the design of mechatronic systems and the real-time control of legged robots.



Yanran Ding received B.S. degree from UM-SJTU joint institute, Shanghai Jiao Tong University, Shanghai, China in 2015. He has received the M.Sc. degree from the Mechanical Science and Engineering Department, University of Illinois at Urbana-Champaign, Champaign, IL, USA in 2017, and he is currently working towards his Ph.D. degree at the Dynamic Robotics Laboratory. His research interests include the design and control agile robotics systems. His research include mechanism design, actuation systems and concentrates on using applied optimization and control theory to enable legged robots to achieve dynamic motions. He is one of the best student paper finalists in IROS 2017.



Hae-Won Park is an Assistant Professor of Mechanical Engineering at the Korean Advanced Institute of Science and Technology (KAIST). He received B.S. and M.S. degrees from Yonsei University, Seoul, Korea, in 2005 and 2007, respectively, and the Ph.D. degree from the University of Michigan, in 2012, all in mechanical engineering. His research interests lie at the intersection of control, dynamics, and mechanical design of robotic systems, with special emphasis on legged locomotion robots. He is the recipient of the 2018 National Science Foundation (NSF) CAREER Award, NSF's most prestigious awards in support of early-career faculty.