Real-Time Planning and Nonlinear Control for Quadrupedal Locomotion with Articulated Tails

Randall T. Fawcett
M.S. Student
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061
Email: randallf@vt.edu

Abhishek Pandala
Ph.D. Student
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061
Email: agp19@vt.edu

Jeeseop Kim
Ph.D. Student
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061
Email: jeeseop@vt.edu

Kaveh Akbari Hamed[*]
Assistant Professor
Mechanical Engineering
Virginia Tech
Blacksburg, VA 24061
Email: kavehakbarihamed@vt.edu

The primary goal of this paper is to develop a formal foundation to design nonlinear feedback control algorithms that intrinsically couple legged robots with bio-inspired tails for robust locomotion in the presence of external disturbances. We present a hierarchical control scheme in which a high-level and real-time path planner, based on an event-based model predictive control (MPC), computes the optimal motion of the center of mass (COM) and tail trajectories. The MPC framework is developed for an innovative reduced-order linear inverted pendulum (LIP) model that is augmented with the tail dynamics. At the lower level of the control scheme, a nonlinear controller is implemented through the use of quadratic programming (QP) and virtual constraints to force the full-order dynamical model to track the prescribed optimal trajectories of the COM and tail while maintaining feasible ground reaction forces at the leg ends. The potential of the analytical results is numerically verified on a full-order simulation model of a quadrupedal robot augmented with a tail with a total of 20 degrees of freedom. The numerical studies demonstrate that the proposed control scheme coupled with the tail dynamics can significantly reduce the effect of external disturbances during quadrupedal locomotion.
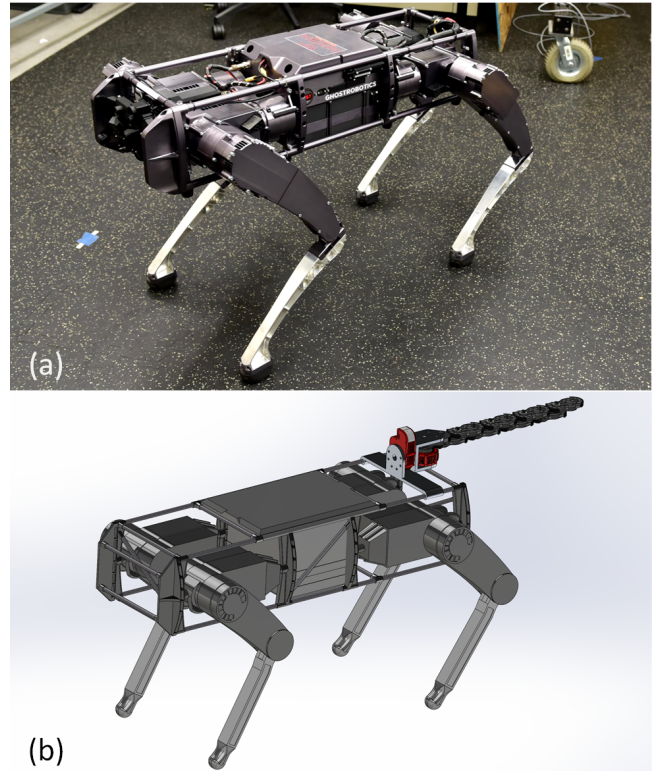
Fig. 1. (a) Vision 60 robot with 18 DOFs whose full-order dynamical model will be used for the numerical simulations. (b) CAD representation of Vision 60 augmented with a 2-DOF tail mechanism.

1 Introduction

This paper presents a systematic approach to design a hierarchical control algorithm, based on reduced-order models, model predictive control (MPC), quadratic programming (QP), and virtual constraints, to intrinsically couple bio-inspired articulated tails with legged robots for robust lo-

comotion. In the proposed approach, the high-level MPC computes an optimal center of mass (COM) trajectory for the body and an optimal tail trajectory in real time. More

---

[*]Corresponding Author

specifically, we develop an MPC formulation for an innovative and augmented linear inverted pendulum (LIP) model which has been extended to consider the forces introduced by the tail dynamics. The high-level MPC is implemented in an event-based manner in the sense that it is solved at the beginning of each continuous-time domain subject to the predefined foot placements, the feasibility of the net ground reaction force (GRF), and the tail joint and actuation limits. In order to close the gap between the reduced-order model used by the high-level MPC and the full-order model of the robot augmented with a tail, a low-level nonlinear controller, based on convex QP and virtual constraints, is implemented to track the prescribed COM and tail trajectories while ensuring feasible GRFs at the contacting leg ends. The potential of the proposed control approach is numerically demonstrated through extensive full-order simulations of the Vision 60 quadruped augmented with a serpentine tail with a total of 20 degrees of freedom (DOFs) (see Fig. 1). It is shown that extending existing MPC frameworks for locomotion to consider additional dynamics induced by a tail can produce greater robustness to external disturbances during locomotion than a legged system without the addition of a tail like structure.

1.1 Background, Motivation, and Challenges

Related Work on Legged Locomotion: The most widely used control method for legged locomotion is based on a heuristic notion of gait stability, known as the zero moment point (ZMP) criterion [1, 2]. This control methodology produces quasi-static gaits and assumes full-actuation. Alternatively, legged locomotion can be described by hybrid systems [3, 4] with continuous-time domains (i.e., phases) to represent the Lagrangian dynamics and discrete-time transitions between continuous-time domains to represent the abrupt changes in the velocity components of the state vector during the infinitesimal period of the rigid impacts between the swing leg ends and the environment [5–29]. State-of-the-art nonlinear control algorithms that address the hybrid nature of legged locomotion to stabilize gait patterns include controlled symmetries [8], hybrid reduction [30, 31], transverse linearization [10, 32], and hybrid zero dynamics (HZD) [7, 18, 33, 34]. Of these methods, only HZD and transverse linearization are capable of addressing the general case of underactuation inherent to dynamic locomotion. In the case of HZD, a set of kinematic constraints, referred to as virtual constraints [5, 34], are enforced via input-output (I-O) feedback linearization [35] in order to impose coordinated motion among the limbs. Virtual constraint controllers have been validated both numerically and experimentally for the stable locomotion of bipedal robots [7, 17, 20, 26, 34, 36–39], quadrupedal robots [40–42], lower extremity powered prostheses [43–45], and exoskeletons [46]. The gait planning for virtual constraint controllers is typically performed offline by solving a nonlinear programming (NLP) problem. While algorithms exist in which agile gait planning can be performed efficiently [26, 47], these methods *cannot* be used in an online manner, thereby restricting the use of HZD methods for

navigating complex environments.

A variety of QP- and MPC-based approaches also exist for locomotion, including LIP-based MPC path planning [42, 48–51], centroidal dynamics-based MPC [52, 53], nonlinear MPC [54], policy-regularized MPC [55], and whole-body QP-based control [56, 57], though LIP and centroidal methods have been studied most extensively. In LIP-based MPC, optimal COM and center of pressure (COP) trajectories are determined subject to the ZMP conditions and feasible net GRF. This approach has the benefit of not requiring offline trajectory optimization and may be coupled with a number of different low-level control techniques to track the prescribed trajectories. While LIP-based MPC has primarily been applied to bipedal systems [48–50], it has recently been extended to quadrupedal locomotion as well, [42, 58]. Centroidal-based MPC techniques determine optimal GRFs using reduced-order dynamics, and has the primary benefit of considering the feasibility of GRFs at each contact point. While these optimization methods have shown to produce robust locomotion, it is a significant computational burden to solve the MPC at every time step even considering the problems are typically formulated as convex QPs. For this reason, event-based optimization is of great interest [42, 59]. This is especially true as the number of decision variables and constraints increase, as is the case when augmented with a serpentine tail.

Motivation and Related Work on Tails for Legged Locomotion: Although the technology involved in the development of sophisticated quadrupedal machines is rapidly advancing, the knowledge regarding the synthesis of feedback control paradigms that intrinsically couple the motion of tails and legged robots for improved dynamic stability is lacking. The use of tails in nature has primarily been studied in terms of dynamic reorientation during leaps and falls [60, 61], but there has been evidence that large cats use their tails for dynamic locomotion as well [62]. Similar to their biological counterpart, the use of biomimetic pendulum-like tails have been explored on bipedal [63–66], quadrupedal [67–69], and hexapedal [70] systems for stability. While these tails have been shown to negate external disturbances [67], perform aerial reorientation [63–65, 69–71] in an effort to regain stability, and increase rapid accelerations [68], the dynamic stability of locomotion for quadrupeds augmented with biomimetic tails has been addressed in very few studies, see e.g., [72, 73]. These investigations primarily explore the use of tails during static stance [67], flight phase [63–65, 69, 70], and gait initiation [68], allowing for simplified models and controllers which *cannot* easily be generalized to varying single- and multi-contact domains present during quadrupedal locomotion. Additionally, control of legged systems with the addition of serpentine appendages (as opposed to pendulum-like appendages)–which greatly increases the nonlinearities present–has been largely unexplored.

Significant Challenges: Existing nonlinear controller design approaches for legged locomotion are tailored specifically to systems without the addition of tails, and the integration of tails introduces significant overhead in terms of
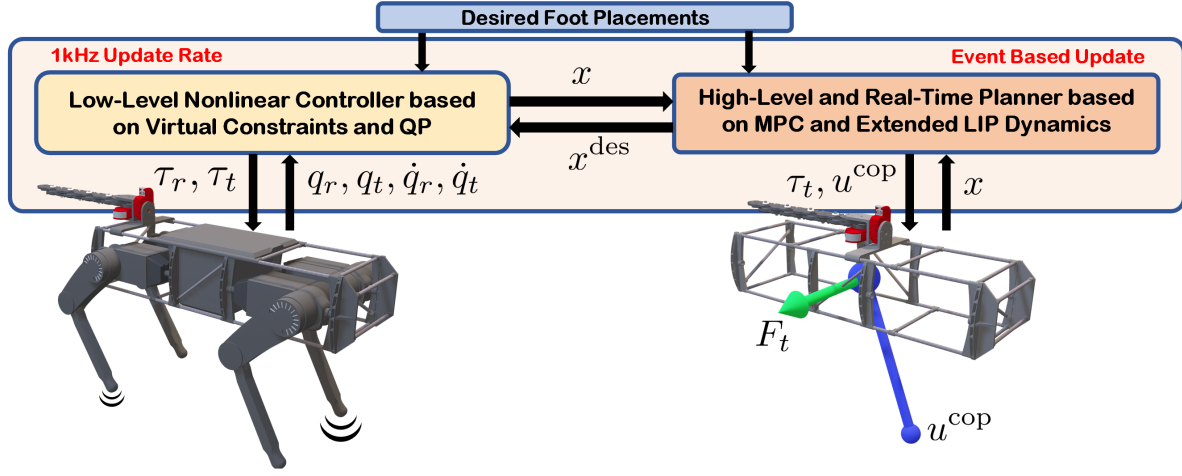
Fig. 2. Block diagram of the proposed hierarchical control scheme. A high-level and real-time planner, based on event-based MPC, computes the optimal COM and tail trajectories for a reduced-order model, referred to as the extended LIP dynamics, subject to the feasibility of the net GRF and tail motion. A low-level nonlinear controller, based on virtual constraints and QP, imposes the full-order dynamical model of locomotion to track the prescribed and reduced-order optimal trajectories subject to the feasibility of the GRFs at the contacting leg ends.

complexity and dimensionality of these already sophisticated machines. This, in turn, further complicates the design of robust nonlinear control algorithms. We aim to answer the following *questions*: 1) how can we extend the LIP dynamics to include external forces induced by a tail? 2) how can we systematically plan for the motion of the tail in real time to ensure robust locomotion? and 3) how can we implement these reduced-order trajectories on a full-order model using nonlinear control approaches?

1.2 Goals, Objectives, and Contributions

The *overarching goal* of this paper is to present a formal foundation to design real-time motion planning and nonlinear feedback control algorithms for robust locomotion of quadrupedal robots augmented with serpentine tails. The *specific objectives* and *key contributions* of the paper are as follows:

1. We present an innovative reduced-order model, referred to as the extended LIP dynamics, for the real-time planning of the COM and tail motions. In particular, the presented reduced-order model integrates the LIP dynamics with the nonlinear tail dynamics to enable us to formulate an online optimal control problem.

2. We present a hierarchical nonlinear control scheme for the motion control of quadrupedal robots with robotic tails. At the higher-level of the control scheme, we propose an MPC formulation for the real-time path planning of the reduced-order model to effectively generate the optimal trajectories for the robot's COM and tail joints. The MPC considers the feasibility of the tail motion, COP constraints, and the feasibility of the net GRF. As the proposed reduced-order model is nonlinear, we linearize the extended LIP model at the beginning of each continuous-time domain and then solve the MPC algorithm in an event-based manner to formulate a convex QP (see Fig. 2).

3. At the lower-level of the control scheme, a nonlinear

controller, based on QP and virtual constraints, is presented for the motion control of the full-order dynamical system to track the prescribed motions of the COM and tail while imposing feasibility of the GRFs at all contact points.

4. We investigate the effect of robotic tails on quadrupedal locomotion intrinsically coupled with the proposed control algorithms. For this purpose, a series of extensive and full-order numerical simulations is presented to demonstrate the effectiveness and robustness of the proposed control approach for locomotion of a 20-DOF advanced quadrupedal robot, Vision 60, augmented with a serpentine tail in the presence of external disturbances (see Fig. 1). We numerically show that the integration of the tail dynamics with the developed nonlinear feedback control algorithms can significantly reduce the effect of external disturbances on quadrupedal locomotion.

References [72] and [73] have provided an interesting approach to address the dynamic stability of legged robots augmented with a single-DOF pendulum-like tail, but in both cases the controller used is a central pattern generator (CPG), where the trajectory is calculated offline and the robot is controlled in an open-loop manner. Reference [72] provided an analysis regarding the general design of the tail using a spring loaded inverted pendulum (SLIP) model, though it is not used as part of any control algorithm and only encompasses movement in the sagittal plane. Our work differs completely in that, 1) we consider the use of a serpentine tail, 2) we propose real-time planning algorithms for both locomotion and the tail by using an extended LIP model, and 3) we track the generated reduced-order trajectories in a closed-loop manner through a QP-based virtual constraint controller. The work presented in the current paper also differs from the previous work [42] in that [42] did *not* address locomotion with tail dynamics. Here, we derive and use a novel extended LIP model in which additional external forces may be addressed. Furthermore, the current work

includes the dynamics of the appendage in the path planner in order to increase gait stability and robustness as opposed to treating the appendage as a disturbance. We also extend the event-based MPC formulation of [42] to the *nonlinear* reduced-order model, arising from the extended LIP dynamics and plan for the optimal COM motion of the robot as well as the tail trajectory.

The paper is organized as follows. The full-order nonlinear models for quadrupedal locomotion with robotic tails are presented in Section 2. Section 3 presents the extended reduced-order model for real-time planning. Section 4 addresses the optimal control problem and formulation of the higher-level MPC for the extended LIP model. The lower-level nonlinear controller, based on QP and virtual constraints, for the whole-body motion control is presented in Section 5. Section 6 presents the numerical simulations for quadrupedal locomotion with tails to verify the effectiveness of the proposed control algorithms. Finally, Sections 7 and 8 present discussion and concluding remarks, respectively.

2 Full-Order Nonlinear Model of Locomotion with Tail
2.1 Robot and Tail Model

In this paper, we consider the full-order dynamical model for locomotion of the quadrupedal robot Vision 60, made by Ghost Robotics[1], augmented with a serpentine tail (see Fig. 1). Vision 60 can be modeled with 18 DOFs, 12 of which consist of leg DOFs while the other 6 represent the absolute position and orientation of the floating base in an inertial world frame. Each leg of the robot consists of an actuated 2 DOF hip joint (roll and extension) plus a 1 DOF knee joint, ending with a point foot. The tail design used in this paper has 2 DOFs, one allows for a rolling motion, while the other allows for yaw motion (see Fig. 3). The yaw portion of the tail consists of eight holonomically constrained gears which are driven by a single actuator. The gear ratio is held constant from one link to the next, therefore each link is constrained to rotate by the same amount as the previous link in order to obtain uniform deformation along the length of the tail. This design is partially inspired by the designs found in [74] and [71]. To keep the tail mechatronics light and simple, each actuated joint uses a single Hebi Robotics[2] actuator which will provide sufficient torque to drive the tail. We note that the total mass of the tail is approximately 1.51 (kg), in which the base, roll link, and individual yaw links are approximately 0.4 (kg), 0.68 (kg), and 0.053 (kg), respectively. Throughout this paper, the tail mechanism is assumed to be mounted on a point, referred to as the base of the robot (see Fig. 1).

2.2 Nonlinear Dynamics

The generalized coordinates for Vision 60 are expressed as $q_r := \text{col}(p_r, \alpha_r, q_b) \in Q_r \subset \mathbb{R}^{18}$, in which the subscript "r" stands for the robot, $p_r \in \mathbb{R}^3$ and $\alpha_r \in \mathbb{R}^3$ represent the absolute position and orientation of the body of the robot in

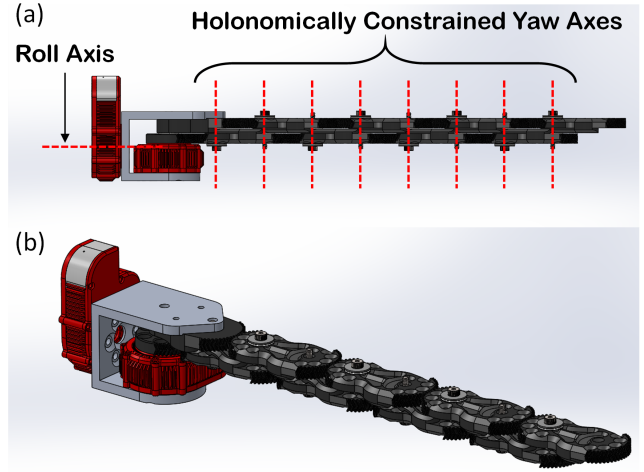(a) Holonomically Constrained Yaw Axes
Roll Axis

(b)

Fig. 3. (a) Side view of the tail CAD design annotated with the axes of rotation. (b) Isometric view of the tail CAD design.

the inertial world frame, respectively, and $q_b \in \mathbb{R}^{12}$ denotes the body angles (i.e., shape of the robot). The torque and force inputs applied to the robot are denoted by $u_r \in \mathbb{R}^{12}$, $\lambda_c \in \mathbb{R}^{3n_c}$, and $\lambda_t \in \mathbb{R}^6$, which are the actuator torque inputs, the GRFs at the contacting leg ends, and the tail reaction wrench (forces and moments), respectively. In our notation, "col" represents the column vector and n_c denotes the number of contacting legs with the ground. The evolution of the robot is then described by the Euler-Lagrange equations and principle of virtual work as follows:

$$D_r(q_r)\ddot{q}_r + H_r(q_r, \dot{q}_r) = B_r \tau_r + J_c^\top(q_r)\lambda_c + J_{br}^\top(q_r)\lambda_t, \quad (1)$$

where $D_r(q_r) \in \mathbb{R}^{18 \times 18}$ is the symmetric and positive definite mass-inertia matrix, $H_r(q_r, \dot{q}_r) \in \mathbb{R}^{18}$ denotes the Coriolis, centrifugal, and gravitational terms, and $B_r \in \mathbb{R}^{18 \times 12}$ is the input distribution matrix. In addition, $J_c(q_r) \in \mathbb{R}^{3n_c \times 18}$ is the ground contact Jacobian matrix, and $J_{br}(q_r) := \frac{\partial p_{br}}{\partial q_r}(q_r) \in \mathbb{R}^{6 \times 18}$ is the Jacobian of the base of the robot on which the tail is connected.

The tail is modeled with 8 DOFs in total, where 6 DOFs come from the floating base, and the additional 2 DOFs are the internal roll and yaw motions of the tail. The generalized coordinates of the tail are then defined as $q_t := \text{col}(p_t, \alpha_t, \theta) \in Q_t \subset \mathbb{R}^8$, where the subscript "$t$" stands for the tail, $p_t \in \mathbb{R}^3$ and $\alpha_t \in \mathbb{R}^3$ represent the absolute position and orientation of the base of the tail in the inertial world frame, respectively, and $\theta := \text{col}(\theta_{\text{roll}}, \theta_{\text{yaw}}) \in \mathbb{R}^2$ are the internal roll and yaw angles. Lastly, $\tau_t := \text{col}(\tau_{\text{roll}}, \tau_{\text{yaw}}) \in \mathbb{R}^2$ are the torque inputs to the tail roll and yaw actuators. The equations of motion can be described as follows:

$$D_t(q_t)\ddot{q}_t + H_t(q_t, \dot{q}_t) = B_t \tau_t - J_{bt}^\top(q_t)\lambda_t \quad (2)$$

where $D_t(q_t) \in \mathbb{R}^{8 \times 8}$, $H_t(q_t, \dot{q}_t) \in \mathbb{R}^8$, and $B_t \in \mathbb{R}^{8 \times 2}$ denote the mass-inertia matrix, Coriolis, centrifugal, and gravitational forces, and input distribution matrix, respectively.

Furthermore, $J_{bt}(q_t) := \frac{\partial p_{bt}}{\partial q_t}(q_t) \in \mathbb{R}^{6\times 8}$ is the Jacobian of the base of the tail in the tail inertial frame. The holonomic constraints between the robot and tail can be expressed as $p_{br}(q_r) = p_{bt}(q_t)$, in which $p_{br}(q_r) \in \mathbb{R}^6$ and $p_{bt}(q_t) \in \mathbb{R}^6$ represent the absolute position and orientation of the base of the tail in q_r and q_t coordinates, respectively. Differentiating the holonomic constraints twice yields

$$J_{br}(q_r)\ddot{q}_r + \dot{J}_{br}(q_r,\dot{q}_r) = J_{bt}(q_t)\ddot{q}_t + \dot{J}_{bt}(q_t,\dot{q}_t). \quad (3)$$

Equations (1)-(3) may then be put into a system of algebraic differential equations to solve for the acceleration of each body and the interacting forces between them, that is,

$$\begin{bmatrix} D_r & \mathbf{0} & -J_{br}^\top \\ \mathbf{0} & D_t & J_{bt}^\top \\ J_{br} & -J_{bt} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{q}_r \\ \ddot{q}_t \\ \lambda_t \end{bmatrix} = \begin{bmatrix} B_r\tau_r - H_r + J_c^\top\lambda_c \\ B_t\tau_t - H_t \\ \dot{J}_{bt} - \dot{J}_{br} \end{bmatrix}, \quad (4)$$

where $\mathbf{0}$ represents a matrix of zeros of the correct dimension. Note that the final equation in (4) ensures that there is a rigid connection between the tail and the quadruped. By eliminating the Lagrange multiplier λ_t (i.e., internal wrench), the equations of motion in the nonlinear coupled dynamics (4) can be expressed as an input-affine system as follows:

$$\dot{z} = f(z) + g(z)\tau + w(z)\lambda_c, \quad (5)$$

where $z := \mathrm{col}(q_r, q_t, \dot{q}_r, \dot{q}_t)$ denotes the state vector and $\tau := \mathrm{col}(\tau_r, \tau_t)$ represents the control inputs.

3 Extended LIP Model

The objective of this section is to derive a reduced-order model based on the LIP dynamics to study the equations of motion for the COM and the tail dynamics. The model, referred to as the extended LIP dynamics, will be utilized for the development of the real-time motion planning algorithm in Section 4.

The reduced-order LIP model can be described by [51]

$$\begin{bmatrix} \ddot{x}^{\mathrm{com}} \\ \ddot{y}^{\mathrm{com}} \end{bmatrix} = \frac{g_0}{z^{\mathrm{com}}} \begin{bmatrix} x^{\mathrm{com}} - x^{\mathrm{cop}} \\ y^{\mathrm{com}} - y^{\mathrm{cop}} \end{bmatrix}, \quad (6)$$

where x^{com} and y^{com} represent the Cartesian coordinates of the COM with respect to the inertial world frame in the x and y directions, respectively, having been projected onto the xy-plane. In addition, z^{com} is the vertical height of the center of mass, which is assumed to remain constant throughout the duration of the gait, g_0 is the gravitational constant, and $u^{\mathrm{cop}} := \mathrm{col}(x^{\mathrm{cop}}, y^{\mathrm{cop}}) \in \mathbb{R}^2$ represents the Cartesian coordinates of the COP in the world frame. In order to include external forces in the LIP model, we consider the following dynamics

$$\begin{bmatrix} \ddot{x}^{\mathrm{com}} \\ \ddot{y}^{\mathrm{com}} \end{bmatrix} = \frac{1}{m_{\mathrm{tot}}} \begin{bmatrix} F_x + F_{gx} + F_{M_px} + F_{M_yx} \\ F_y + F_{gy} + F_{M_ry} + F_{M_yy} \end{bmatrix}, \quad (7)$$
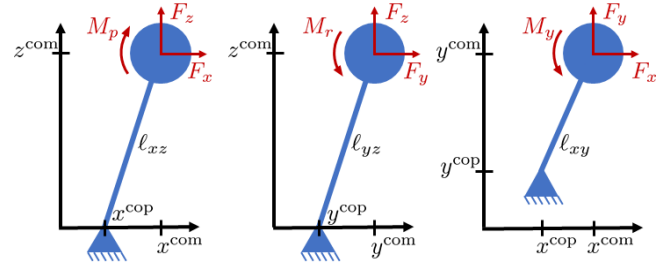

Fig. 4. Free body diagram of the forces and moments induced by the tail on the LIP dynamics. Note that the z^{cop} is always zero.

where m_{tot} is the total robot mass, F_x denotes the external force in the x direction applied by the tail, F_{gx} represents the GRF in the x direction, F_{M_rx} denotes the equivalent force in the x direction due to an external roll moment, F_{M_px} represents the equivalent force in the x direction due to an external pitch moment, and F_{M_yx} denotes the equivalent force in the x direction due to an external yaw moment (see Fig. 4). The notation is defined similarly for both the y and z directions. In order to solve for the net GRF, we consider the equation of motion in the z direction as

$$m_{\mathrm{tot}}\ddot{z}^{\mathrm{com}} = F_z + F_{gz} + F_{M_rz} + F_{M_pz} - m_{\mathrm{tot}}g_0 \quad (8)$$

and let $\ddot{z}^{\mathrm{com}} = 0$ for the duration of the gait. For this purpose, we first study the components of the GRF and the equivalent forces due to moments according to the geometry of the model. In particular, it can be shown that $F_{gx} = F_g\frac{x^{\mathrm{com}}-x^{\mathrm{cop}}}{l}$, $F_{gy} = F_g\frac{y^{\mathrm{com}}-y^{\mathrm{cop}}}{l}$, and $F_{gz} = F_g\frac{z^{\mathrm{com}}}{l}$, where F_g denotes the net GRF and $l := \sqrt{(x^{\mathrm{com}}-x^{\mathrm{cop}})^2 + (y^{\mathrm{com}}-y^{\mathrm{cop}})^2 + (z^{\mathrm{com}})^2}$. In addition, we can easily translate the moments into equivalent forces as follows (see Fig. 4)

$$F_{M_px} = \frac{M_p}{\ell_{xz}^2}z^{\mathrm{com}}, \qquad F_{M_pz} = \frac{-M_p}{\ell_{xz}^2}(x^{\mathrm{com}}-x^{\mathrm{cop}})$$

$$F_{M_ry} = \frac{-M_r}{\ell_{yz}^2}z^{\mathrm{com}}, \qquad F_{M_rz} = \frac{M_r}{\ell_{yz}^2}(y^{\mathrm{com}}-y^{\mathrm{cop}})$$

$$F_{M_yx} = \frac{-M_y}{\ell_{xy}^2}(y^{\mathrm{com}}-y^{\mathrm{cop}}), \quad F_{M_yy} = \frac{M_y}{\ell_{xy}^2}(x^{\mathrm{com}}-x^{\mathrm{cop}}),$$

where $\ell_{xz} := \sqrt{(x^{\mathrm{com}}-x^{\mathrm{cop}})^2 + (z^{\mathrm{com}})^2}$ represents the distance between the COM and COP in the xz-plane, and similarly, $\ell_{yz} := \sqrt{(y^{\mathrm{com}}-y^{\mathrm{cop}})^2 + (z^{\mathrm{com}})^2}$ and $\ell_{xy} := \sqrt{(x^{\mathrm{com}}-x^{\mathrm{cop}})^2 + (y^{\mathrm{com}}-y^{\mathrm{cop}})^2}$. These geometric relations together with (8) and $\ddot{z}^{\mathrm{com}} = 0$ finally result in the following net GRF

$$F_g = \frac{l}{z^{\mathrm{com}}}\left(m_{\mathrm{tot}}g_0 - F_z + \frac{M_p(x^{\mathrm{com}}-x^{\mathrm{cop}})}{(x^{\mathrm{com}}-x^{\mathrm{cop}})^2 + (z^{\mathrm{com}})^2} \right.$$
$$\left. - \frac{M_r(y^{\mathrm{com}}-y^{\mathrm{cop}})}{(y^{\mathrm{com}}-y^{\mathrm{cop}})^2 + (z^{\mathrm{com}})^2} \right). \quad (9)$$

The GRF in (9) can then be combined with (7) to obtain the extended LIP dynamics which are nonlinear due to the addition of several nonlinear components, including F_z and the
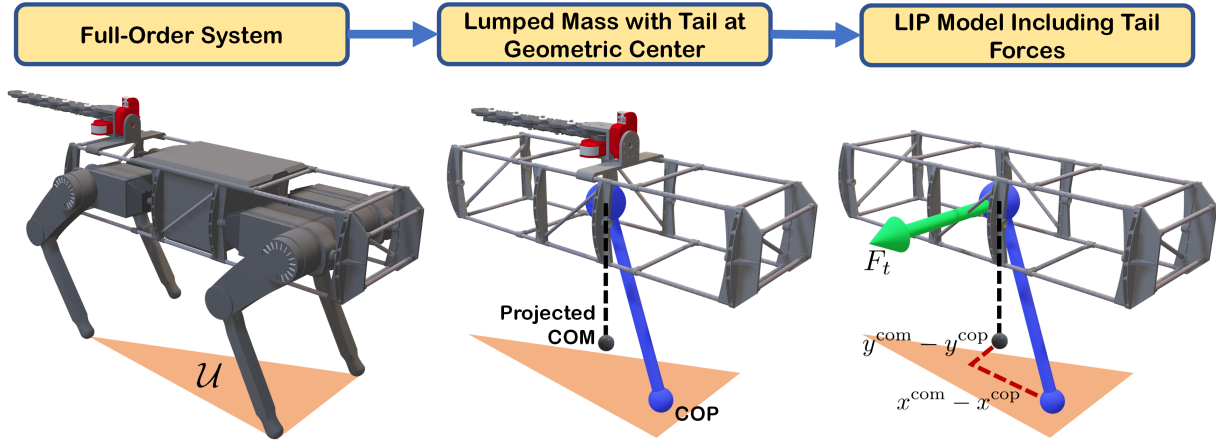
Fig. 5. Illustration of the reduced-order extended LIP model.

moments about the base of the tail (i.e., M_r, M_p, and M_y). Since the dominant external wrenches that are applied by the tail are assumed to be those involving F_x and F_y, the vertical force and the moments induced about the base of the tail are neglected for the purpose of planning, but all forces and moments are considered during the full-order controller and simulations (see Sections 5 and 6). Furthermore, ℓ_{xy} goes to zero as the COM approaches the COP which causes $F_{M_y x}$ and $F_{M_y y}$ to become undefined due to division by zero. We therefore plan only for the standard COP inputs (i.e., x^{cop} and y^{cop}) together with F_x and F_y which are to be applied by the tail. We can show that this assumption will reduce the nonlinear dynamics of the extended LIP model to the following linear dynamics

$$\begin{bmatrix} \ddot{x}^{\text{com}} \\ \ddot{y}^{\text{com}} \end{bmatrix} = \frac{g_0}{z^{\text{com}}} \begin{bmatrix} x^{\text{com}} - x^{\text{cop}} \\ y^{\text{com}} - y^{\text{cop}} \end{bmatrix} + \frac{1}{m_{\text{tot}}} \begin{bmatrix} F_x \\ F_y \end{bmatrix}. \quad (10)$$

3.1 Addition of Tail Dynamics

In order to properly plan for the movement of the tail in the high-level MPC, the nonlinear tail dynamics must be integrated with (10). For the sake of keeping the reduced-order model as simple as possible while capturing the overall dynamics, the tail is assumed to be attached to the geometric center of the robot, as shown in Fig. 5. Due to the nature of B_t and J_b, the equations of motion in (2) can be rewritten more concisely as

$$D_t(q_t)\ddot{q}_t + H_t(q_t, \dot{q}_t) = \begin{bmatrix} -F_t \\ -M_t \\ \tau_t \end{bmatrix}, \quad (11)$$

where $\lambda_t = \text{col}(F_t, M_t)$ with $F_t := \text{col}(F_x, F_y, F_z) \in \mathbb{R}^3$ being the forces at the base of the tail in the x, y, and z directions, and $M_t := \text{col}(M_r, M_p, M_y) \in \mathbb{R}^3$ being the moments at the base of the tail in the roll, pitch, and yaw directions.

Considering that we are only interested in the forces in the x and y directions, only the first two rows of (11) are used for the high-level MPC. We may then solve for the reaction forces caused by the tail in the x and y directions by noting

the tail mass matrix, D_t, and nonlinear vector, H_t, may be decomposed into individual parts, denoted by D_{ij} for $1 \leq i, j \leq 8$ and H_i for $1 \leq i \leq 8$, respectively. More specifically, we have

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} -D_{11}\ddot{x}^{\text{com}} - D_{17}\ddot{\theta}_{\text{roll}} - D_{18}\ddot{\theta}_{\text{yaw}} - H_1 \\ -D_{22}\ddot{y}^{\text{com}} - D_{27}\ddot{\theta}_{\text{roll}} - D_{28}\ddot{\theta}_{\text{yaw}} - H_2 \end{bmatrix}, \quad (12)$$

where θ_{roll} and θ_{yaw} denote the actuated roll and yaw angles (i.e., shape variables) of the tail. Here, we assume that the roll, pitch, and yaw of the quadruped (and the base of the tail) are zero, and z^{com} is constant. Hence, (12) does not contain the portions of the mass-inertia matrix pertaining to the orientation and height of the base of the tail. We further remark that $D_{12} = D_{21} = 0^3$. Combining (10) and (12), the LIP equations with external forces in the x and y directions may then be written as

$$\begin{bmatrix} \ddot{x}^{\text{com}} \\ \ddot{y}^{\text{com}} \end{bmatrix} = \begin{bmatrix} \frac{g_0}{z^{\text{com}}}(x^{\text{com}} - x^{\text{cop}}) + \frac{-D_{11}\ddot{x}^{\text{com}} - D_{17}\ddot{\theta}_{\text{roll}} - D_{18}\ddot{\theta}_{\text{yaw}} - H_1}{m_{\text{tot}}} \\ \frac{g_0}{z^{\text{com}}}(y^{\text{com}} - y^{\text{cop}}) + \frac{-D_{22}\ddot{y}^{\text{com}} - D_{27}\ddot{\theta}_{\text{roll}} - D_{28}\ddot{\theta}_{\text{yaw}} - H_2}{m_{\text{tot}}} \end{bmatrix}. \quad (13)$$

Similarly, the dynamics for the shape of the tail (i.e. $\ddot{\theta}_{\text{roll}}$ and $\ddot{\theta}_{\text{yaw}}$) may be written as

$$\begin{bmatrix} D_{71}\ddot{x}^{\text{com}} + D_{72}\ddot{y}^{\text{com}} + D_{77}\ddot{\theta}_{\text{roll}} + D_{78}\ddot{\theta}_{\text{yaw}} \\ D_{81}\ddot{x}^{\text{com}} + D_{82}\ddot{y}^{\text{com}} + D_{87}\ddot{\theta}_{\text{roll}} + D_{88}\ddot{\theta}_{\text{yaw}} \end{bmatrix} = \begin{bmatrix} \tau_{\text{roll}} - H_7 \\ \tau_{\text{yaw}} - H_8 \end{bmatrix}. \quad (14)$$

Combining (13) and (14), and moving all acceleration terms to the left hand side, we obtain the final equations for the reduced-order model as a set of algebraic differential equations of the form

$$W(x)\dot{x} = Ax + Bu + N(x), \quad (15)$$

where $x := \text{col}(x^{\text{com}}, y^{\text{com}}, \theta_{\text{roll}}, \theta_{\text{yaw}}, \dot{x}^{\text{com}}, \dot{y}^{\text{com}}, \dot{\theta}_{\text{roll}}, \dot{\theta}_{\text{yaw}}) \in \mathbb{R}^8$ and $u := \text{col}(x^{\text{cop}}, y^{\text{cop}}, \tau_{\text{roll}}, \tau_{\text{yaw}}) \in \mathbb{R}^4$ denote the state

---

[3]Due to the nature of the floating base coordinates, the upper-left 3×3 block of the D matrix is diagonalized.

and control inputs, respectively. In addition, $W(x) \in \mathbb{R}^{8 \times 8}$ is a weighting matrix, $A \in \mathbb{R}^{8 \times 8}$ represents the state matrix, $B \in \mathbb{R}^{4 \times 8}$ denotes the input distribution matrix, and $N(x) \in \mathbb{R}^8$ contains the remaining nonlinear terms.

3.2 Linearization of the Extended LIP Model

The final reduced-order equations (15) are linearized by taking the first order Taylor series expansion. Due to the complexity of symbolically inverting the W matrix, the Taylor series expansion is applied to both the left and right hand sides of the equation separately yielding

$$
G(x, \dot{x}) := W(x)\dot{x}
$$
$$
\approx G(x_0, \dot{x}_0) + \frac{\partial G}{\partial x}\bigg|_{(x_0, \dot{x}_0)} (x - x_0) + \frac{\partial G}{\partial \dot{x}}\bigg|_{(x_0, \dot{x}_0)} (\dot{x} - \dot{x}_0) \tag{16}
$$

and

$$
F(x, u) := Ax + Bu + N(x)
$$
$$
\approx F(x_0, u_0) + \frac{\partial F}{\partial x}\bigg|_{(x_0, u_0)} (x - x_0) + \frac{\partial F}{\partial u}\bigg|_{(x_0, u_0)} (u - u_0), \tag{17}
$$

where x_0, $\dot{x}_0$, and u_0 are the operating points around which the system is linearized. Combining (16) and (17) and solving for $\dot{x}$, we obtain the linearized equations of motion

$$
\dot{x} = \frac{\partial G}{\partial \dot{x}}\bigg|_{(x_0, \dot{x}_0)}^{-1} \left(\frac{\partial F}{\partial x}\bigg|_{(x_0, u_0)} - \frac{\partial G}{\partial x}\bigg|_{(x_0, \dot{x}_0)} \right) (x - x_0)
$$
$$
+ \frac{\partial G}{\partial \dot{x}}\bigg|_{(x_0, \dot{x}_0)}^{-1} \frac{\partial F}{\partial u}\bigg|_{(x_0, u_0)} (u - u_0) + \dot{x}_0 \tag{18}
$$

as $G(x_0, \dot{x}_0) = F(x_0, u_0)$. For future reference, one can rewrite (18) in a compact form as the following control affine system

$$
\dot{x} = A_\ell (x_0, \dot{x}_0) x + B_\ell (x_0, u_0) u + d_\ell (x_0, \dot{x}_0, u_0), \tag{19}
$$

where the subscript "ℓ" stand for the linearized dynamics.

The dynamics (19) are then discretized via an Euler formulation such that $x_{k+1} = x_k + \dot{x}_k T_s$ where $k \in \mathbb{Z}_{\geq 0}$ and T_s is the sampling time. Using (19), we are left with a set of difference equations of the form

$$
x_{k+1} = A_d x_k + B_d u_k + d, \tag{20}
$$

where $A_d \in \mathbb{R}^{8 \times 8}$ and $B_d \in \mathbb{R}^{8 \times 4}$ are the discrete state and input distribution matrices, respectively, and $d \in \mathbb{R}^8$ is a vector of constants. The aforementioned matrices and vector are constant for a given linearization/discretization, and will be reevaluated in an event-based manner for the real-time path planning problem (see Section 4).

4 High-Level Path Planner

The objective of this section is to formulate a real-time optimal control problem for the extended reduced-order model to compute an optimal trajectory for the COM and tail motions subject to the COP remaining within the support polygon created by the contacting leg ends, feasible net GRF, and joint and actuation limits of the tail. In particular, we formulate a high-level MPC problem to be solved in an event-based manner (i.e., beginning of each continuous-time domain) to drive the state of the extended reduced-order system from an arbitrary initial state, x_i, to some final state, x_f, over a finite number of continuous time domains, $M \geq 1$. We consider a general locomotion pattern with a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with the vertices set $\mathcal{V}$ and the edges set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ (see Fig. 6). The vertices denote the continuous-time domains of locomotion including double-, triple-, and quadruple-contact phases, and edges represent the discrete-time transitions between the continuous-time domains. Each continuous-time domain is assumed to have $N_d \geq 1$ evenly temporally spaced grid points over which the trajectory is optimized. In order to indicate the continuous-time domain for every time sample, we define the index function as $\zeta : \mathbb{Z}_{\geq 0} \rightarrow \{1, 2, \cdots, M\}$ by $\zeta[k] := \lfloor \frac{k}{N_d} \rfloor + 1$ for all $0 \leq k < MN_d$ and $\zeta[k] := M$ for $k \geq MN_d$. In our notation, $\lfloor . \rfloor$ represents the floor function.

MPC Formulation: We formulate an event-based optimal control problem for the real-time planning of the extended reduced-order model. In particular, we first linearize the nonlinear extended LIP dynamics (15) about its current operating points at the event times (beginning of each continuous-time domain), that is, $k = rN_d$ for some $r \in \mathbb{Z}_{\geq 0}$. We then set up an MPC problem using the above-mentioned linearization over a finite control horizon $N_c = nN_d$ for some $n \geq 1$. This reduces the optimal control problem of the nonlinear system into QP that can be solved in real time. The optimal control inputs are then employed for N_d grid points of the current domain while we discard the remaining control inputs for the next domains. At the beginning of the next domain (i.e., new event), the Jacobian linearization is updated according to the current state of the nonlinear reduced-order model and a new MPC problem is solved. This iterative procedure continues for all future events.

To make this notion more precise, let us assume that we are at the event time $k = rN_d$ for some $r \in \mathbb{Z}_{\geq 0}$. We next evaluate the Jacobian matrices A_ℓ and B_ℓ together with the vector d_ℓ as discussed in Section 3.2 for the following nonlinear dynamics

$$
W(x_k^o) \dot{x}_k^o = F(x_k^o, u_{k-1}^o), \tag{21}
$$

where x_k^o, $\dot{x}_k^o$, and u_{k-1}^o represent the current state of the original reduced-order system, the time derivative of the state, and the optimal control input for the previous time sample, respectively. Furthermore, the superscript "o" stands for the original system. These matrices are then kept constant for the formulation of the optimal control problem. At the next event, the matrices will be reevaluated and will be utilized again for a new optimal control problem. We now consider
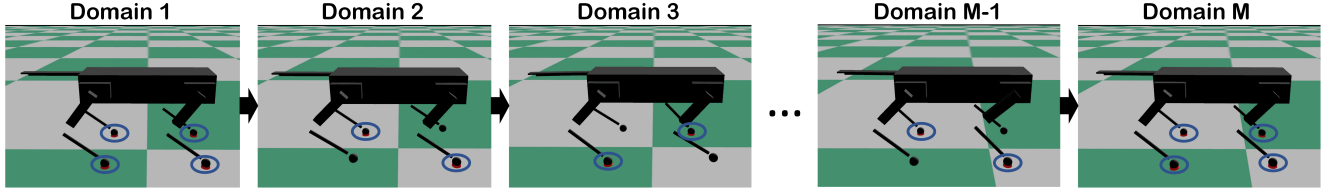
Fig. 6. Snapshots of a nominal trot gait to illustrate the corresponding directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with M continuous-time domains. The first and last vertices of the graph represent quadruple-contact domains for starting and stopping, whereas the middle vertices represent double-contact domains. The contacting leg ends are depicted by the circles.

the affine dynamics (20) with the initial condition $x_k = x_k^o$ for all future times. The optimal control inputs must be constrained such that u_k^{cop} (i.e., COP components of the input) belongs to the convex hull of the contacting points with the ground, $\mathcal{U}_{\zeta[k]} \subset \mathbb{R}^2$, for all domains. That is, $u_k^{\text{cop}} \in \mathcal{U}_{\zeta[k]}$, ensuring that the COP is within the support polygon for all time samples. Additionally, we constrain the net GRF to remain within the friction cone. From the linearized extended LIP dynamics, the friction cone constraints can be given by

$$\pm (A_f x_k + B_f u_k + d_f) \leq \eta, \qquad (22)$$

in which the subscript "f" stands for friction cone. More specifically, $A_f \in \mathbb{R}^{2 \times 8}$ and $B_f \in \mathbb{R}^{2 \times 4}$ are the state and input matrices which are composed of the rows of A_ℓ and B_ℓ pertaining to the accelerations in the x and y directions. In a similar manner, $d_f \in \mathbb{R}^2$ is composed of the appropriate rows of d_ℓ. Lastly, we define $\eta := \mathbf{1}\frac{g_0 \mu}{\sqrt{2}}$ to represent the friction cone that the net GRF must stay within, where μ denotes the static friction coefficient, g_0 represents the gravitational constant, and $\mathbf{1}$ is an appropriately sized vector of ones.

Finally, because the tail joint positions and torques are explicitly included in the path planning problem, we may bound them in order to ensure shape and input feasibility. We define $\tau_{t,k} \in \mathbb{R}^2$ and $\theta_k \in \mathbb{R}^2$ to be the tail torque inputs and the actuated tail configurations at the time step k, respectively. We also define τ_t^{lb}, τ_t^{ub}, θ^{lb}, and θ^{ub} to denote the lower and upper bounds of the tail torque inputs and joint positions. We are now in a position to set up the MPC problem to be solved in an event-based manner (i.e., beginning of each continuous-time domain) in order to create an optimal sequence of inputs (i.e., COP and τ_t) which drives the system from x_0 to x_f over the control horizon. In particular, we consider the following optimal control problem at $k = r N_d$

$$\min_{U_{k \to k+N_c-1|k}} \mathcal{J}_k \left(x_k, U_{k \to k+N_c-1|k} \right) = p \left(x_{k+N_c|k} \right)$$

$$+ \sum_{i=0}^{N_c-1} \mathcal{L} \left(x_{k+i|k}, u_{k+i|k} \right)$$

$$\text{s.t.} \quad x_{k+i+1|k} = A_d x_{k+i|k} + B_d u_{k+i|k} + d$$
$$u_{k+i|i}^{\text{cop}} \in \mathcal{U}_{\zeta[k+i]}$$
$$\pm (A_f x_{k+i|k} + B_f u_{k+i|k} + d_f) \leq \eta$$
$$\tau_t^{\text{lb}} \leq \tau_{t,k+i|i} \leq \tau_t^{\text{ub}}$$
$$\theta^{\text{lb}} \leq \theta_{k+i|k} \leq \theta^{\text{ub}}, \quad i = 0, 1, \cdots, N_c - 1, \quad (23)$$

where $U_{k \to k+N_c-1|k} := \text{col}(u_{k|k}, \ldots, u_{k+N_c-1|k})$ denotes the sequence of control inputs and $x_{k+i|k}$ is the predicted state vector at time $k+i$ computed at time k subject to the dynamics $x_{k+i+1|k} = A_d x_{k+i|k} + B_d u_{k+i|k} + d$ with the initial condition $x_{k|k} := x_k$. In a similar manner, we denote $u_{k+i|k}$ to be the COP and tail torque inputs at time $k+i$ computed at time k. The terminal cost is expressed as $p(x_{k+N_c|k}) := ||x_{k+N_c|k} - x_{k+N_c|k}^{\text{des}}||_P^2$ for some symmetric positive definite matrix $P \in \mathbb{R}^{8 \times 8}$. In this notation $x_{k+i|k}^{\text{des}}$ represents the desired state vector at time $k+i$ which is a smooth trajectory beginning at x_k and ending at x_f, and $||x||_P^2 := x^\top P x$. The stage cost is also defined as $\mathcal{L}(x_{k+i|k}, u_{k+i|k}) := ||x_{k+i|k} - x_{k+i|k}^{\text{des}}||_Q^2 + ||u_{k+i|k}||_R^2$ where $Q \in \mathbb{R}^{8 \times 8}$ and $R \in \mathbb{R}^{4 \times 4}$ are symmetric positive definite matrices.

Remark 1. *The matrices A_d and B_d together with the vector d are reevaluated at the beginning of each new continuous-time domain using the current state of the original system and the previous optimal input, and are held constant for the duration of the MPC. The same is true for the friction matrices A_f and B_f, and the friction vector d_f. This reduces the MPC into QP that can be effectively solved in real-time (see Section 6 for more details).*

Remark 2. *The optimal COM and tail motions for the reduced-order model that are prescribed by the MPC for the current domain will be utilized as a reference trajectory for the full-order model in Section 5. In particular, the low-level nonlinear controller will impose the full-order dynamics to asymptotically track the optimal reduced-order trajectory while imposing all contact forces to remain within the friction cone.*

5   Low-Level Nonlinear Controller

This section presents the low-level controller (i.e., whole-body motion controller) which is developed based on virtual constraints and QP to track the optimal reduced-order motions generated by the high-level path planner. Specifically, we derive a nonlinear control algorithm which tracks the COM and tail trajectories prescribed by the event-based MPC for the current domain. Here, we extend the low-level nonlinear control algorithm of [42] to locomotion with robotic tails. While the path planner directly provides torques for the tail, the torques are based upon the linearized model and are not likely to be accurate enough to use on the full-order system. In addition to tracking the trajectories of

the COM and the tail, the low-level controller is responsible for coordinating the motion of the swing legs such that they arrive at the next desired foot placement while ensuring the feasibility of all GRFs at the contacting legs.

We consider the following time-varying virtual constraints

$$y(z,t) := h(q,t) := h_0(q) - h_d(s,\alpha), \qquad (24)$$

where $h_0(q)$ represents a set of holonomic parameters to be controlled and $h_d(s,\alpha)$ denotes the desired evolution of the holonomic outputs h_0 on the gait. In particular, $h_d(s,\alpha)$ is taken as a Bézier polynomial that passes through the discrete waypoints generated by the MPC path planner, in which s is a gait phasing variable taken as the normalized time and α represents the coefficients of the Bézier polynomial. The dimension of the outputs vary depending on the number of contact points with the ground. During quadruple-contact, h_0 consists of 8 components including the COM position, body orientation (i.e., roll, pitch, and yaw), and two internal joints of the tail. The idea here is to impose the full-order system to track the prescribed COM and tail motions while regulating the Euler angles of the floating base. During triple- and double-contact domains, these controlled variables are augmented with the Cartesian coordinates of the swing legs ends for the foot placement. Here, we would like the swing leg ends follow a desired trajectory in the work space starting from the former foothold and ending at the next one. This would increase the dimension of the controlled variables to 11 and 14 during the triple- and double-contact domains, respectively.

Differentiating (24) along (5) we obtain the following output dynamics using I-O linearization

$$\ddot{y} = L_g L_f y(z,t)\tau + L_w L_f y(z,t)\lambda_c + L_f^2 y(z,t) + \frac{\partial^2 y}{\partial t^2}(z,t)$$
$$= -K_D \dot{y} - K_P y, \qquad (25)$$

where $L_g L_f y$, $L_w L_f y$, and $L_f^2 y$ are Lie derivatives and K_P and K_D are positive definite gain matrices. Furthermore, because the GRFs cannot be measured on the quadruped, we estimate the GRFs through the use of a rigid contact model. This assumes that the acceleration of the foot remains zero after contact which may be expressed as $\ddot{p} = J_c(q_r)\ddot{q}_r + \frac{\partial}{\partial q_r}(J_c(q_r)\dot{q}_r)\dot{q}_r = 0$. This in tandem with (5) yields

$$\ddot{p} = L_g L_f p(z)\tau + L_w L_f p(z)\lambda_c + L_f^2 p(z) = 0. \qquad (26)$$

Analogous to [42], we set up a convex QP to solve for the optimal joint-level torques τ for the robot as well as the tail, while imposing the feasibility of GRFs at all contacting

legs. More precisely, we consider the following real-time QP

$$\min_{(\tau,\lambda_c)} \quad \frac{1}{2}\|\tau\|^2$$

$$\begin{bmatrix} L_g L_f y & L_w L_f y \\ L_g L_f p & L_w L_f p \end{bmatrix} \begin{bmatrix} \tau \\ \lambda_c \end{bmatrix} = \begin{bmatrix} -K_D \dot{y} - K_P y - L_f^2 y - \frac{\partial^2 y}{\partial t^2} \\ -L_f^2 p \end{bmatrix}$$

$$\lambda_c \in \mathcal{FC}$$

$$\tau_{\min} \le \tau \le \tau_{\max}, \qquad (27)$$

where $\mathcal{FC}$ denotes the friction cone, and $\tau_{\min}$ and $\tau_{\max}$ represent the lower and upper bounds for the torque inputs, respectively. By formulating the low-level controller in terms of a QP, the controller also becomes more robust to singularities that could arise from matrix inversions during I-O linearization. We remark that the low-level QP for nonlinear control is executed in real-time (e.g., 1kHz) to compute the optimal torques for the full-order model of locomotion, while the higher-level MPC is solved at a slower rate (i.e., in an event-based manner) for trajectory re-planing based on the extended LIP model.

6 Numerical Simulations

The objective of this section is to numerically verify the effectiveness of the proposed control algorithm for robust and stable quadrupedal locomotion with robotic tails. We present a multitude of full-order simulations to demonstrate the increased robustness that an articulated tail provides using the proposed control technique. In all cases, we consider a trot gait with start and stop conditions as shown in Fig. 6.

6.1 Controller Parameters and Nominal Closed-Loop Simulations

The extended LIP model is discretized using a sampling time of $T_s = 80$ (ms). The high-level MPC computes the optimal reduced-order trajectory over a control horizon of $n = 2$ domains, where each domain consists of $N_d = 4$ grid points. We utilize the optimal solution for the current domain, meaning that only the first four optimal trajectory points from the high-level MPC–that correspond to the current domain– are used as inputs to the low-level nonlinear controller. The rest of the parameters for the MPC are taken as $P = \text{diag}(10^4, 10^4, 10^3, 10^3, 10^4, 10^4, 1, 1)$, $Q = \text{diag}(10, 10, 10, 10, 0.1, 0.1, 0.1, 0.1)$, and $R = I_{4\times4}$, where I represents the identity matrix and "diag" denotes the diagonalization operator. Conceptually, we aim to give the MPC the ability to deviate from the desired tail trajectory (which is always 0) and prioritize tracking the desired COM motion. Additionally, we attempt to drive the tail back to zero at the end of the path planner to allow the tail to begin in an uncompromised position at the start of the next domain. However, we put relatively little weight on the velocity of the tail at the end of each domain, thereby allowing the tail to keep momentum. We begin by simulating the original and extended LIP models integrated with the proposed MPC approach to
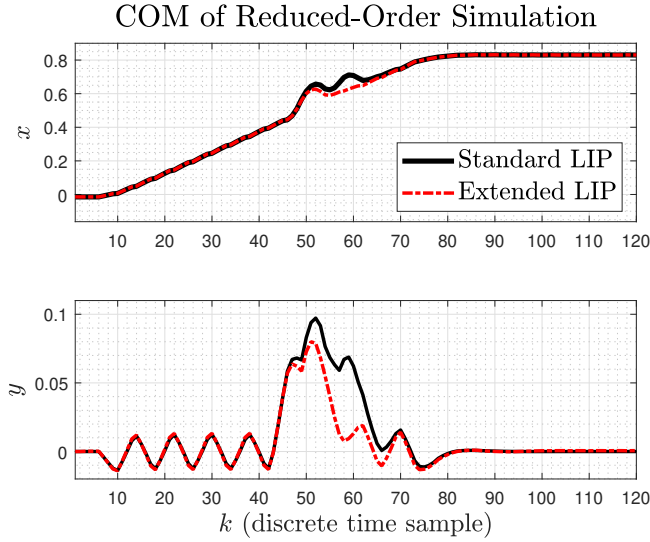
COM of Reduced-Order Simulation

Fig. 7. Reduced-order simulation of the event-based MPC acting on the standard LIP model (without tail) and the extended LIP model (with tail) for a forward trot with a 10 (cm) step length. Here we consider a 30 (N) pulse disturbance in the lateral direction which is applied for the duration of two MPC grid points (i.e., $2T_s$), starting at time sample 41.

show the effectiveness of the higher-level controller in rejecting pulse-like disturbances, see Fig. 7. Here, we consider a forward trot gait with 20 domains subject to a pulse disturbance with a magnitude of 30 (N) in the lateral direction (i.e., y-axis) which is applied for the duration of two grid points, i.e., $2T_s = 160$ (ms), in domain 11. The reduced-order simulation is done in MATLAB and uses ECOS [75] in order to solve the QP arising from the event-based MPC. The stability of final target point for both the standard LIP (without tail) and the extended LIP (with tail) models subject to the event-based MPC is clear, but it is evident that the closed-loop system using the extended LIP model more adequately attenuates the disturbance. In particular, the system without the tail obtains a maximum displacement from the y-axis of approximately 9.7 (cm) while the system with the tail reaches a maximum displacement of 8 (cm) and settles back to steady state faster.

Full-order simulations are then done primarily in RaiSim [76], where we utilize qpSWIFT [77] to solve the high- and low-level QPs. Here, the MPC problem (23) is solved at the beginning of each continuous-time domain, that is approximately every $4T_s = 320$ (ms), whereas the low-level nonlinear control QP is solved at 1kHz. We can show that the QP for the higher-level MPC has 120, 112, and 128 decision variables for the first, middle, and last continuous-time domains, respectively. In addition, the typical computation time of the MPC problem on a desktop computer with an Intel(R) Xeon(R) W-2125 CPU at 4.00 GHz (4 cores) and 16 GB of RAM is 0.32 (ms) to 0.42 (ms). The lower-level QP has 40 decision variables for both the double- and quadruple-contact domains of the trot gait with a typical computation time of 0.19 (ms) to 0.3 (ms).

RaiSim uses a universal robot description file (URDF) representation of the system. This presents a hurdle in that URDFs cannot represent holonomically constrained systems. In order to get around this, a URDF is created assuming that each yaw link in the tail is actuated instead of being holonomically constrained. In order to get the same dynamical behavior of the holonomically constrained tail in Fig. 3, RBDL [78] is then employed to compute the equivalent torque at each yaw joint of the tail via inverse dynamics by using τ_t and λ_t computed by the low-level QP.

We consider three nominal simulations, two of which are conducted in RaiSim, and the third in MATLAB. We consider multiple simulation environments in order to demonstrate the robustness of the closed-loop system against non-parametric uncertainties arising from different contact models. The robot hardware has a small compliant element at the toe, but the provided damping is assumed to be small and the rebound during impact is assumed to be zero. It is therefore reasonable to assume that the compliant dynamics at the toe are negligible when compared to the overall dynamics and impacts may be treated as rigid. For this reason, the majority of the simulations are conducted in RaiSim [76], where a rigid contact model between the leg ends and the ground is used, whereas few simulations are conducted in MATLAB, in which we utilize a LuGre model [79] to represent a compliant contact model. Figures 8 (a) and (b) depict the output (i.e., virtual constraints) and torque profiles of the quadrupedal robot and tail for forward and diagonal trots in RaiSim. Figure 8 (c) shows a full-order simulation of the system run in MATLAB/Simulink subject to LuGre contact model. In addition, Fig. 8 (c) illustrates the components of the ground reaction force experienced at one of the legs. For the purpose of this paper, we use a step length of 10 (cm) for the forward trot and step lengths of (7,4) (cm) in the x and y directions for the diagonal trot.

6.2 Robustness Analysis

In order to study the robustness of the closed-loop system against external disturbances, we consider two different trot gaits, namely forward and diagonal trots. The robustness simulations fall under two different categories, the first being a pulse-like disturbance and the latter being a time-varying disturbance.

6.2.1 Pulse Disturbance Rejection

In order to analyze the closed-loop system's robustness, we investigate the effects of a pulse perturbation applied to the system for 200 (ms) at the COM. First we consider the forward trot gait with two different pulse disturbances, the first being 75 (N) in the lateral direction, and the second being 100 (N) in the forward direction. The virtual constraints and joint-level torques for these simulations are depicted in Figs. 9 (a) and (b). We also consider two additional simulations with the diagonal trot gait, where the first simulation is subject to a pulse disturbance directly opposing the direction of motion (i.e., along the unit vector $(-0.8682, -0.4961, 0)$), and the second is subject to a pulse disturbance which is orthogonal to the direction of motion (i.e., along the unit vector $(0.4961, -0.8682, 0)$). The pulse-
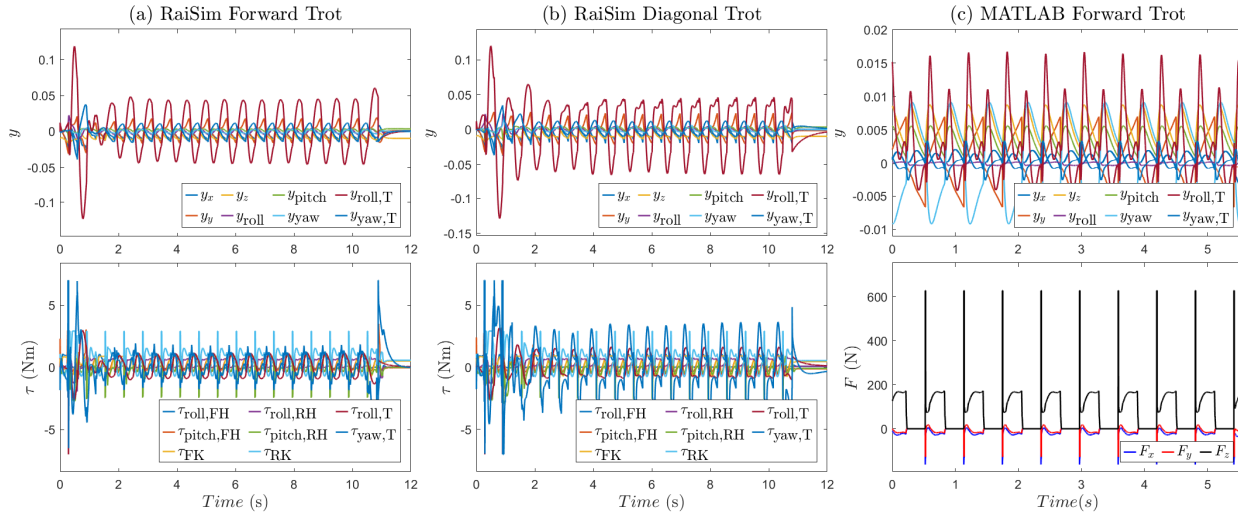
Fig. 8. (a) and (b) Time profiles of the outputs (i.e., virtual constraints) and joint torques for the forward and diagonal trot gaits in RaiSim. (c) Time profile of the virtual constraints and the GRF experienced at one of the leg ends during forward trot gait with a LuGre contact model in MATLAB. Here, we only plot the components of the virtual constraints that correspond to the absolute position of the COM (i.e., y_x, y_y, and y_z), absolute orientation of the robot's floating base (i.e., y_{roll}, y_{pitch}, and y_{yaw}), and shape variables of the tail (i.e., $y_{\mathrm{roll,T}}$ and $y_{\mathrm{yaw,T}}$). Other components are not shown to simplify the illustration. For the torque plots, the subscripts "FH", "FK", "RH", "RK", and "T" stand for the front hip, front knee, rear hip, and rear knee of the left-hand-side of the robot as well as the tail, respectively. In addition, "roll" and "pitch" for the torque plots represent the roll and pitch motions of the joints.

like forces in the latter two simulations have magnitudes of 75 (N) and 110 (N), respectively, and the corresponding virtual constraints and torques are shown in Figs. 9 (c) and (d). The same disturbances were then applied to the system without the addition of the tail and the corresponding plots are provided in Fig. 10 to allow direct comparison. Figure 11 furthermore provides snapshots for the simulations found in Fig. 9 (a) and Fig. 10 (a). It is clear that the quadruped robot with the tail and the proposed control algorithm can successfully reach the final target in the presence of the aforementioned disturbance, whereas its counterpart without the tail cannot reach the target destination.

It is important to note that the ability for the closed-loop system to reject a pulse disturbance is greatly influenced by when the disturbance is applied. Specifically, it depends on what point in the gait cycle the robot is currently in, and the state of the tail at that point. With this in mind, we aim to quantify the level of improvement the tail adds by considering a forward trot with a lateral pulse, where the pulse is applied at different times. In this regard, twelve different simulations were conducted, where in each simulation the pulse was applied 0.1 (s) later than the previous simulation. During each test, the largest magnitude pulse that the system was able to withstand while still reaching the target destination was recorded. On average, the system including the tail was able to sustain 61.7 (N) while still being able to reach the target destination which is an increase of 32.6% compared to its tailless counterpart. A similar procedure was done for the diagonal trot but was instead subject to a disturbance orthogonal to the direction of locomotion (i.e., along the unit vector $(0.4961, -0.8682, 0)$). The system with the tail could reject a pulse of 83.4 (N) on average. This is a 25.6% increase in

magnitude compared to the system without the tail.

6.2.2 Time-Varying Disturbance Rejection

In this section, we consider four additional simulations in which the system is subject to a persistent time-varying disturbance. Similar to the pulse disturbances, we first investigate two forward trots, and then two diagonal trots, each subject to a different persistent disturbance. For each simulation, we note the largest magnitude at which the system can walk for 80 continuous-time domains both with and without the tail in order to compare the results. In the first scenario, we consider a forward trot gait with a lateral disturbance of the form $\beta \sin(4t)$, where the largest magnitude the system with the tail could withstand was $\beta = 21.5$ (N), which is 35.2% larger than the system without the tail. We then consider a second simulation of the forward trot gait subject to a disturbance of the form $\beta(\sin(4t), \cos(4t))$ in the x and y directions. In this case, the system with the tail could reject a maximum of $\beta = 11$ (N) while still being able to walk for 80 domains, up from $\beta = 7.4$ (N) without the tail. Figures 12 (a) and (b) depict the closed-loop system's behavior for the forward trot gait subject to the aforementioned time-varying disturbances. Finally, we consider two different simulations with a diagonal trot, where each system is subject to a disturbance of the form $\beta \sin(4t)$ but in different directions. During the first diagonal simulation, the disturbance acts along the direction of motion $\big($i.e., along the unit vector $(0.8682, 0.4961, 0)\big)$, and in the latter, the disturbance acts along the unit vector $(0.8682, -0.4961, 0)$. During these simulations, the system could reject $\beta = 14.3$ and $\beta = 18$ (N), which is an increase of 20.2% and 73% compared to the tail-
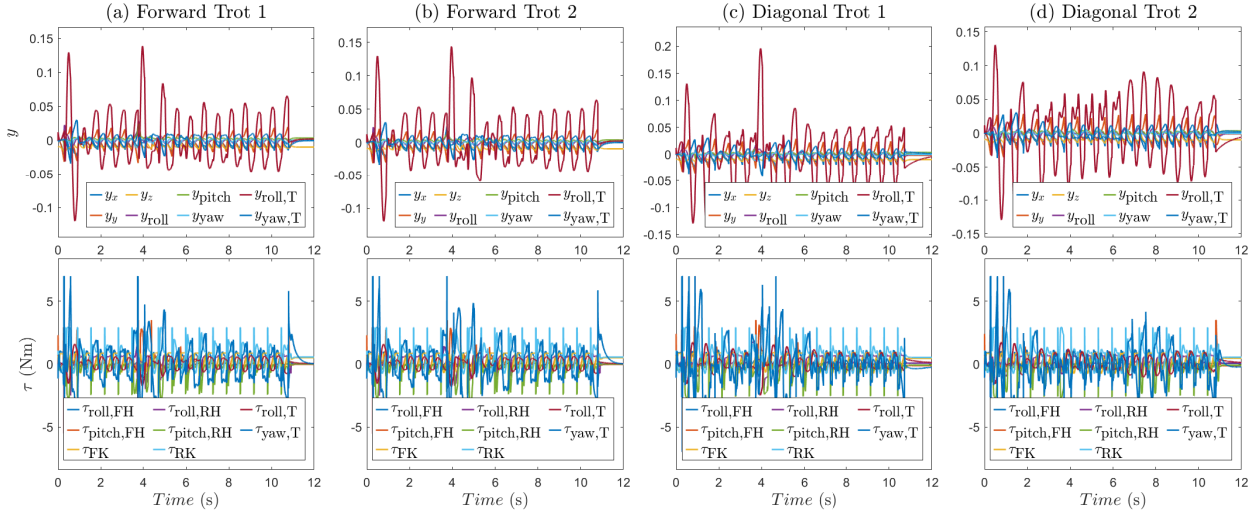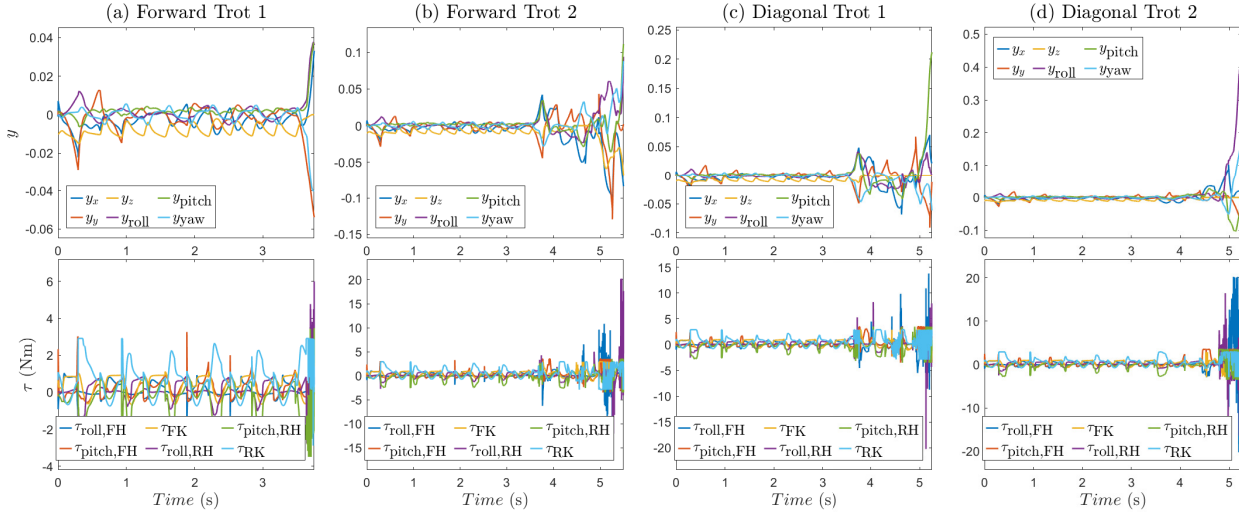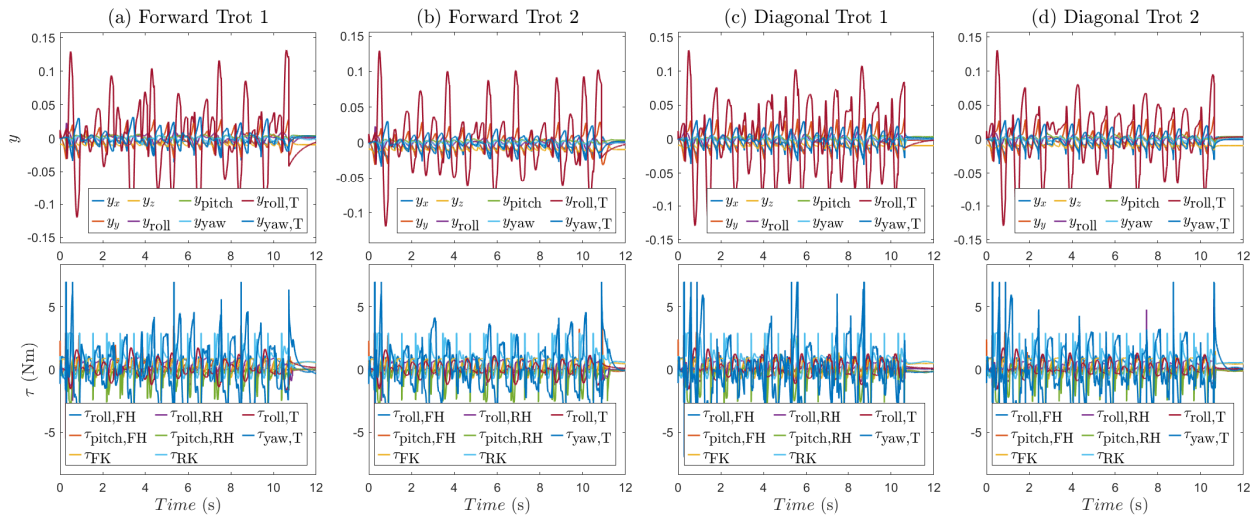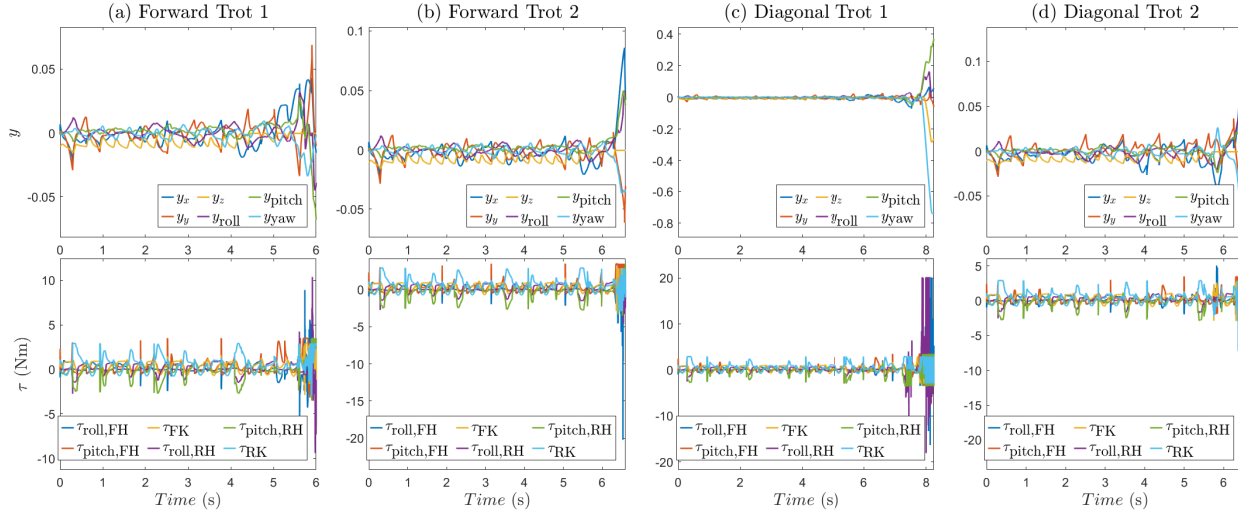
Fig. 9. Time profiles of the outputs (i.e., virtual constraints) and joint torques for quadrupedal locomotion with the robotic tail subject to various pulse disturbances, all beginning at 3.5 (s) and lasting 200 (ms). The proposed controller can successfully reject the effect of disturbances while keeping the output and input profiles bounded. (a) Closed-loop system's behavior for a forward trot gait subject to a pulse disturbance of -75 (N) in the y direction. (b) Closed-loop system's behavior for a forward trot gait subject to a pulse disturbance of 100 (N) in the forward direction. (c) Closed-loop system's behavior for a diagonal trot gait subject to a pulse disturbance of 75 (N) orthogonal to the direction of locomotion in the xy-plane. (d) Closed-loop system's behavior for a diagonal trot gait subject to a pulse disturbance of 110 (N) opposing the direction of locomotion.

Fig. 10. Time profiles of the outputs (i.e., virtual constraints) and joint torques for quadrupedal locomotion without the robotic tail subject to various pulse disturbances, all beginning at 3.5 (s) and lasting 200 (ms). Here, we use the same disturbances that are applied to the system with the robotic tail in Fig. 9. The instability of the gaits is clear. In particular, the outputs diverge after a while with the torques reaching the maximum values. (a) System's behavior for a forward trot gait subject to a pulse disturbance of -75 (N) in the y direction. (b) System's behavior for a forward trot gait subject to a pulse disturbance of 100 (N) in the forward direction. (c) System's behavior for a diagonal trot gait subject to a pulse disturbance of 75 (N) orthogonal to the direction of locomotion. (d) System's behavior for a diagonal trot gait subject to a pulse disturbance of 110 (N) opposing the direction of locomotion.

less system, respectively. Figures 12 (c) and (d) illustrate the time profiles of the virtual constraints and joint torques for these simulation. In order to demonstrate the robustness of the closed-loop system with the tail mechanism, Fig. 13 contains the same simulations having been performed on the system without the tail. In addition, Fig. 14 (a) and (b) provides snapshots of the simulations corresponding to Fig. 12

(b) and Fig. 13 (b), respectively. Animations of these simulations can be found online[4].

---

[4]https://youtu.be/Bi7Zec66nbA
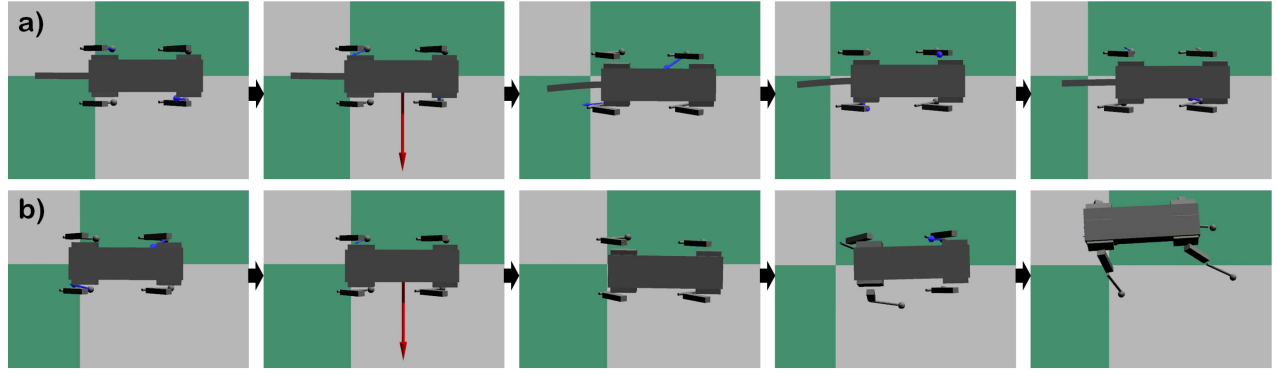
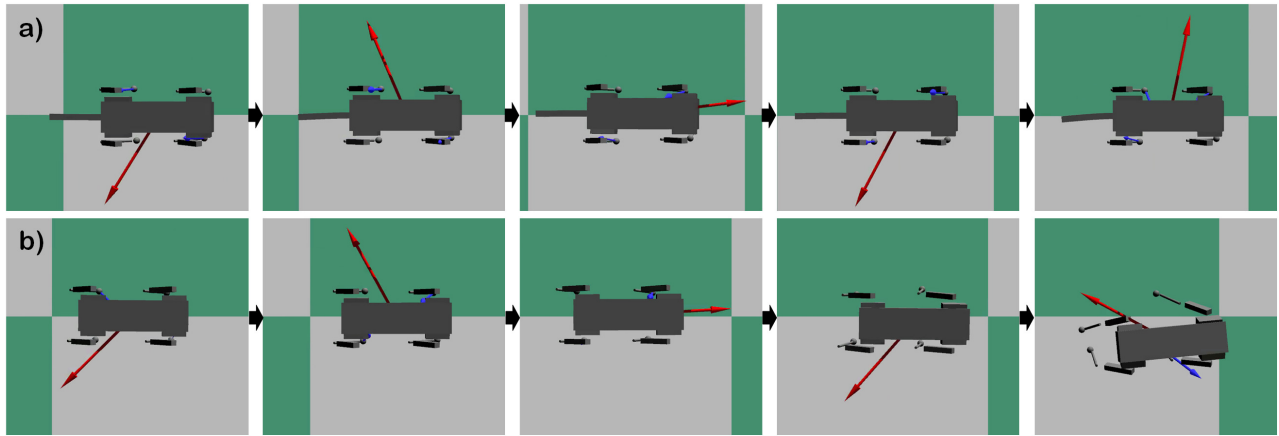Fig. 11. Snapshots of the quadrupedal locomotion with and without robotic tails and subject to the pulse-like disturbance in RaiSim. (a) Snapshots for the closed-loop system's behavior with the robotic tail as simulated in Fig. 9 (a). Here, we consider a forward trot gait subject to a pulse disturbance of -75 (N) in the y direction, starting at 3.5 (s) and lasting 200 (ms). (b) Snapshots for the system's behavior without the robotic tail and the same disturbance as simulated in Fig. 10 (a). The arrow illustrates the disturbance. The instability of the gait without the tail is clear.

Fig. 12. Time profiles of the outputs (i.e., virtual constraints) and joint torques for quadrupedal locomotion with the robotic tail subject to various sinusoidal disturbances. The proposed controller can successfully reject the effect of time-varying disturbances while keeping the output and input profiles bounded. (a) Closed-loop system's behavior for a forward trot gait subject to a disturbance of the form $18\sin(4t)$ (N) in the y direction. (b) Closed-loop system's behavior for a forward trot gait subject to a disturbance of the form $(8.5\sin(4t), 8.5\cos(4t))$ (N) in the x and y directions, respectively. (c) Closed-loop system's behavior for a diagonal trot gait subject to a disturbance of the form $13\sin(4t)$ (N) along the direction of locomotion. (d) Closed-loop system's behavior for a diagonal trot gait subject to a disturbance of the form $13\sin(4t)$ (N) along the unit vector $(0.8682, -0.4961, 0)$.

7 Discussion

The numerical studies and full-order simulations of this paper show significant improvements in rejecting both pulse-like and time-varying disturbances during quadrupedal locomotion integrated with robotic tails and the proposed control algorithms. The amount of improvements depends largely on the gait and the properties of the disturbance applied. The tail mechanism integrated with the proposed hierarchical control algorithm has been shown to allow the closed-loop system to withstand pulse-like disturbances up to 32.6% and 25.6% greater in magnitude during forward and diagonal trots when compared to a system without a tail. Furthermore, the tailed closed-loop system was able to withstand persistent pertur-

bations up to 21.5% and 73% greater in magnitude during forward and diagonal trots, respectively.

It is noteworthy that the tail does not produce aggressive motions in order to reject disturbances. However, because of the nonlinearities of the tail dynamics, it only takes a small angular deflection at the base of the yaw links to deflect the end effector of the tail greatly. In addition, the gaits provided are quasi-static gaits where it stands to reason that the tail motions would be quasi-static as well. In a more aggressive gait, the tail would likely produce faster motions of larger amplitude. Furthermore, the roll of the tail remains relatively constant throughout the gait which indicates that the yaw portion of the tail is more important in maintaining

Fig. 13. Time profiles of the outputs (i.e., virtual constraints) and joint torques for quadrupedal locomotion without the robotic tail subject to various sinusoidal disturbances. Here, we utilize the same time-varying disturbances that are applied to the system augmented with the robotic tail in Fig. 12. The instability of the gaits is clear. In particular, the outputs diverge after a while with the torques reaching the maximum values. (a) System's behavior for a forward trot gait subject to a disturbance of the form $18\sin(4t)$ (N) in the y direction. (b) System's behavior for a forward trot gait subject to a disturbance of the form $(8.5\sin(4t), 8.5\cos(4t))$ (N) in the x and y directions, respectively. (c) System's behavior for a diagonal trot gait subject to a disturbance of the form $13\sin(4t)$ (N) along the direction of locomotion. (d) System's behavior for a diagonal trot gait subject to a disturbance of the form $13\sin(4t)$ (N) along the unit vector $(0.8682, -0.4961, 0)$.

Fig. 14. Snapshots of the quadrupedal locomotion with and without robotic tails and subject to a persistent sinusoidal disturbance in RaiSim. (a) Snapshots for the closed-loop system's behavior with the robotic tail as simulated in Fig. 12 (b). Here, we consider a forward trot gait subject to a time-varying disturbance of the form $(8.5\sin(4t), 8.5\cos(4t))$ (N) in the x and y directions. (b) Snapshots for the system's behavior without the robotic tail and the same disturbance as simulated in Fig. 13 (b). The arrow illustrates the disturbance. The instability of the gait without the tail is clear.

robust and stable locomotion on flat ground.

While increasing the complexity of the reduced-order model to a nonlinear template model such as the inverted pendulum (IP) model [80] or spring-loaded inverted pendulum (SLIP) model [19] could potentially offer increased performance, it comes at the cost of increased computational demand. More specifically, these templates necessitate solving a NLP in real-time for path planning which significantly increases the computational demand compared to working with a linearized model. Furthermore, in the case of quasi-static quadrupedal locomotion, the assumption that the COM

height remains constant is a very good approximation as shown by simulation results above, making the LIP model a suitable choice for the reduced-order model. During more agile locomotion where these assumptions might be violated, a more complex reduced-order model could play a larger role in increasing robustness. However, for the quasi-static case with multi-contact domains considered here, the additional tail hardware can play a larger role in attenuating disturbance forces applied to the robot than increasing model complexity due to the tails ability to directly apply forces to the body of the robot to reject said disturbances.

The numerical results above have shown that the event-based framework coupled with the assumption that the roll, pitch, and yaw of the body are zero, along with the height of the COM being constant, leads to robust locomotion when maneuvering over flat, even terrain. In particular, under these conditions, linearizing the extended LIP model once per domain is sufficient due to the quasi-static nature of the gait and small deflections of the tail. However, the high-level path planner in its current form would likely need to be altered in order to address more complex environments wherein the terrain is unknown or unstructured. This is largely due to several reasons, the first being that in the current framework the desired foot locations are calculated offline, the second being that moving up or down slopes would inherently violate the assumptions regarding the COM height and body orientation, and finally, the event-based linearization/planning may not be sufficient to adequately plan for locomotion along unstructured terrain considering higher planning rates are typically required when the environment is not well known.

8   Conclusion

This paper presented a formal foundation to design hierarchical and nonlinear feedback control algorithms that effectively couple bio-inspired articulated tails with legged robots to produce robust locomotion patterns in the presence of external disturbances. The proposed control approach utilizes a high-level and real-time path planner, based on an event-based MPC, to compute the optimal robot's COM and tail trajectories for an innovative reduced-order model, referred to as the extended LIP dynamics. The MPC problem considers the feasibility of the net GRF and the tail's motion during the steering problem of the extended LIP model. A nonlinear whole-body motion controller, based on virtual constraints and QP, is then utilized in the lower-level of the control scheme to impose the full-order dynamical model of locomotion to track the optimal and reduced-order trajectories prescribed by the MPC, while ensuring the feasibility of the GRFs at the contacting legs. The potential of the proposed control approach was demonstrated through a variety of full-order simulations for an 18-DOF quadruped robot augmented with a 2-DOF serpentine tail. It is shown that the proposed control solution integrated with the tail dynamics can significantly reduce the effect of disturbances on quadrupedal locomotion.

For future research, we will extend the framework to more agile gaits wherein the tail could have an even greater impact on the overall stability of locomotion. We will also study alternative tail designs with different structures and DOF to further investigate the effects of tails on quadrupedal locomotion. Moreover, we will experimentally employ the developed control algorithms for locomotion of the Vision 60 platform coupled with a robotic tail. Finally, we will consider extending the proposed framework to uneven, unstructured, and sloped terrain to allow for greater versatility.

Nomenclature

$(.)_r$, $(.)_t$ Variables corresponding to the robot, variables corresponding to the tail

$(.)_\ell$, $(.)_d$ Variables corresponding to the linearized system, variables corresponding to the discretized system

$D_{(.)}$, $H_{(.)}$ Mass inertia matrix, nonlinear vector (i.e. Coriolis, centrifugal, and gravitational terms)

$B_{(.)}$, $J_{b(.)}$ Input distribution matrix, Jacobian matrix at the base of the tail

q_r, q_t, z Configuration and state variables for the robot, the tail, and the augmented system

θ_{roll}, θ_{yaw} Internal roll and yaw angles of the tail

τ_{roll}, τ_{yaw} Torques applied to the roll and yaw actuators of the tail

τ_r, τ_t, τ Inputs to the robot, the tail, and the augmented system

λ_c, λ_t Ground reaction forces, tail wrench

(f, g, w) Vector fields of the affine continuous-time dynamics

$(.)^{\text{com}}$, $(.)^{\text{cop}}$ Variables corresponding to the center of mass (COM) having been projected onto the xy-plane, variables corresponding to the center of pressure (COP)

F_x, F_y, F_z Forces in the x, y, and z directions

M_r, M_p, M_y Moments in the roll, pitch, and yaw directions

F_t, M_t Forces at the base of the tail, moments at the base of the tail

W, $A_{(.)}$, N, d Extended LIP weighting matrix, state matrix, nonlinear vector function, vector of constants

x, u State variables for the extended LIP model including tail dynamics, inputs to the extended LIP model including tail dynamics

x_k^o, $\dot{x}_k^o$, u_k^o State and input variables for the original and extended reduced-order system at time k

x_0, $\dot{x}_0$, u_0 Operating points around which the augmented LIP model is linearized

$\zeta[k]$, $\mathcal{U}_{\zeta[k]}$ Current domain function, convex hull formed by stance legs

M, n Total number of continuous-time domains, number of continuous-time domains to plan over

N_d, N_c, T_s Number of grid points to plan over, control horizon, time step between grid points

y, h_0, h_d Holonomic output function, holonomic control variables, desired evolution of controlled variables

$L_g L_f$, $L_w L_f$, L_f^2 Lie derivatives along the continuous-time dynamics

References

[1] Vukobratović, M., Borovac, B., and Surla, D., 1990. *Dynamics of Biped Locomotion*. Springer.

[2] Goswami, A., 1999. "Postural stability of biped robots and the foot-rotation indicator (FRI) point". *The International Journal of Robotics Research,* **18**(6), pp. 523–533.

[3] Haddad, W., Chellaboina, V., and Nersesov, S., 2006. *Impulsive and Hybrid Dynamical Systems: Stability, Dissipativity, and Control*. Princeton University Press, July.

[4] Goebel, R., Sanfelice, R., and Teel, A., 2012. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, March.

[5] Grizzle, J., Abba, G., and Plestan, F., 2001. "Asymptotically stable walking for biped robots: Analysis via systems with impulse effects". *IEEE Transactions on Automatic Control,* **46**(1), Jan, pp. 51–64.

[6] Chevallereau, C., Grizzle, J., and Shih, C.-L., 2009. "Asymptotically stable walking of a five-link underactuated 3-D bipedal robot". *IEEE Transactions on Robotics,* **25**(1), Feb, pp. 37–50.

[7] Ames, A., Galloway, K., Sreenath, K., and Grizzle, J., 2014. "Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero dynamics". *IEEE Transactions on Automatic Control,* **59**(4), April, pp. 876–891.

[8] Spong, M., and Bullo, F., 2005. "Controlled symmetries and passive walking". *IEEE Transactions on Automatic Control,* **50**(7), July, pp. 1025–1031.

[9] Spong, M., Holm, J., and Lee, D., 2007. "Passivity-based control of bipedal locomotion". *IEEE Robotics Automation Magazine,* **14**(2), June, pp. 30–40.

[10] Manchester, I., Mettin, U., Iida, F., and Tedrake, R., 2011. "Stable dynamic walking over uneven terrain". *The International Journal of Robotics Research,* **30**(3), pp. 265–279.

[11] Dai, H., and Tedrake, R., 2013. "$\mathcal{L}_2$-gain optimization for robust bipedal walking on unknown terrain". In IEEE International Conference on Robotics and Automation, pp. 3116–3123.

[12] Song, G., and Zefran, M., 2006. "Underactuated dynamic three-dimensional bipedal walking". In IEEE International Conference on Robotics and Automation, pp. 854–859.

[13] Gregg, R., and Righetti, L., 2013. "Controlled reduction with unactuated cyclic variables: Application to 3D bipedal walking with passive yaw rotation". *IEEE Transactions on Automatic Control,* **58**(10), Oct, pp. 2679–2685.

[14] Byl, K., and Tedrake, R., 2008. "Approximate optimal control of the compass gait on rough terrain". In IEEE International Conference on Robotics and Automation, pp. 1258–1263.

[15] Akbari Hamed, K., and Gregg, R. D., 2019. "Decentralized event-based controllers for robust stabilization of hybrid periodic orbits: Application to underactuated 3D bipedal walking". *IEEE Transactions on Automatic*

Control, **64**(6), June, pp. 2266–2281.

[16] Akbari Hamed, K., and Grizzle, J., 2014. "Event-based stabilization of periodic orbits for underactuated 3-D bipedal robots with left-right symmetry". *IEEE Transactions on Robotics,* **30**(2), April, pp. 365–381.

[17] Chevallereau, C., Abba, G., Aoustin, Y., Plestan, F., Westervelt, E., Canudas-de Wit, C., and Grizzle, J., 2003. "RABBIT: A testbed for advanced control theory". *IEEE Control Systems Magazine,* **23**(5), Oct, pp. 57–79.

[18] Morris, B., and Grizzle, J., 2009. "Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots". *IEEE Transactions on Automatic Control,* **54**(8), Aug, pp. 1751–1764.

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

[20] Sreenath, K., Park, H.-W., Poulakakis, I., and Grizzle, J. W., 2011. "Compliant hybrid zero dynamics controller for achieving stable, efficient and fast bipedal walking on MABEL". *The International Journal of Robotics Research,* **30**(9), Aug., pp. 1170–1193.

[21] Collins, S., Ruina, A., Tedrake, R., and Wisse, M., 2005. "Efficient bipedal robots based on passive-dynamic walkers". *Science,* **307**(5712), pp. 1082–1085.

[22] Johnson, A. M., Burden, S. A., and Koditschek, D. E., 2016. "A hybrid systems model for simple manipulation and self-manipulation systems". *The International Journal of Robotics Research,* **35**(11), pp. 1354–1392.

[23] Burden, S. A., Sastry, S. S., Koditschek, D. E., and Revzen, S., 2016. "Event–selected vector field discontinuities yield piecewise–differentiable flows". *SIAM Journal on Applied Dynamical Systems,* **15**(2), pp. 1227–1267.

[24] Vasudevan, R., 2017. *Hybrid System Identification via Switched System Optimal Control for Bipedal Robotic Walking*. Springer International Publishing, Cham, pp. 635–650.

[25] Veer, S., Rakesh, and Poulakakis, I., 2019. "Input-to-state stability of periodic orbits of systems with impulse effects via poincaré analysis". *IEEE Transactions on Automatic Control,* **64**(11), pp. 4583–4598.

[26] Hereid, A., Hubicki, C. M., Cousineau, E. A., and Ames, A. D., 2018. "Dynamic humanoid locomotion: A scalable formulation for HZD gait optimization". *IEEE Transactions on Robotics,* **34**(2), April, pp. 370–387.

[27] Posa, M., Tobenkin, M., and Tedrake, R., 2016. "Stability analysis and control of rigid-body systems with impacts and friction". *IEEE Transactions on Automatic Control,* **61**(6), June, pp. 1423–1437.

[28] Hurmuzlu, Y., and Marghitu, D. B., 1994. "Rigid body collisions of planar kinematic chains with multiple contact points". *The International Journal of Robotics Research,* **13**(1), pp. 82–92.

[29] Akbari Hamed, K., Safaee, B., and Gregg, R. D., 2019.

"Dynamic output controllers for exponential stabilization of periodic orbits for multi-domain hybrid models of robotic locomotion". *ASME Journal of Dynamic Systems, Measurement, and Control,* **141**(12), pp. 121011–1–121011–15.

[30] Ames, A. D., Gregg, R. D., Wendel, E. D. B., and Sastry, S., 2007. "On the geometric reduction of controlled three-dimensional bipedal robotic walkers". In Lagrangian and Hamiltonian Methods for Nonlinear Control 2006, Springer Berlin Heidelberg, pp. 183–196.

[31] Gregg, R. D., and Spong, M. W., 2010. "Reduction-based control of three-dimensional bipedal walking robots". *The International Journal of Robotics Research,* **29**(6), May, pp. 680–702.

[32] Shiriaev, A., Freidovich, L., and Gusev, S., 2010. "Transverse linearization for controlled mechanical systems with several passive degrees of freedom". *IEEE Transactions on Automatic Control,* **55**(4), April, pp. 893–906.

[33] Westervelt, E., Grizzle, J., and Koditschek, D., 2003. "Hybrid zero dynamics of planar biped walkers". *IEEE Transactions on Automatic Control,* **48**(1), Jan, pp. 42–56.

[34] Westervelt, E., Grizzle, J., Chevallereau, C., Choi, J., and Morris, B., 2007. *Feedback Control of Dynamic Bipedal Robot Locomotion.* Taylor & Francis/CRC.

[35] Isidori, A., 1995. *Nonlinear Control Systems.* Springer; 3rd edition.

[36] Sreenath, K., Park, H.-W., Poulakakis, I., and Grizzle, J., 2013. "Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on MABEL". *The International Journal of Robotics Research,* **32**(3), pp. 324–345.

[37] Martin, A. E., Post, D. C., and Schmiedeler, J. P., 2014. "The effects of foot geometric properties on the gait of planar bipeds walking under HZD-based control". *The International Journal of Robotics Research,* **33**(12), pp. 1530–1543.

[38] Ramezani, A., Hurst, J., Akbai Hamed, K., and Grizzle, J., 2013. "Performance analysis and feedback control of ATRIAS, a three-dimensional bipedal robot". *Journal of Dynamic Systems, Measurement, and Control December, ASME,* **136**(2), December.

[39] Akbari Hamed, K., Buss, B., and Grizzle, J., 2016. "Exponentially stabilizing continuous-time controllers for periodic orbits of hybrid systems: Application to bipedal locomotion with ground height variations". *The International Journal of Robotics Research,* **35**(8), pp. 977–999.

[40] Akbari Hamed, K., Ma, W., and Ames, A. D., 2019. "Dynamically stable 3D quadrupedal walking with multi-domain hybrid system models and virtual constraint controllers". In American Control Conference (ACC), pp. 4588–4595.

[41] Cao, Q., and Poulakakis, I., 2016. "Quadrupedal running with a flexible torso: control and speed transitions with sums-of-squares verification". *Artificial Life and Robotics,* **21**(4), Dec, pp. 384–392.

[42] Akbari Hamed, K., Kim, J., and Pandala, A., 2020. "Quadrupedal locomotion via event-based predictive control and QP-based virtual constraints". *IEEE Robotics and Automation Letters,* **5**(3), pp. 4463–4470.

[43] Gregg, R., and Sensinger, J., 2014. "Towards biomimetic virtual constraint control of a powered prosthetic leg". *IEEE Transactions on Control Systems Technology,* **22**(1), Jan, pp. 246–254.

[44] Zhao, H., Horn, J., Reher, J., Paredes, V., and Ames, A. D., 2016. "First steps toward translating robotic walking to prostheses: a nonlinear optimization based control approach". *Autonomous Robots*, pp. 1–18.

[45] Martin, A. E., and Gregg, R. D., 2017. "Stable, robust hybrid zero dynamics control of powered lower-limb prostheses". *IEEE Transactions on Automatic Control,* **62**(8), pp. 3930–3942.

[46] Agrawal, A., Harib, O., Hereid, A., Finet, S., Masselin, M., Praly, L., Ames, A., Sreenath, K., and Grizzle, J., 2017. "First steps towards translating HZD control of bipedal robots to decentralized control of exoskeletons". *IEEE Access,* **5**, pp. 9919–9934.

[47] Hereid, A., and Ames, A. D., 2017. "FROST: Fast robot optimization and simulation toolkit". In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 719–726.

[48] Griffin, R. J., Wiedebach, G., Bertrand, S., Leonessa, A., and Pratt, J., 2017. "Walking stabilization using step timing and location adjustment on the humanoid robot, Atlas". In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 667–673.

[49] Pratt, J., Carff, J., Drakunov, S., and Goswami, A., 2006. "Capture point: A step toward humanoid push recovery". In 2006 6th IEEE-RAS International Conference on Humanoid Robots, pp. 200–207.

[50] Englsberger, J., Ott, C., Roa, M. A., Albu-Schäffer, A., and Hirzinger, G., 2011. "Bipedal walking control based on capture point dynamics". In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4420–4427.

[51] Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., and Hirukawa, H., 2003. "Biped walking pattern generation by using preview control of zero-moment point". In IEEE International Conference on Robotics and Automation, Vol. 2, pp. 1620–1626 vol.2.

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

[53] Ding, Y., Pandala, A., and Park, H., 2019. "Real-time model predictive control for versatile dynamic motions in quadrupedal robots". In International Conference on Robotics and Automation, pp. 8484–8490.

[54] Neunert, M., Stäuble, M., Giftthaler, M., Bellicoso, C. D., Carius, J., Gehring, C., Hutter, M., and Buchli, J., 2018. "Whole-body nonlinear model predictive control through contacts for quadrupeds". *IEEE Robotics*

and Automation Letters, **3**(3), July, pp. 1458–1465.

[55] Bledt, G., Wensing, P. M., and Kim, S., 2017. "Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the mit cheetah". In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4102–4109.

[56] Fahmi, S., Mastalli, C., Focchi, M., and Semini, C., 2019. "Passive whole-body control for quadruped robots: Experimental validation over challenging terrain". *IEEE Robotics and Automation Letters,* **4**(3), July, pp. 2553–2560.

[57] Kuindersma, S., Permenter, F., and Tedrake, R., 2014. "An efficiently solvable quadratic program for stabilizing dynamic locomotion". In IEEE International Conference on Robotics and Automation, pp. 2589–2594.

[58] Winkler, A. W., Farshidian, F., Pardo, D., Neunert, M., and Buchli, J., 2017. "Fast trajectory optimization for legged robots using vertex-based ZMP constraints". *IEEE Robotics and Automation Letters,* **2**(4), Oct, pp. 2201–2208.

[59] Lehmann, D., Henriksson, E., and Johansson, K. H., 2013. "Event-triggered model predictive control of discrete-time linear systems subject to disturbances". In 2013 European Control Conference (ECC), pp. 1156–1161.

[60] Libby, T., Moore, T. Y., Chang-Siu, E., Li, D., Cohen, D. J., Jusufi, A., and Full, R. J., 2012. "Tail-assisted pitch control in lizards, robots and dinosaurs". *Nature*, pp. 2–7.

[61] Jusufi, A., Kawano, D. T., Libby, T., and Full, R. J., 2010. "Righting and turning in mid-air using appendage inertia : reptile tails , analytical models and bio-inspired robots". *Bioinspiration and Biomimetics*.

[62] Attenborough, D., 2002 - 2003. The life of Mammals: the Complete Series - Episode 5: meat Eaters. BBC Documentary.

[63] Brill, A. L., De, A., Johnson, A. M., and Koditschek, D. E., 2015. "Tail-assisted rigid and compliant legged leaping". In IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 6304–6311.

[64] Zhao, J., Zhao, T., Xi, N., Cintrón, F. J., Mutka, M. W., and Xiao, L., 2013. "Controlling aerial maneuvering of a miniature jumping robot using its tail". In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 3802–3807.

[65] De, A., and Koditschek, D. E., 2015. "Parallel composition of templates for tail-energized planar hopping". In IEEE International Conference on Robotics and Automation, IEEE, pp. 4562–4569.

[66] Wenger, G., De, A., and Koditschek, D. E., 2016. "Frontal plane stabilization and hopping with a 2dof tail". In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 567–573.

[67] Briggs, R., Lee, J., Haberland, M., and Kim, S., 2012. "Tails in biomimetic design: Analysis, simulation, and experiment". In IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 1473–1480.

[68] Patel, A., and Braae, M., 2015. "An actuated tail increases rapid acceleration manoeuvres in quadruped robots". In *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*. Springer, pp. 69–76.

[69] Machairas, K., and Papadopoulos, E., 2015. "On quadruped attitude dynamics and control using reaction wheels and tails". In European Control Conference, pp. 753–758.

[70] Libby, T., Johnson, A. M., Chang-Siu, E., Full, R. J., and Koditschek, D. E., 2016. "Comparative design, scaling, and control of appendages for inertial reorientation". *IEEE Transactions on Robotics,* **32**(6), pp. 1380–1398.

[71] Rone, W. S., Saab, W., Kumar, A., and Ben-Tzvi, P., 2019. "Controller design, analysis, and experimental validation of a robotic serpentine tail to maneuver and stabilize a quadrupedal robot". *ASME Journal of Dynamic Systems, Measurement, and Control,* **141**(8), 03, pp. 081002–1–081002–9.

[72] Heim, S. W., Ajallooeian, M., Eckert, P., Vespignani, M., and Ijspeert, A. J., 2016. "On designing an active tail for legged robots: simplifying control via decoupling of control objectives". *Industrial Robot: An International Journal,* **43(3)**, pp. 338–346.

[73] Zhang, X., Gong, J., and Yao, Y., 2016. "Effects of head and tail as swinging appendages on the dynamic walking performance of a quadruped robot". *Robotica,* **34**(12), pp. 2878–2891.

[74] Mishima, Y., and Ozawa, R., 2014. "Design of a robotic finger using series gear chain mechanisms". *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems,* **1**(IROS), pp. 2898–2903.

[75] Domahidi, A., Chu, E., and Boyd, S., 2013. "ECOS: An SOCP solver for embedded systems". In European Control Conference, pp. 3071–3076.

[76] Hwangbo, J., Lee, J., and Hutter, M., 2018. "Per-contact iteration method for solving contact dynamics". *IEEE Robotics and Automation Letters,* **3**(2), April, pp. 895–902.

[77] Pandala, A. G., Ding, Y., and Park, H., 2019. "qpSWIFT: A real-time sparse quadratic program solver for robotic applications". *IEEE Robotics and Automation Letters,* **4**(4), Oct, pp. 3355–3362.

[78] Felis, M. L., 2016. "RBDL: an efficient rigid-body dynamics library using recursive algorithms". *Autonomous Robots*, pp. 1–17.

[79] De Wit, C., Olsson, H., Astrom, K., and Lischinsky, P., 1995. "A new model for control of systems with friction". *IEEE Transactions on Automatic Control,* **40**(3), Mar, pp. 419–425.

[80] Vanderborght, B., Verrelst, B., Van Ham, R., Van Damme, M., and Lefeber, D., 2008. "Objective locomotion parameters based inverted pendulum trajectory generator". *Robotics and Autonomous systems,* **56**(9), pp. 738–750.

List of Figures