Planar Bipedal Locomotion with Nonlinear Model Predictive Control: Online Gait Generation using Whole-Body Dynamics

Manuel Y. Galliker[1†], Noel Csomay-Shanklin[2†], Ruben Grandia[1], Andrew J. Taylor[2],
Farbod Farshidian[1], Marco Hutter[1], Aaron D. Ames[2]

Abstract— The ability to generate dynamic walking in real-time for bipedal robots with input constraints and underactuation has the potential to enable locomotion in dynamic, complex and unstructured environments. Yet, the high-dimensional nature of bipedal robots has limited the use of full-order rigid body dynamics to gaits which are synthesized offline and then tracked online. In this work we develop an online nonlinear model predictive control approach that leverages the full-order dynamics to realize diverse walking behaviors. Additionally, this approach can be coupled with gaits synthesized offline via a desired reference to enable a shorter prediction horizon and rapid online re-planning, bridging the gap between online reactive control and offline gait planning. We demonstrate the proposed method, both with and without an offline gait, on the planar robot AMBER-3M in simulation and on hardware.

I. INTRODUCTION

From complex terrains inaccessible by wheels to human-centered infrastructure impractical for quadrupeds, bipedal robots hold the potential to operate in diverse environments in which other robots struggle. To achieve this potential, it is necessary to demonstrate a rich set of locomotion behaviors that are dynamically stable. Bipedal robots capable of demonstrating diverse behaviors, much like their human counterparts, leverage phases of underactuation. This underactuation necessitates the dynamic coordination of the whole-body dynamics of the robot—planning for the next foot strike must occur throughout the step—in a manner that accounts for the inherently nonlinear passive dynamics of the system. Achieving diverse locomotion behaviors in complex environments, therefore, motivates that whole-body planning be done on the robot in real-time, thereby going beyond pre-planned periodic walking gaits.

The challenge of underactuation present in bipedal locomotion has historically been approached through the synthesis of gaits, i.e. dynamically stable reference trajectories. Many of these approaches for gait synthesis use condensed stability conditions like Zero-Moment Point (ZMP) [1], or
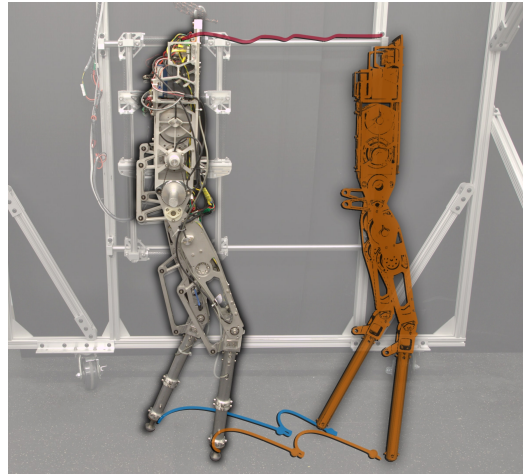

Fig. 1: AMBER-3M platform using the whole-body nonlinear MPC incorporating an HZD gait. The optimized feet and torso trajectories are visualized along the prediction horizon.

rely on other reduced-order models that simplify elements such as leg mass [2]–[4]. Alternatively, the method of Hybrid Zero Dynamics (HZD) has presented a tool for synthesizing *periodic gaits* that account for the underactuated and hybrid nature of the full system dynamics [5]. Not only do gaits synthesized via HZD possess formal stability guarantees, but they have shown great efficacy when deployed experimentally [6], [7]. Despite these successes, ensuring stability guarantees for high-dimensional bipedal systems often induces computational requirements that limit gait synthesis via HZD to an offline procedure. Adding a measure of flexibility to bipedal locomotion is often done by synthesizing a library of gaits [8]–[10], although this is limited to periodic gaits, requires complex engineering solutions for gait transitions and ensuring constraint satisfaction during execution.

In contrast, Model Predictive Control (MPC) provides a tool for the online synthesis of general, *aperiodic trajectories*, allowing feedback of environmental parameters to be incorporated into dynamic motion planning [11], [12]. In particular, by optimizing directly over contact forces, these methods have seen significant use in online motion planning for quadrupedal robotics, with extensive experimental results [13]–[15]. Online motion planning for bipedal robots has typically required ZMP conditions [16]–[18], used simplified template models for planning and whole-body tracking controllers [19]–[22], or used whole-body planning only for non-walking tasks such as reaching [23]. Whole-body motion planning results for bipedal walking have been predomi-
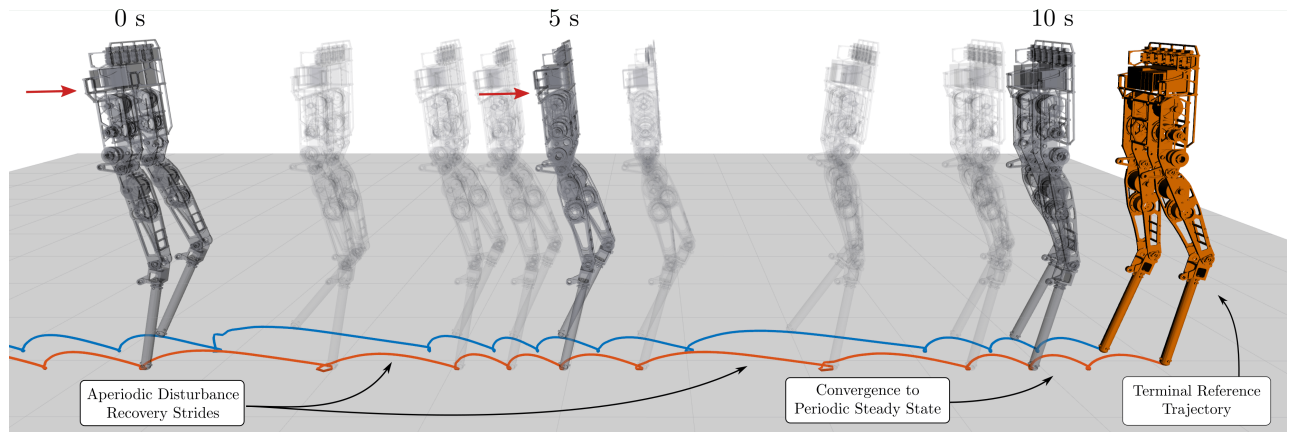
Fig. 2: Push recovery using the proposed method under a disturbance – notice the aperiodic stepping that was planned online in order to reject the disturbance, something that is not possible with traditional HZD based methods.

nantly in simulation [24]–[27], or cancelled nonlinear dynamics through feedback-linearization before planning [28]. Notably, the online motion planning tools with remarkable experimental results for quadrupedal locomotion have not yet achieved commensurate results for bipedal robotics.

One of the key challenges in online whole-body motion planning is computational limitations, as producing stable locomotion requires optimizing over a sufficiently long horizon. The methods for quadrupeds that have yielded experimental results typically exploit low leg inertia to neglect leg dynamics, reducing the state dimension in the optimization. Transferring this reduction to bipedal systems is difficult, however, as the legs compose a relatively high fraction of the system's total inertia. At the same time, simultaneously considering both leg and torso dynamics results in many degrees of freedom, making optimization over long time horizons computationally intensive. Furthermore, the narrow stance width and high center of mass of bipeds necessitate a high planning frequency to counteract disturbances in under-actuated dimensions. Thus, it is paramount to design whole-body motion planners that balance the trade-off between horizon length, model complexity, and planning frequency.

We make three contributions in this work. First, we propose a nonlinear MPC approach for online whole-body motion planning of bipedal robotic locomotion based on existing methods used for quadrupedal locomotion [29], which achieved a wide range of stable behaviors with a planning horizon of $2\,\mathrm{s}$ and update frequency of up to $270\,\mathrm{Hz}$. Second, to reduce the computational burden of online whole-body planning, we incorporate a stable walking gait synthesized offline via Hybrid Zero Dynamics (HZD) into the nonlinear optimization problem. This information permits robust walking while optimizing over shorter horizon lengths $(0.2\,\mathrm{s})$ that require less computational effort and allow rapid re-planning $(850\,\mathrm{Hz})$ – this will be important for achieving whole-body planning on 3D walking systems. Lastly, we experimentally validate the proposed approach on the planar bipedal robot AMBER-3M [30], demonstrating standing, stepping in place, and walking. To the best of our knowledge, this is the first experimental demonstration of online whole-body motion planning for bipedal walking.

II. BACKGROUND

A. System Dynamics

As walking consists of phases of intermittent contact with the world, it is naturally modeled as a hybrid system with of phases of continuous dynamics followed by discrete transition events. The configuration of the robot may be described by a set of d generalized coordinates:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_b^\top & \mathbf{q}_j^\top \end{bmatrix}^\top \in \mathcal{Q} \triangleq SE(3) \times \mathcal{Q}_j, \qquad (1)$$

which include the base coordinates $\mathbf{q}_b$ and joint coordinates $\mathbf{q}_j$ of the robot, respectively. To capture the various contact modes the robot may evolve under, consider a collection of domains $\{\mathcal{D}_c\}$ with $\mathcal{D}_c \subseteq \mathcal{X}$ for $c = 1, \ldots, p$, where $\mathcal{X}$ is the tangent bundle of $\mathcal{Q}$ and p denotes the number of contact modes. Associated with these domains are a collection of guards $\{\mathcal{S}_c\}$ with $\mathcal{S}_c \subseteq \mathcal{X}$ and reset maps $\{\Delta_c\}$ with $\Delta_c : \mathcal{X} \to \mathcal{X}$ which are used to define how the system behaves during transitions between contact modes. Additionally, in each domain $\mathcal{D}_c$ the coordinates of the robot are subject to a collection of n_c *holonomic constraints* with associated contact Jacobians $\mathbf{J}_c : \mathcal{Q} \to \mathbb{R}^{6n_c \times d}$.

Using the Euler-Lagrange method, the system dynamics in a given domain $\mathcal{D}_c$ are given by:

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}(\mathbf{q})\boldsymbol{\tau} + \mathbf{J}_c(\mathbf{q})^\top \boldsymbol{\lambda}, \qquad (2)$$

$$\mathbf{J}_c(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{0}, \qquad (3)$$

with symmetric positive definite inertia matrix $\mathbf{D} : \mathcal{Q} \to \mathbb{S}_{\succ 0}^d$, centrifugal, Coriolis, and gravitational terms $\mathbf{h} : \mathcal{X} \to \mathbb{R}^d$, actuation matrix $\mathbf{B} : \mathcal{Q} \to \mathbb{R}^{d \times m}$, torques $\boldsymbol{\tau} \in \mathbb{R}^m$, and constraint forces $\boldsymbol{\lambda} \in \mathbb{R}^{6n_c}$. Defining the state $\mathbf{x} \in \mathcal{X}$ as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q}^\top & \dot{\mathbf{q}}^\top \end{bmatrix}^\top, \qquad (4)$$

and solving for the constraint forces $\boldsymbol{\lambda}$ via (2)-(3), the system dynamics in a given domain $\mathcal{D}_c$ can be rewritten as:

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{D}^{-1}(-\mathbf{h} + \mathbf{J}_c^\top \boldsymbol{\lambda}) \end{bmatrix}}_{\mathbf{f}_c(\mathbf{x})} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{D}^{-1}\mathbf{B} \end{bmatrix}}_{\mathbf{g}(\mathbf{x})} \boldsymbol{\tau}, \qquad (5)$$

where the dependence on $\mathbf{q}$ and $\dot{\mathbf{q}}$ has been dropped for notational simplicity. The resulting functions $\mathbf{f}_c : \mathcal{X} \to \mathbb{R}^{2d}$ and $\mathbf{g} : \mathcal{X} \to \mathbb{R}^{2d \times m}$ are assumed to be continuously differentiable on the domain $\mathcal{D}_c$.

To model a transition from contact mode c to contact mode c', consider a state $\mathbf{x}^- \triangleq (\mathbf{q}^-, \dot{\mathbf{q}}^-) \in \mathcal{S}_c$. The discrete transition map is given by:

$$\mathbf{D}(\mathbf{q}^-)(\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}^-) = \mathbf{J}_{c'}^\top(\mathbf{q}^-)\mathbf{F}, \qquad (6)$$

$$\mathbf{J}_{c'}(\mathbf{q}^-)\dot{\mathbf{q}}^+ = \mathbf{0}, \qquad (7)$$

whereby solving for the impulse force $\mathbf{F} \in \mathbb{R}^{n_c}$ yields:

$$\mathbf{x}^+ \triangleq \begin{bmatrix} \mathbf{q}^+ \\ \dot{\mathbf{q}}^+ \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{q}^- \\ \dot{\mathbf{q}}^- + \mathbf{D}(\mathbf{q}^-)^{-1}\mathbf{J}_{c'}^\top(\mathbf{q}^-)\mathbf{F} \end{bmatrix}}_{\Delta_c(\mathbf{x}^-)}. \qquad (8)$$

B. Nonlinear Model Predictive Control

Nonlinear MPC solves a optimization problem in a receding horizon manner by solving the following finite time nonlinear optimal control problem:

$$\underset{\mathbf{u}(\cdot)}{\text{minimize}} \quad \phi(\mathbf{x}(t_H)) + \int_0^{t_H} l(\mathbf{x}(t), \mathbf{u}(t), t)\mathrm{d}t, \qquad (9a)$$

$$\text{subject to:} \quad \mathbf{x}(0) = \mathbf{x}_0, \qquad (9b)$$

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \qquad (9c)$$

$$\mathbf{x}(t_i^+) = \Delta_c(\mathbf{x}(t_i)), \qquad (9d)$$

$$\mathbf{h}_{eq}(\mathbf{x}, \mathbf{u}, t) = \mathbf{0}, \qquad (9e)$$

$$\mathbf{h}_{in}(\mathbf{x}, \mathbf{u}, t) \geq \mathbf{0}, \qquad (9f)$$

where t_H is the length of the horizon, $\phi : \mathcal{X} \to \mathbb{R}$ is the terminal cost, $l : \mathcal{X} \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ is the time-varying running state-input cost, and t_i are times of contact mode transitions. The optimal control problem is solved in real-time by updating the initial conditions (9b) with the measured state of the system. Eq. (9c) describes the system dynamics. $\mathbf{h}_{eq} : \mathcal{X} \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{eq}$ and $\mathbf{h}_{in} : \mathcal{X} \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{in}$ are generalized path equality and inequality constraints, respectively. There exist various approaches to solve this problem as outlined in [31]. We take a direct-multiple shooting transcription of the problem together with a sequential quadratic programming approach to handle nonlinearities [32]. Inequality constraints (9f) are implemented through relaxed-barrier penalty functions [33].

A key component in establishing closed loop stability and recursive feasibility is the choice of terminal components [34], either as terminal cost in (9a) or as constraints on the terminal state, $\mathbf{x}(t_H)$, to lie in a control invariant set. In practice, for nonlinear complex systems, it is challenging to prove that such conditions hold. Extending the prediction horizon is a common choice to reduce the relative importance of the terminal components [35]. However, for systems where long prediction horizons are not feasible due to computational limits, careful choice of terminal components directly translates to the overall performance of the controller, as we will empirically show in this work.

C. Hybrid Zero Dynamics (HZD)

The HZD framework provides a formal method for producing walking behaviors for robotic systems, and has been successfully employed on a variety of platforms [5]–[7]. Synthesis of gaits via HZD is centered around defining *outputs* $\mathbf{y} : \mathbb{R}^d \times \mathbb{R}^r \to \mathbb{R}^o$ as:

$$\mathbf{y}(\mathbf{q}, \boldsymbol{\alpha}) = \mathbf{y}_a(\mathbf{q}) - \mathbf{y}_d(\mathbf{q}, \boldsymbol{\alpha}), \qquad (10)$$

where $\mathbf{y}_a : \mathbb{R}^d \to \mathbb{R}^o$ and $\mathbf{y}_d : \mathbb{R}^d \times \mathbb{R}^r \to \mathbb{R}^o$ are the *actual* and *desired* outputs, respectively, and are assumed to be continuously differentiable. The actual outputs $\mathbf{y}_a$ are chosen to satisfy a controllability property allowing them to be driven to the desired outputs [5]. The desired outputs $\mathbf{y}_d$ depend on the set of parameters $\boldsymbol{\alpha} \in \mathbb{R}^r$, which are chosen to regulate the underactuated coordinates of the system. More precisely, the *zero dynamics manifold* is defined as the subspace of state coordinates for which the outputs and their derivatives are zero:

$$\mathcal{Z}_c = \{(\mathbf{q}, \dot{\mathbf{q}}) \in \mathcal{X} : \mathbf{y}(\mathbf{q}, \boldsymbol{\alpha}) = \dot{\mathbf{y}}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\alpha}) = \mathbf{0}\}. \qquad (11)$$

The parameters are then chosen to satisfy the *hybrid invariance condition*, $\Delta_c(\mathcal{Z}_c \cap \mathcal{S}_c) \subset \mathcal{Z}_c$, for each contact mode c, which ensures that the underactuated coordinates of the system remain stable through impacts. Ensuring this condition is achieved by finding values of $\boldsymbol{\alpha}$ through nonlinear optimization [36]. Given a desired trajectory $\mathbf{y}_d$ from an HZD optimization program, a time-varying state and input trajectory $(\mathbf{x}_{\text{ref}}(t), \mathbf{u}_{\text{ref}}(t))$ can be reconstructed. Importantly, as this trajectory is designed for the full-order hybrid dynamics, it will serve as a control invariant that will be incorporated in our MPC formulation.

III. Whole-Body Motion Planning & Control

Our nonlinear MPC problem will be constructed using the *OCS2* toolbox [37], which provides convenient interfaces to the *Pinocchio* [38] rigid body library and *CppAd* [39] automatic differentiation tools. Our formulation assumes that the contact schedule associated with a given locomotion mode (standing, stepping in place, walking) is given by the user. The fixed contact schedule assumption simplifies the optimization problem as the sequence of domains and timing of contact mode transitions does not need to be optimized [11], [29]. The position of the foot at contact is captured in the optimization problem through its kinematic relationship with joint coordinates. Moreover, we assume the user provides a desired base pose and velocity to the MPC. In this section we discuss the formulation of bipedal locomotion planning as an MPC problem as posed in (9).

A. System Dynamics

Due to the affine relationship between generalized accelerations $\ddot{\mathbf{q}}$, torques $\boldsymbol{\tau}$, and contact forces $\boldsymbol{\lambda}$ in (2), and assuming the torques do not directly impact the floating-base equations of motion, the system dynamics in (5) may be rewritten to interpret the joint accelerations $\ddot{\mathbf{q}}_j$ and contact forces $\boldsymbol{\lambda}$, instead of the torques $\boldsymbol{\tau}$, as inputs. The computational benefit
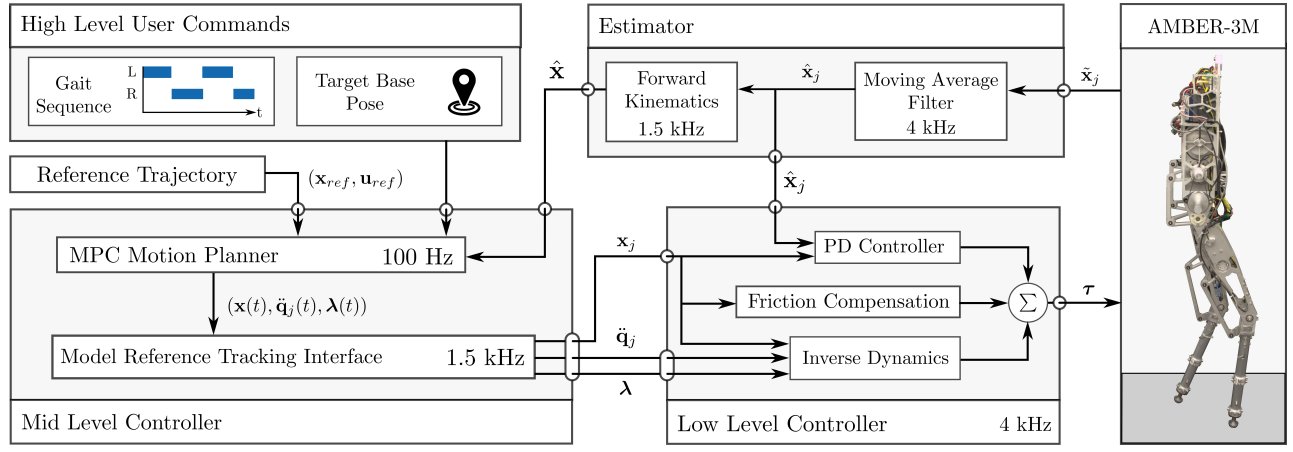
Fig. 3: Multi-rate control architecture incorporating whole-body planning via MPC and low-level tracking controller.

of this reparametrization has been shown for reactive whole-body control [40] and offline trajectory optimization [41]. To see this, we write the dynamics (2) in terms of non-actuated base coordinates and fully actuated joint coordinates:

$$\begin{bmatrix} \mathbf{D}_{bb}, & \mathbf{D}_{bj} \\ \mathbf{D}_{bj}^\top & \mathbf{D}_{jj} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_b \\ \ddot{\mathbf{q}}_j \end{bmatrix} + \begin{bmatrix} \mathbf{h}_b \\ \mathbf{h}_j \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_j \end{bmatrix} \boldsymbol{\tau} + \begin{bmatrix} \mathbf{J}_{c,b}^\top \\ \mathbf{J}_{c,j}^\top \end{bmatrix} \boldsymbol{\lambda}. \quad (12)$$

The base acceleration may be expressed as:

$$\ddot{\mathbf{q}}_b = -\mathbf{D}_{bb}^{-1} \left(\mathbf{h}_b + \begin{bmatrix} \mathbf{D}_{bj} & -\mathbf{J}_{c,b}^\top \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_j \\ \boldsymbol{\lambda} \end{bmatrix} \right), \quad (13)$$

and assuming the legs are fully actuated ($\mathbf{B}_j$ is invertible), the corresponding joint torques may be expressed as:

$$\boldsymbol{\tau} = \mathbf{B}_j^{-1} \left(\mathbf{D}_{bj}^\top \ddot{\mathbf{q}}_b + \mathbf{h}_j + \begin{bmatrix} \mathbf{D}_{jj} & -\mathbf{J}_{c,j}^\top \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_j \\ \boldsymbol{\lambda} \end{bmatrix} \right), \quad (14)$$

maintaining an affine dependence on $\ddot{\mathbf{q}}_j$ and $\boldsymbol{\lambda}$. The base dynamics in (13) fully encode the challenge of under-actuation and encapsulate the core of the floating-base dynamics. Equation (14) plays a secondary role and is only required when formulating torques constraints. We may view the control inputs to optimize over as:

$$\mathbf{u} = \begin{bmatrix} \ddot{\mathbf{q}}_j^\top, & \boldsymbol{\lambda}^\top \end{bmatrix}^\top, \quad (15)$$

with the corresponding system dynamics defined as:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{D}_{bb}^{-1} \left(-\mathbf{h}_b - \mathbf{D}_{bj} \ddot{\mathbf{q}}_j + \mathbf{J}_{c,b}^\top \boldsymbol{\lambda} \right) \\ \ddot{\mathbf{q}}_j \end{bmatrix}. \quad (16)$$

In order to avoid large discontinuities in the optimized trajectory, the contact transition maps in (9d) have been set to identity maps for the online MPC program, with exponential damping of the contact point velocity after impact being regulated through the stance foot constraint in (20), defined in section III-D. Inclusion of the impact dynamics in (8) will be pursued in future work. Due to the assumption of a fixed contact schedule, inclusion of the contact transition map does not fundamentally change the complexity of the optimization problem [11].

B. Cost Functions

The cost function is formulated as a nonlinear least square cost around a given state and input reference trajectory. To that end we define the set of tracking errors as follows:

$$\boldsymbol{\varepsilon}_x = \mathbf{x} - \mathbf{x}_{\mathrm{ref}}(t), \; \boldsymbol{\varepsilon}_u = \mathbf{u} - \mathbf{u}_{\mathrm{ref}}(t), \; \boldsymbol{\varepsilon}_i = \begin{bmatrix} \mathbf{p}_i - \mathbf{p}_{i,\mathrm{ref}}(t) \\ \mathbf{v}_i - \mathbf{v}_{i,\mathrm{ref}}(t) \\ \mathbf{a}_i - \mathbf{a}_{i,\mathrm{ref}}(t), \end{bmatrix},$$

where $\mathbf{x}_{\mathrm{ref}}$ is the state reference, $\mathbf{u}_{\mathrm{ref}}$ is the input reference, and $\mathbf{p}_i, \mathbf{v}_i, \mathbf{a}_i \in \mathbb{R}^3$ with $i \in \{1, 2\}$ are the Cartesian position, velocity, and accelerations of the ith foot, with corresponding references $\mathbf{p}_{i,\mathrm{ref}}, \mathbf{v}_{i,\mathrm{ref}}, \mathbf{a}_{i,\mathrm{ref}}$. The references $\mathbf{x}_{\mathrm{ref}}$ and $\mathbf{u}_{\mathrm{ref}}$ are defined heuristically (see Section III-C) or via a walking gait synthesized offline using HZD. The running state-input cost l is given by:

$$l(\mathbf{x}, \mathbf{u}, t) = \frac{1}{2} \boldsymbol{\varepsilon}_x^\top \mathbf{Q} \boldsymbol{\varepsilon}_x + \frac{1}{2} \boldsymbol{\varepsilon}_u^\top \mathbf{R} \boldsymbol{\varepsilon}_u + \frac{1}{2} \sum_i \boldsymbol{\varepsilon}_i^\top \mathbf{W} \boldsymbol{\varepsilon}_i, \quad (17)$$

where $\mathbf{Q}, \mathbf{R}$, and $\mathbf{W}$ are positive definite weighting matrices.

To pick an appropriate weighting for the terminal cost, we approximate the infinite horizon cost by solving an unconstrained Linear Quadratic Regulator (LQR) problem using a linear approximation of the dynamics and a quadratic approximation of the running costs (17) around the nominal stance configuration of the robot. The positive definite Riccati matrix $\mathbf{S}_{\mathrm{LQR}}$ of the cost-to-go is used to define the quadratic cost around the terminal reference state:

$$\phi(\mathbf{x}) = \frac{\rho}{2} \boldsymbol{\varepsilon}_x(T)^\top \mathbf{S}_{\mathrm{LQR}} \boldsymbol{\varepsilon}_x(T), \quad (18)$$

where $\rho > 0$ is a hyperparameter. Setting $\rho = 1.0$ would express approximately equal importance of the integrated running cost and terminal cost, and $\rho \to \infty$ would make the terminal cost behave as an equality constraint. We found good performance for the heuristic reference at $\rho = 1.0$ and for the HZD reference at $\rho = 10.0$. Note that this cost does not penalize deviation from the stance configuration used to produce the LQR solution, but rather provides a systematic way to scale the relative importance of all of the degrees of freedom of the robot in the cost.

C. Reference Trajectories

a) HZD Trajectory: HZD state and input reference trajectories, $\mathbf{x}_{\mathrm{ref}}(t)$ and $\mathbf{u}_{\mathrm{ref}}(t)$, are found offline for the whole-body nonlinear dynamics using the FROST toolbox [36] and stored as Bézier polynomials. This process is completed by fixing a target gait sequence and a forward velocity, and adding various other state and input constraints to a nonlinear trajectory optimization program which ensure the underactuated dynamics of the system display stable periodic behavior. For planar systems, stability can be enforced directly in the optimization program [42], and for general systems it can be verified *a-posteriori* via the Poincaré return map [5]. The foot references $\mathbf{p}_{i,\mathrm{ref}}(t)$, $\mathbf{v}_{i,\mathrm{ref}}(t)$, and $\mathbf{a}_{i,\mathrm{ref}}(t)$ are entirely determined by $\mathbf{x}_{\mathrm{ref}}(t)$.

b) Heuristic Trajectory: To evaluate the relative impact of using a gait synthesized offline via HZD in the cost function, we produce a heuristic reference trajectory to be compared against. In particular, the state trajectory $\mathbf{x}_{\mathrm{ref}}(t)$ is composed of a user-commanded base pose and velocity, and a static nominal joint configuration. The input reference $\mathbf{u}_{\mathrm{ref}}(t)$ is defined with zero joint accelerations and contact forces that are evenly distributed among each foot in contact in the nominal joint configuration such that the weight of the robot is compensated. The foot references $\mathbf{p}_{i,\mathrm{ref}}(t)$, $\mathbf{v}_{i,\mathrm{ref}}(t)$, and $\mathbf{a}_{i,\mathrm{ref}}(t)$ are designed by extracting the nominal touchdown and liftoff locations below the hip at the middle of the contact phase and fitting a smooth hand-designed swing reference trajectory. The heuristic and HZD-based terminal state are visualized in Fig. 4.

D. Constraints

The following constraints are imposed in problem (9). All inequality constraints are implemented as soft constraints with relaxed log barrier functions [37].

a) Gait-Dependant Constraints: These constraints capture the different modes of each leg at any given point in time determined by the specified gait sequence. We enforce the user-defined gait and avoid foot scuffing of a swing leg by constraining the swing foot motion in the orthogonal direction to the ground surface, $\mathbf{n} \in \mathbb{R}^3$, to follow the Cartesian reference trajectory:

$$\mathbf{n}^\top(\mathbf{a}_i - \mathbf{a}_{i,\mathrm{ref}}(t) + k_d(\mathbf{v}_i - \mathbf{v}_{i,\mathrm{ref}}(t)) \\ + k_p(\mathbf{p}_i - \mathbf{p}_{i,\mathrm{ref}}(t))) = \mathbf{0} \quad (19)$$

where $k_d, k_p \in \mathbb{R}_{\geq 0}$ are feedback gains chosen to achieve asymptotic tracking in the constrained space. The foot position in the direction parallel to the ground is not directly constrained; rather, tracking is enforced via the cost function described in Section III-B. For a stance leg we enforce a stationarity constraint in Cartesian space through:

$$\mathbf{a}_i + k_d\mathbf{v}_i = \mathbf{0} \quad (20)$$

b) Contact Force Constraints: The following constraints require the contact forces at each foot to match the designation of swing and stance legs:

$$\begin{cases} \boldsymbol{\lambda}_i = \mathbf{0}, & i \text{ is a swing leg} \\ \boldsymbol{\lambda}_i \in \mathcal{C}(\mathbf{n}, \mu_c), & i \text{ is a stance leg.} \end{cases} \quad (21)$$
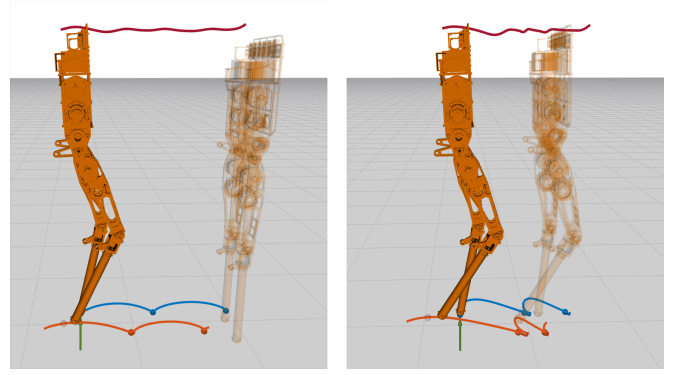

Fig. 4: Heuristic (left) and HZD (right) terminal states.

The first constraint requires no contact force from a swing leg, as it does not contact the ground. The second constraint requires the contact force of a stance leg to lie in the friction cone $\mathcal{C}(\mathbf{n}, \mu_c)$ defined by the surface normal $\mathbf{n}$ and the friction coefficient $\mu_c = 0.6$. This is a second-order cone constraint and is expressed in the local surface aligned frame:

$$\mu_c\lambda_{i,3} - \sqrt{\lambda_{i,1}^2 + \lambda_{i,2}^2} \geq 0. \quad (22)$$

Note that only linear forces are present as the robot has point feet, and that the friction constraint also enforces a unilateral contact constraint as it requires $\lambda_{i,3} \geq 0$.

c) Joint and Torque Limits: The joint coordinates and joint coordinate velocities are enforced to lie in the set of minimum and maximum joint positions and velocities through state inequality constraints: $\mathbf{x} \in [\mathbf{x}_{\min}, \mathbf{x}_{\max}]$. Similarly, the joint torques can be computed by Eq. (14) and should lie within joint torque limits $\boldsymbol{\tau} \in [\boldsymbol{\tau}_{\min}, \boldsymbol{\tau}_{\max}]$.

E. Low-Level Controller

As shown in Fig. 3, the state and input trajectories generated by MPC are interpolated at a high frequency and converted to a feed-forward control torques, $\boldsymbol{\tau}_{\mathrm{MPC}}$, via (14). As the feed-forward torque is model-based, we compensate for model errors when executing the controller on hardware by adding a proportional-derivative torque, $\boldsymbol{\tau}_{\mathrm{PD}}$, and a friction compensation torque, $\boldsymbol{\tau}_{\mathrm{FC}}$, to the feed-forward torque:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\mathrm{MPC}} + \boldsymbol{\tau}_{\mathrm{PD}} + \boldsymbol{\tau}_{\mathrm{FC}}. \quad (23)$$

The total torque $\boldsymbol{\tau}$ is send to the open loop torque controlled BLDC motors.

IV. AMBER IMPLEMENTATION & RESULTS

The AMBER-3M platform is a 5-link planar bipedal robot, which has four open loop torque controlled BLDC motors connected via harmonic drives to the hip and knee joints. The total mass of the robot amounts to $21.6\,\mathrm{kg}$, approximately $40\,\%$ of which is located in the legs. The joint coordinates are given by $\mathbf{q}_j \in \mathcal{Q}_j \subset \mathbb{R}^4$, and, due to the planar nature of the robot, the base coordinates are given by $\mathbf{q}_b = \begin{bmatrix} x_b, z_b, \theta_b \end{bmatrix}^\top \in SE(2)$ resulting in a state vector $\mathbf{x} \in \mathbb{R}^{14}$. The input $\mathbf{u} = (\ddot{\mathbf{q}}_j, \boldsymbol{\lambda}) \in \mathbb{R}^8$ contains the joint accelerations and 2D Cartesian contact forces at the point-feet. Position and velocity measurements of the

TABLE I: MPC Planning Frequency (10 SQP Iterations)

Horizon Length [s]	2.0	1.0	0.5	0.2
MPC Frequency [Hz]	270	480	670	850

four joint angles and the base angle are present and used to estimate the linear base position and velocity. No impact measurement was used during the experiments. The time discretization in the multiple shooting scheme is set to $15\,\mathrm{ms}$ and we allow for a maximum of 10 SQP iterations per MPC problem. All planning, control, and estimation loops were done on separate threads on an offboard Ryzen 9 5950x CPU @ $3.4\,\mathrm{GHz}$. Benchmarks of the maximum obtainable MPC frequency for different horizon lengths can be seen in Table I. To isolate how the system's behavior depends on horizon length, all experiments were conducted with a consistent MPC frequency of $100\,\mathrm{Hz}$.

As can be seen in the supplementary video [43], the proposed MPC formulation is capable of simultaneously stabilizing the underactuated system dynamics and synthesizing valid motion trajectories for a broad range of gait pattern and target velocities both in simulation and on hardware. To evaluate the effect of changing reference signals on the feasibility and robustness of the full control pipeline, a sequence of disturbances of increasing magnitude was applied in simulation with the following MPC configurations:

- *MPC with No Terminal*: The proposed whole-body MPC with heuristic references for the running cost (refer to Sec.III-C) and no terminal cost.
- *MPC with Heuristic Terminal*: Same as above, but with heuristic references included as a terminal cost.
- *MPC with HZD Reference*: The proposed whole-body MPC with HZD-based references for the running and terminal cost (refer to Sec. III-C).
- *Lumped Mass MPC*: Uses a simplified dynamics model for the planning stage by moving leg inertia to the torso, otherwise identical to *MPC with Heuristic Terminal*.
- *HZD with PD*: An offline generated HZD trajectory tracked by a joint level PD controller.

The results of these simulations are summarized in Table II. First, we remark that the Lumped Mass MPC model was introduced to highlight the effects of planning over the full system dynamics for the given platform. The particular structure of this model was chosen to resemble some properties of the simplified models mentioned in Sec. I, while allowing for an implementation independent comparison. Although the Lumped Mass MPC could withstand similar disturbances to the whole-body MPC for a specified standing

TABLE II: Maximum disturbance rejection and step adaption range (difference between smallest and largest observed step length). MPC planning frequency clamped at 100 Hz.

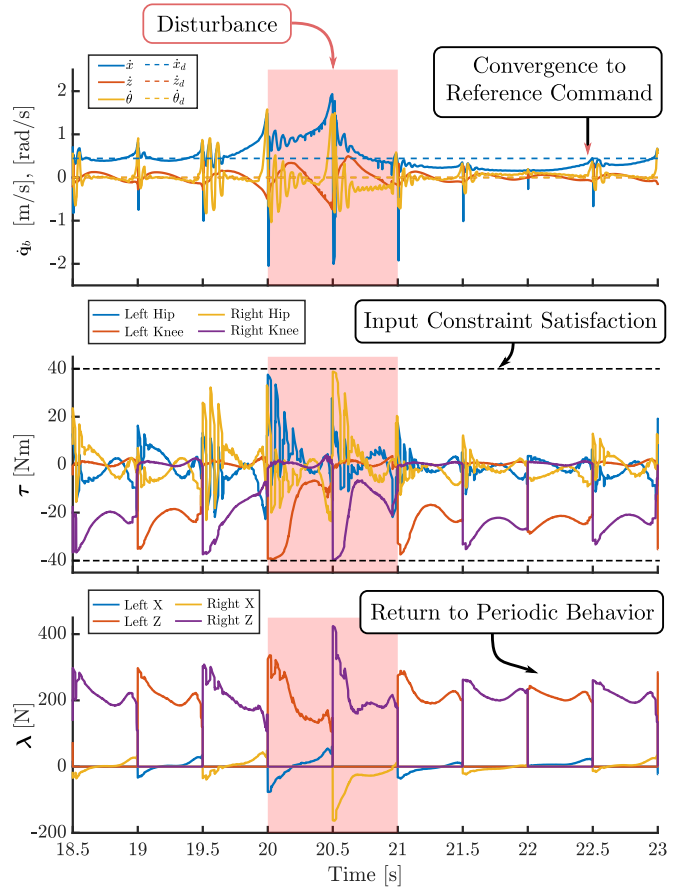|	Disturbance Rejection			Step Range
Horizon Length	2 s	0.5 s	0.2 s	2 s
Lumped Mass MPC	2 N	-	-	-
MPC + No Terminal	22 N	-	-	0.63 m
MPC + Heuristic	22 N	22 N	-	0.67 m
MPC + HZD	22 N	22 N	20 N	1.10 m
HZD + PD	30 N			0.14 m

Fig. 5: Simulation results for the MPC with Heuristic Terminal controller under a disturbance of $20\,\mathrm{N}$ applied in the forward (X) direction during the marked time of $1\,\mathrm{s}$, including states (top), torques (middle), and contact forces (bottom). The commanded forward walking velocity is $0.5\,\mathrm{m/s}$.

position, it was observed to have only a marginal ability to reject disturbances during dynamic motions like stepping in place and walking, no matter the horizon length. This confirms the need for whole-body online planning methods, especially for robots like AMBER-3M which have a non-negligible mass distribution concentrated in the legs.

Next, note that the MPC approach fails quickly when no terminal cost is present. When a heuristic terminal component is added, the robustness of the system dramatically increases. Furthermore, when the proposed MPC approach is combined with an HZD-based reference trajectory for running and terminal costs, the horizon length can be shortened to as low as 0.2 seconds, which drastically reduces the computational complexity. This could be essential to enable the whole body NMPC approach to be applied to a 3D biped with a larger number of degrees of freedom. These results emphasize the importance of the careful design of reference components, as their construction is tightly coupled with the performance of the overall system. Finally, it is important to note that at a disturbance of $22\,\mathrm{N}$ during walking the foot begins to slip, causing all of the MPC based methods to fail. The HZD with PD method exhibits more robustness to foot slipping and is therefore able to endure
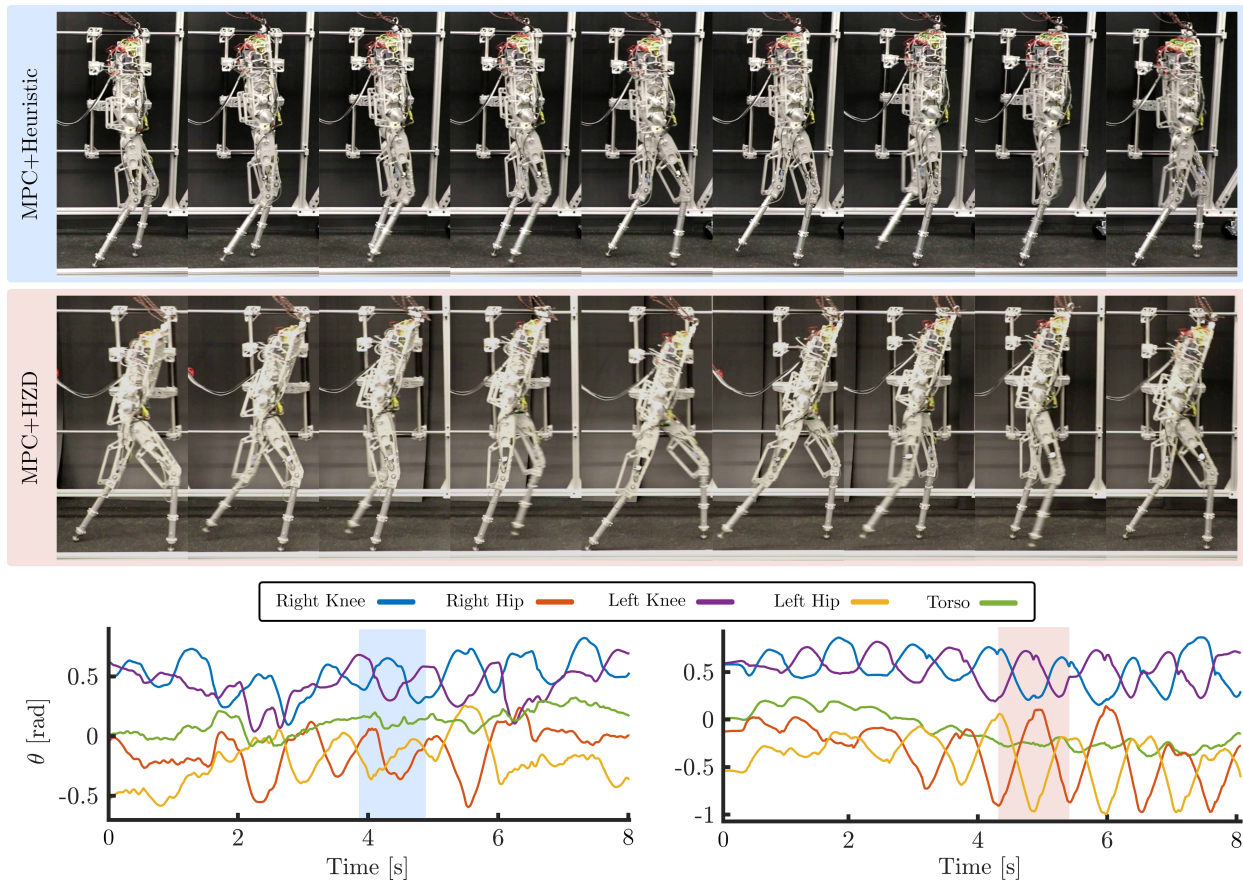
Fig. 6: Gait tiles and joint angle trajectories for forward walking behavior of the whole body MPC at a horizon length of 1 second (top, left), and the whole body MPC+HZD at a horizon length of 0.5 seconds (bottom, right). The HZD reference induces stronger periodic behaviors in the joint coordinates, correlated with the periodic nature of an HZD gait.

larger disturbances, as it does not model the disturbances. Note that the HZD and PD method is limited to periodic motions, and during disturbance rejection it heavily restricts the allowable stepping range, as reported in Table II. On the other hand, the MPC methods naturally have a large variability in footstep locations in order to stabilize the system. We believe that the ability to modulate step width and exhibit aperiodic motions will be critical for bipedal robots operating on real-world terrain. Future work will seek to combine the robustness of the HZD and PD method with the flexibility of the proposed MPC methods.

As seen in the supplementary video [43], the various proposed approaches react differently to disturbances. Specifically, the phase-based HZD with PD control achieves stability via implicit modification of the contact times, where the limbs are accelerated along the predefined reference trajectory. While this allows for significant disturbance rejection, it leads to the inputs being saturated for non-negligible amounts of time. On the other hand, in this MPC-based formulation, stability is achieved via explicit modification of the footstep locations, and is able to converge back to the desired reference trajectory in one to two steps while still satisfying state and input constraints. Incorporating optimization over the contact times into the MPC program is left as important future work, which would combine the benefits of both

approaches. A depiction of the disturbance rejection behavior of the MPC method can be seen in Fig. 5.

The MPC with a heuristic reference trajectory and a horizon length of 1.0 second, and the MPC with an HZD trajectory and a horizon length of 0.5 seconds were then deployed on the AMBER hardware. As seen in Fig. 6, both methods produce forward walking and have a visually distinct gait. We see in the joint angle trajectory data that the MPC with HZD method displays strong periodic behavior, similar to periodic motions expected with an HZD approach.

V. CONCLUSION AND OUTLOOK

In this work, we proposed a whole-body nonlinear MPC framework that enables online gait optimization using the full rigid body dynamics of a bipedal system. The viability of the presented control structure was shown in simulation and on hardware in a variety of robust dynamic behaviors, including standing, stepping in place, and walking. The addition of a trajectory tracking cost around an offline generated HZD reference enabled similarly robust locomotion at a significantly shorter planning horizon when compared with a heuristic reference or no reference. Motivated by the experimental results and promising reduction in computational complexity, future work will investigate the theoretical properties of using HZD trajectories as terminal components, as well as extensions to 3D walking bipeds.

REFERENCES

[1] M. Vukobratović and B. Borovac, "Zero-moment point—thirty five years of its life," *International journal of humanoid robotics*, vol. 1, no. 01, pp. 157–173, 2004.

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *International Conference on Intelligent Robots and Systems (IROS)*, vol. 1. IEEE/RSJ, 2001, pp. 239–246.

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

[4] P. M. Wensing and D. E. Orin, "High-speed humanoid running through control with a 3d-slip model," in *International Conference on Intelligent Robots and Systems*. IEEE/RSJ, 2013, pp. 5134–5140.

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

[6] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on mabel," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1170–1193, 2011.

[7] J. Reher, W.-L. Ma, and A. D. Ames, "Dynamic walking with compliance on a cassie bipedal robot," in *European Control Conference (ECC)*. IEEE, 2019, pp. 2589–2595.

[8] Y. Gong, R. Hartley, X. Da, A. Hereid, O. Harib, J.-K. Huang, and J. Grizzle, "Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway," in *American Control Conference (ACC)*. IEEE, 2019, pp. 4559–4566.

[9] Q. Nguyen, X. Da, J. Grizzle, and K. Sreenath, "Dynamic walking on stepping stones with gait library and control barrier functions," in *Alg. Foundations of Robotics XII*. Springer, 2020, pp. 384–399.

[10] J. Reher and A. D. Ames, "Inverse dynamics control of compliant hybrid zero dynamic walking," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2040–2047.

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

[12] J.-P. Sleiman, F. Farshidian, M. V. Minniti, and M. Hutter, "A unified mpc framework for whole-body dynamic locomotion and manipulation," *Robotics and Automation Letters*, vol. 6, no. 3, pp. 4688–4695, 2021.

[13] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the mit cheetah 3 through convex model-predictive control," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2018, pp. 1–9.

[14] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, "Feedback mpc for torque-controlled legged robots," in *International Conference on Intelligent Robots and Systems*. IEEE/RSJ, 2019, pp. 4730–4737.

[15] O. Villarreal, V. Barasuol, P. M. Wensing, D. G. Caldwell, and C. Semini, "Mpc-based controller with terrain insight for dynamic legged locomotion," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2436–2442.

[16] M. Krause, J. Englsberger, P.-B. Wieber, and C. Ott, "Stabilization of the capture point dynamics for bipedal walking based on model predictive control," *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 165–171, 2012.

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

[18] N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "Mpc for humanoid gait generation: Stability and feasibility," *Transactions on Robotics*, vol. 36, no. 4, pp. 1171–1188, 2020.

[19] M. Naveau, M. Kudruss, O. Stasse, C. Kirches, K. Mombaur, and P. Souères, "A reactive walking pattern generator based on nonlinear model predictive control," *Robotics and Automation Letters*, vol. 2, no. 1, pp. 10–17, 2016.

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

[21] T. Apgar, P. Clary, K. Green, A. Fern, and J. W. Hurst, "Fast online trajectory optimization for the bipedal robot cassie." in *Robotics: Science and Systems*, vol. 101, 2018, p. 14.

[22] X. Xiong and A. Ames, "3-d underactuated bipedal walking via h-lip based gait synthesis and stepping stabilization," *IEEE Transactions on Robotics*, 2022.

[23] J. Koenemann, A. Del Prete, Y. Tassa, E. Todorov, O. Stasse, M. Bennewitz, and N. Mansard, "Whole-body model-predictive control applied to the hrp-2 humanoid," in *International Conference on Intelligent Robots and Systems*. IEEE/RSJ, 2015, pp. 3346–3351.

[24] H. Diedam, D. Dimitrov, P.-B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2008, pp. 1121–1126.

[25] S. Faraji, S. Pouya, C. G. Atkeson, and A. J. Ijspeert, "Versatile and robust 3d walking with a simulated humanoid robot (atlas): A model predictive control approach," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1943–1950.

[26] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *International Conference on Humanoid Robots*. IEEE-RAS, 2014, pp. 295–302.

[27] C. Brasseur, A. Sherikov, C. Collette, D. Dimitrov, and P.-B. Wieber, "A robust linear mpc approach to online generation of 3d biped walking motion," in *International Conference on Humanoid Robots (Humanoids)*. IEEE-RAS, 2015, pp. 595–601.

[28] M. J. Powell, E. A. Cousineau, and A. D. Ames, "Model predictive control of underactuated bipedal robotic walking," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 5121–5126.

[29] F. Farshidian, M. Neunert, A. W. Winkler, G. Rey, and J. Buchli, "An efficient optimal planning and control framework for quadrupedal locomotion," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 93–100.

[30] E. Ambrose, W.-L. Ma, C. Hubicki, and A. D. Ames, "Toward benchmarking locomotion economy across design configurations on the modular robot: Amber-3m," in *Conference on Control Technology and Applications (CCTA)*. IEEE, 2017, pp. 1270–1276.

[31] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.

[32] M. Diehl, H. G. Bock, H. Diedam, and P.-B. Wieber, "Fast direct multiple shooting algorithms for optimal robot control," in *Fast motions in biomechanics and robotics*. Springer, 2006, pp. 65–93.

[33] C. Feller and C. Ebenbauer, "Relaxed logarithmic barrier function based model predictive control of linear systems," *Transactions on Automatic Control*, vol. 62, no. 3, pp. 1223–1238, 2016.

[34] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[35] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[36] A. Hereid and A. D. Ames, "Frost∗: Fast robot optimization and simulation toolkit," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2017, pp. 719–726.

[37] "OCS2: An open source library for optimal control of switched systems," [Online]. Available: https://github.com/leggedrobotics/ocs2.

[38] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The pinocchio c++ library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *International Symposium on System Integration (SII)*. IEEE/SICE, 2019, pp. 614–619.

[39] B. M. Bell, "Cppad: a package for c++ algorithmic differentiation," *Computational Infrastructure for Operations Research*, vol. 57, no. 10, 2012.

[40] C. D. Bellicoso, F. Jenelten, P. Fankhauser, C. Gehring, J. Hwangbo, and M. Hutter, "Dynamic locomotion and whole-body control for quadrupedal robots," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2017, pp. 3359–3365.

[41] H. Ferrolho, V. Ivan, W. Merkt, I. Havoutis, and S. Vijayakumar, "Inverse dynamics vs. forward dynamics in direct transcription formulations for trajectory optimization," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 12 752–12 758.

[42] A. D. Ames, "Human-inspired control of bipedal walking robots," *Transactions on Automatic Control*, vol. 59, no. 5, pp. 1115–1130, 2014.

[43] "Video of the experimental results." https://youtu.be/3g8ZNsCWdOA.