

Received January 29, 2020, accepted February 21, 2020, date of publication March 12, 2020, date of current version March 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2980446

Variational-Based Optimal Control of Underactuated Balancing for Dynamic Quadrupeds

MATTHEW CHIGNOLI¹ AND PATRICK M. WENSING², (Member, IEEE)

¹Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

²Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46665, USA

Corresponding author: Matthew Chignoli (chignoli@mit.edu)

This work was supported in part by the National Science Foundation under Grant CMMI-1835186.

ABSTRACT This paper presents a control strategy for quadruped balancing that enables postural control in underactuated contact configurations (e.g., when standing on two point feet). Underactuated balancing has received considerable attention with prototype control models such as the cart pendulum or acrobot. Yet, when attempting to transition these solutions to balance in legged robots, technical challenges related to friction-limited contacts and the underlying manifold structure of the configuration space prevent straightforward application. This paper presents a new balance control framework that combines constrained optimal control strategies with recent variational-based linearization approaches to solve the balancing problem for a common simplified quadruped model. The controller is implemented as a convex quadratic program (QP) that uses an unconstrained optimal control solution to approximate a friction-constrained optimal policy. Unlike state-of-the-art QP-based balance controllers, the method is able to handle balance in underactuated regimes. Via comparison to model-predictive control strategies, the proposed formulation is highly compact, requiring less computation, while still showing the ability to handle extreme friction limitations. Simulation and hardware results with the MIT Mini Cheetah demonstrate the capabilities of the controller to exploit body angular momentum for disturbance recovery on two feet, and to recover from cases where the center of mass exits the support polygon. These results and the generality of the formulation suggest exploration for further application to bipeds and humanoids.

INDEX TERMS Dynamics, legged locomotion, linearization techniques, optimal control, robot control.

I. INTRODUCTION

The development of robot platforms capable of dynamic legged locomotion offers a solution to the mobility bottleneck currently limiting the impact of mobile robots. The ability to negotiate challenging terrains, both natural and constructed, with high speed and agility will allow robots to work seamlessly alongside humans as well as in places too dangerous for humans. There are a number of technical challenges that prevent current platforms from fulfilling this goal. Features of underactuation, friction limited contacts, and nonlinear hybrid dynamics complicate the control problem, motivating substantial research into locomotion control algorithms in the past decades. In this work, we focus specifically

The associate editor coordinating the review of this manuscript and approving it for publication was Chenguang Yang¹.

on the challenging problem of underactuated balancing for quadruped robots (Fig. 1). This canonical problem, and the proposed algorithms to solve it, are more generally applicable to other dynamic behaviors and platforms (e.g., bipeds and humanoids).

Quadratic programming (QP) based approaches to control have driven considerable progress in the field of legged robots [1]–[5]. Existing QP-based controllers are attractive due to their ability to be run in real time while simultaneously reasoning about control input constraints. Balance control QPs are generally based on the idea of optimizing the distribution of ground reaction forces, subject to contact force constraints, to produce motions of the body that track a reference state [1], [2]. Whole-body inverse-dynamics control via QP has produced perturbation robust balancing for bipeds [6], [7] as well as quasi-static quadruped walking [8], [9].



FIGURE 1. The proposed control framework for balancing in underactuated contact configurations achieves two-leg balance of the MIT Mini Cheetah quadruped.

Although the walking in these cases is slow, the quadrupeds are capable of navigating challenging sloped terrains due to the ability of the QP to effectively handle friction constraints.

Despite the numerous advantages of QP-based approaches to whole-body control, these methods are limited by their failure to look beyond satisfaction of instantaneous desired dynamics. The challenge of avoiding undesired myopic behaviors has motivated the increasingly widespread use of model predictive control (MPC). MPC involves computing control inputs through repeated solution of a receding horizon optimal control problem (OCP) [10]. For quadruped balance control and related locomotion problems, classic optimal control approaches for underactuated balancing of nonlinear systems [11] provide important insights for real-time solutions to the OCP. Using a model of the system, optimal controllers find the control policy that minimizes some objective function over a prescribed time horizon. Many strategies involve feedback linearization [11] to avoid dealing with complex nonlinear dynamics. The resulting linear dynamics and often-assumed quadratic cost functions pose a well known optimal control problem, solvable backward in time via a Riccati equation. The drawback of using these optimal controllers for legged systems is the difficulty of enforcing constraints on control inputs and ground forces.

Other recent controllers for nonlinear dynamic systems have shown the ability to solve these optimal control problems using differential dynamic programming (DDP) [12]. DDP uses locally quadratic models of the cost function and dynamics to iteratively sweep backward over the cost and forward through the dynamics until convergence to a locally optimal policy. Such controllers have produced dynamic quadruped gaits [13] and humanoid balance for reaching tasks [14]. The accommodation of box inequality constraints into the DDP formulation has enabled the enforcement of control constraints [15]. Other recent work [16] has introduced coordinate-free extensions for application to Lie groups, increasing the suitability of the method for locomotion tasks. Despite these advances, however, DDP, like other nonlinear-based approaches [17], is complicated by non-convexity that leads to potential issues with local optima and computational challenges that pose an obstacle to real-time implementation.

To avoid non-convexity in the MPC formulation, other MPC strategies often make model simplifications to work with linear dynamic models [18], [19]. The linear dynamic model allows the OCP to be posed as a single convex optimization problem, which avoids the issue of local optima and allows for enforcement of inputs constraints. These controllers have demonstrated impressive locomotion capabilities, but their performance depends on the robot's state being contained within the small locally valid region around the reference trajectory. Additional measures like regularization can improve solution quality when nonlinearities remain [20], but the fundamental challenge of computational cost posed by model complexity and horizon length persists.

This work proposes an alternative control algorithm that synthesizes optimal control and quadratic programming, as inspired by the approach demonstrated in [21]. The approach involves solving an unconstrained linear quadratic regulator (LQR) problem to compute the optimal cost-to-go. Solving the LQR also yields a linear optimal controller, but constraints may prevent this policy from being achieved. Instead, cost-to-go information can be re-purposed to design an input-constrained control law that is able to consider the long-term consequences of present actions. In this previous work [21], the unconstrained cost-to-go was used to control the fully-actuated zero moment point (ZMP) and Center of Mass (CoM) dynamics for a humanoid, which are equivalent to those of the common Linear Inverted Pendulum (LIP) model [22]–[24]. Quadruped dynamics, however, are not easily captured by a fully-actuated LIP model since many contact configurations (e.g., on two feet) are underactuated, and orientation dynamics play a significant role in center of mass regulation.

Traditional linearization techniques also cannot be applied to the quadruped dynamics because even for single rigid-body models [25], the orientation dynamics evolve on the nonlinear manifold $SE(3) = SO(3) \times \mathbb{R}^3$. We address this challenge by applying variational-based linearization (VBL) techniques [26] to a reduced-order model of the quadruped and using that linear model for the unconstrained LQR. This new application of VBL and the resulting variational-based optimal controller for underactuated balancing of nonlinear control-constrained systems in $SE(3)$ constitutes the main contribution of this paper. Our VBL-QP control law, as it is referred to, captures the benefits of larger-scale MPC formulations while maintaining fast solve times and loop rates via a compact formulation. We demonstrate its viability via full-body dynamic simulation as well as in hardware experiments with the MIT Mini Cheetah quadruped.

The remainder of the paper is organized as follows. Section II details the proposed controller design and highlights the paper's technical contributions. Section III describes the rigid-body model used for control, motivated by the design of the Mini Cheetah. Section IV overviews the criteria and steps for applying variational-based linearization on manifolds as well as how the linear variation dynamics are used in control. Section V presents and discusses simulation

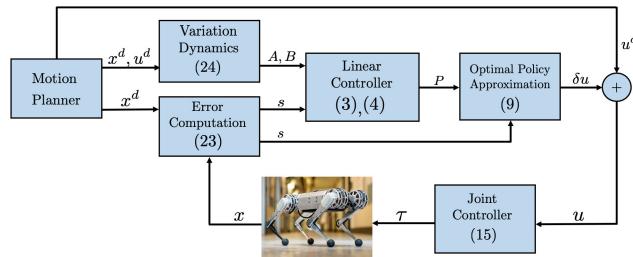


FIGURE 2. Control architecture for the proposed balance controller.

and hardware results demonstrating the capabilities of the controller, and Section VI concludes with a summary.

II. CONTROLLER DESIGN

The proposed VBL-QP framework is predicated on an approximation of a cost-to-go using an unconstrained LQR to drive the linearized state error of the system to zero. The unconstrained cost-to-go from the LQR is used to formulate a QP that descends this cost-to-go while enforcing input constraints. This section motivates the two-leg balance task as a relevant problem in legged locomotion and overviews the two key components of the VBL-QP framework: linearized dynamics on $SE(3)$ and a QP formulation. The architecture for the proposed VBL-QP controller is shown in Fig. 2.

A. TWO-LEG BALANCE

The viability of the VBL-QP controller is demonstrated by its ability to negotiate the control challenges presented when transitioning from four-leg balance to two-leg balance. The first of these challenges arises from the reduced support polygon created by taking two feet out of contact with the ground. Rather than the broad support polygon afforded by four stance feet, the set of stable positions for the CoM when balancing on two point feet becomes the “supporting line” between the two feet in contact. The second challenge comes as a result of the loss of actuation. With only two feet in contact with the ground, the system is underactuated and the set of feasible wrenches that the stance feet can exert on the body is reduced. For instance, a pure moment about the supporting line can no longer be generated. This challenge presents a problem not only for achieving desired orientation changes, but also for stability, since the limited set of wrenches hinders the ability to recover from disturbances.

Addressing this underactuated balancing problem requires exploitation of system dynamics beyond the current instant. The key advantage of the VBL-QP controller’s compact formulation is that it offers these predictive capabilities while maintaining efficient solve times (0.1-1 ms) comparable to those of conventional, non-predictive balance QPs [25]. Unlike constrained MPC approaches that involve nonlinear optimization like DDP or comparatively high dimensional convex optimizations, VBL-QP only requires solving a Riccati equation and using that solution in an efficiently solvable QP. As a result, VBL-QP can be run at a higher frequency (500 Hz vs. 20-30 Hz) and does not suffer from

the same prediction horizon limitations. Simulation results presented in Section V demonstrate why this is advantageous for a task like two-leg balance.

B. LINEARIZED DYNAMICS

The optimal control strategies for balance control proposed in this paper require a linear dynamic model. In order to implement these strategies, a variational-based approach [26] to linearization is applied to the Mini Cheetah’s nonlinear reduced-order model dynamics. The aim of the VBL is to express the dynamics of the system in the form

$$\dot{s} = A(x^d(t))s + B(x^d(t))\delta u, \quad (1)$$

where s approximates the error between the actual and reference states of the system, x^d is a time-varying desired reference trajectory, and δu is a variation to the reference control input u^d such that

$$u = u^d + \delta u. \quad (2)$$

Note, however, the challenge to conventional linearization approaches in coordinates if the state x contains an element of $SO(3)$, as it does in our case. With this challenge in mind, the variations s can roughly be considered as a local approximation of the displacement between two points on a manifold. The VBL exploits specific variation expressions on the nonlinear manifold $SO(3)$, which allows for parameterization of the error directly based on infinitesimal rotations [26]. The ability to parameterize the error in this way offers a significant advantage over typical dynamic models that employ Euler angles, which can suffer from singularities, or quaternions, which are significantly more complex to deal with and suffer from issues like the unwinding phenomenon.

The VBL, meanwhile, is able to offer a compact linear model that is singularity-free and that describes error in a manner that is invariant to the orientation of the coordinate frame attachment. Furthermore, the resulting linear dynamics from the VBL are controllable for all possible reference trajectories about which the system is linearized [26]. The locally valid region of attraction for the linearization, however, varies depending on the dynamic system. In the case of two-leg balance, we empirically find that the region of attraction is suitably large to capture most kinematically feasible translations and rotations of the robot.

C. OPTIMAL VALUE APPROXIMATION

The main contribution of this paper is a control strategy for underactuated balancing in input-constrained systems using an approximation of the optimal cost-to-go. The optimal cost-to-go is a function that represents the minimum cost, according to an objective function, associated with moving from a given time-dependent state to a desired final state [27]. The approximation proposed in this paper involves using the unconstrained cost-to-go (UCTG) as a proxy for the constrained cost-to-go (CCTG) and then formulating a QP that descends the UCTG while enforcing constraints.

Using the UCTG offers the advantage of considering the effect of system dynamics on current and future costs. Empirically, we find that the additional information afforded by using the UCTG enables underactuated balancing in extreme low-friction regimes, suggesting that underactuation, not friction, can be attributed as the main challenge for balance on two legs.

The computation of the UCTG is enabled by the VBL, which allows the OCP to be posed in a conventional linear-quadratic form [27]

$$\begin{aligned} V_{unc}^*(s_0) = \min_{\delta u(\cdot)} \int_0^\infty & [s(t)^T Q s(t) + \delta u(t)^T R \delta u(t)] dt, \\ \text{s.t. } & \dot{s}(t) = A s(t) + B \delta u(t), \\ & s(0) = s_0, \end{aligned} \quad (3)$$

where $Q \in \mathbb{R}^{12 \times 12}$ and $R \in \mathbb{R}^{12 \times 12}$ are positive-definite weighting matrices. With the control problem posed in this form, the UCTG takes the form $V_{unc}^*(s) = s^T P s$, where $P \in \mathbb{R}^{12 \times 12}$ is found via solution of the continuous-time algebraic Riccati equation [27]

$$0 = Q - P B R^{-1} B^T P + P A + A^T P. \quad (4)$$

Introducing the contact force constraints into (3) creates an OCP that is not tractable to solve

$$\begin{aligned} V^*(s_0) = \min_{\delta u(\cdot)} \int_0^\infty & [s(t)^T Q s(t) + \delta u(t)^T R \delta u(t)] dt, \\ \text{s.t. } & \dot{s}(t) = A s(t) + B \delta u(t), \\ & \delta u(t) \in \mathcal{U}(u^d), \\ & s(0) = s_0, \end{aligned} \quad (5)$$

where $\mathcal{U}(u^d)$ represents a linear approximation of the friction cone

$$\mathcal{U}(u^d) = \{\delta u \mid \underline{d} \leq C(u^d + \delta u) \leq \bar{d}\}.$$

The CCTG, V^* , and optimal variational control policy, $\delta u^*(s)$, for this new control problem might be sought by solving the time-invariant form of Hamilton-Jacobi-Bellman (HJB) Equation

$$0 = \min_{\delta u \in \mathcal{U}(u^d)} \left[l(s, \delta u) + \frac{\partial V^*}{\partial s} [As + B\delta u] \right], \quad (6)$$

where $l(s, \delta u)$ is the running cost term that comes from the objective function.

The advantage of the minimization on the right hand side of the HJB equation is that it considers both the costs incurred at the current moment in time as well as the effect on the lowest possible costs incurred in the future. This consideration of long and short-term costs is incorporated in the VBL-QP approach, which makes the approach well suited for handling the challenge of underactuation when balancing on two feet.

The HJB partial differential equation (6), however, is difficult to solve even approximately, and approximating an infinite horizon solution numerically is computationally expensive for real-time control. Rather than solve this partial differential equation to get the CCTG and the optimal control

policy, the known UCTG V_{unc}^* is used as a proxy for V^* , and the optimal control policy for (5) is then approximated via the constrained optimization

$$\min_{\delta u \in \mathcal{U}(u^d)} \left[l(s, \delta u) + \frac{\partial V_{unc}^*}{\partial s} [As + B\delta u] \right]. \quad (7)$$

This strategy of replicating the minimization step of the HJB equation allows for rapid execution of the control loop, while still providing approximations of the constrained optimal control policy that account for the system dynamics. Perhaps surprisingly, the approximate optimal policy from (7) is found to be sufficiently accurate for balancing across a range of typical operating conditions in simulation as well as on hardware. It is noted that in cases where there exists a variational control input δu^* that satisfies the friction constraints with a zero objective function value in (7), the solution to the minimization (7) is equal to unconstrained optimal policy. Otherwise, the control law will provide a variational control input with lowest long-term anticipated cost that satisfies the current friction constraints.

Applying the objective function, dynamics, and UCTG from (3), the HJB minimization step takes the form

$$\min_{\delta u \in \mathcal{U}(u^d)} \left[s^T Q s + \delta u^T R \delta u + 2s^T P [As + B\delta u] \right]. \quad (8)$$

The quadratic state error terms $s^T Q s$ and $2s^T P A s$ are independent of δu and can thus be removed from the optimization. Future state costs are captured by UCTG via the Riccati solution P in $2s^T P B \delta u$. Additionally, experimentation proved that adding a penalty for deviations from the previous optimal solution improved stability. The resulting optimization problem is solved every control loop

$$\min_{\delta u \in \mathcal{U}(u^d)} \left[\delta u^T R \delta u + 2s^T P B \delta u + \beta \|\delta u - \delta u_{prev}^*\|^2 \right], \quad (9)$$

where δu_{prev}^* is the previous optimal solution and $\beta > 0$ dictates the strength of solution filtering. Since the dynamics are affine and the cost is quadratic, this problem is a QP.

The VBL-QP policy (8) can be viewed as the limiting case of a one-step horizon constrained discrete-time LQR problem [28] as time discretization becomes infinitely fine. For any fixed time discretization, constraints can be enforced over a longer horizon without disrupting this QP form [28], and explicit solutions exist [29] for the case when the cost function and dynamic matrices A, B, Q, R remain fixed. In these cases, there further exist algorithms to determine a prediction horizon [30], [31] that ensures persistent feasibility by considering the concatenation of a finite horizon constrained solution with a subsequent infinite horizon unconstrained solution. The approach here is similar in spirit, but instead employing the shortest prediction horizon possible. Perhaps surprisingly, the information-rich nature of the UCTG and use of the HJB minimization step enables the quadruped to maintain balance on two legs across a wide range of operation and under stringent frictional limitations.

III. SIMPLIFIED DYNAMICS

This section overviews the design of the MIT Mini Cheetah quadruped [32] and justifies a reduced-order dynamic model for control design. The simplified dynamics of the reduced-order model are able to significantly reduce solution times for the constrained optimization in the VBL-QP while minimally affecting the performance of the controller. The impact of assumptions made by the simplified model is reduced by quasi-static compensation for the neglected effect that leg mass has on the dynamics.

A. ROBOT DESIGN OVERVIEW

The MIT Mini Cheetah is a low-cost, lightweight quadruped intended to facilitate hardware experimentation for control of dynamic legged robots [32]. Standing 0.3 m tall with a mass of 9 kg, the Mini Cheetah is significantly smaller than its relative, the MIT Cheetah 3, yet the two share key design features [33]. Both robots feature backdriveable actuators capable of controlling ground reaction forces through proprioception [34], including a hip abduction/adduction actuator that enables full 3D control of ground reaction forces. The majority of the Mini Cheetah's mass, roughly 90%, is contained in the trunk, with the leg design optimized to reduce limb inertia while maximizing range of motion at all joints. The Mini Cheetah has replicated numerous of the Cheetah 3's highly dynamic behaviors such as trot, trot-run, bounding, and pronking as well as novel behaviors such as a 360° backflip from standing [32].

B. REDUCED-ORDER MODEL

The design of the Mini Cheetah allows for its dynamics to be approximated as a single rigid body with massless legs and the CoM at the geometric center of the body. Using this representation for the dynamics, the state of the robot is abstracted by

$$x := [p_c \quad \dot{p}_c \quad R_b \quad \omega], \quad (10)$$

where $p_c \in \mathbb{R}^3$ is the position of the CoM; $\dot{p}_c \in \mathbb{R}^3$ is the velocity of the CoM; $R_b \in SO(3)$ is the orientation of the body frame $\{B\}$ with respect to the inertial frame $\{0\}$, and $^B\omega \in \mathbb{R}^3$ is the body's angular velocity. Variables without a superscript are assumed to be expressed in the inertial frame. Variables expressed in the body-fixed frame will have a leading superscript B . For example, $^B\omega$ is the body's angular velocity expressed in the body frame, while ω is the body's angular velocity in the inertial frame. An illustration of these coordinate frames and other relevant vectors is shown in Fig. 3.

The position and orientation of the body are controlled via the ground reaction forces $f_i \in \mathbb{R}^3$ applied at the contact locations of the feet $p_i \in \mathbb{R}^3$. Contacts are assumed to be stationary. The subscripts i are indices for each foot, with $i = 1, 2, 3, 4$ denoting the front right, front left, back right, and back left feet, respectively. The locations of these feet relative to the body's CoM are denoted by $r_i = p_i - p_c$.

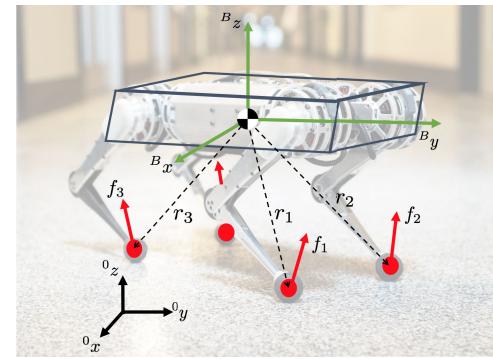


FIGURE 3. Illustration of the rigid-body model and the body- and earth-fixed coordinate frames. The vectors r_i and f_i represent the vectors from the CoM to the feet and the ground reaction forces, respectively.

These ground reaction forces produce a net external wrench on the body consisting of a net force, f , and a net moment about the CoM, τ , given by

$$\begin{bmatrix} f \\ \tau \end{bmatrix} = \sum_{i=1}^4 \begin{bmatrix} I_3 \\ \widehat{r}_i \end{bmatrix} f_i, \quad (11)$$

where I_3 is the 3×3 identity matrix, and \widehat{r}_i maps $\mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ such that $\widehat{r}_i y = r_i \times y$ for all $r_i, y \in \mathbb{R}^3$. The resulting rigid-body dynamics are given by

$$\dot{x} = \frac{d}{dt} \begin{bmatrix} p_c \\ \dot{p}_c \\ R_b \\ ^B\omega \end{bmatrix} = \begin{bmatrix} \dot{p}_c \\ \frac{1}{m}f - g \\ R_b \widehat{B}\widehat{\omega} \\ ^B I^{-1} (R_b^T \tau - ^B\widehat{\omega} ^B I ^B\omega) \end{bmatrix}, \quad (12)$$

where m is the mass of the body, $g \in \mathbb{R}^3$ is the acceleration of the body due to gravity, and $^B I \in \mathbb{R}^{3 \times 3}$ is the inertia tensor about the CoM. The rotational inertia is assumed constant in this work. The contact location at each foot will remain fixed as long as the ground reaction forces lie within the friction cone

$$\left\{ (f_x, f_y, f_z) \in \mathbb{R}^3 \mid \sqrt{f_x^2 + f_y^2} \leq \mu f_z \right\}, \quad (13)$$

where f_x and f_y are tangential components of the ground reaction force at a given leg, f_z is the normal force, and μ is the coefficient of static friction between the foot and the ground. The linear approximation of this friction cone constraint used by the OCP (5) is an inscribed friction pyramid, defined by

$$-\frac{1}{\sqrt{2}}\mu f_z \leq f_x, f_y \leq \frac{1}{\sqrt{2}}\mu f_z. \quad (14)$$

C. LEG MASS COMPENSATION

Controller performance can be significantly improved by a simple technique that partially captures the effect of leg mass on the robot's dynamics while minimally impacting computational cost. The technique involves the low-level torque controller that converts commanded ground reaction forces to joint torques. In addition to converting forces to

torques, the controller also adds configuration-dependent gravity compensation torques that account for the mass of the legs. The joint torques at leg i are computed via

$$\tau_i = J_i^T f_i + \tau_{g,i}, \quad (15)$$

where $J_i \in \mathbb{R}^{3 \times 3}$ is the foot Jacobian for leg i and $g \in \mathbb{R}^{18}$ is the generalized gravity vector for the full-body dynamics of the quadruped

$$\tau_g = \begin{bmatrix} \tau_{g,fb} \\ \tau_{g,1} \\ \vdots \\ \tau_{g,4} \end{bmatrix}. \quad (16)$$

The vector $\tau_{g,fb} \in \mathbb{R}^6$ is the gravity compensation wrench for the floating base and is not used for low-level torque control. The vectors $\tau_{g,i} \in \mathbb{R}^3$ for $i = 1, 2, 3, 4$ are the gravity compensation torques for the joints of each leg.

This generalized gravity vector can be computed via the inverse dynamics algorithm in [35]. This computation takes on the order of $10\mu\text{s}$ and requires only the body orientation from the state estimation and joint position information that comes from the leg controller. Rather than dramatically increase the solve time of the constrained optimization by accounting for the effect of leg mass on the dynamics, this method allows for rapid QP solution and quasi-static compensation for leg mass.

IV. VARIATIONAL-BASED LINEARIZATION

The concept of variational-based linearization for dynamics evolving on nonlinear manifolds was introduced in Section II. As was mentioned, a linear dynamic model in the form (1) is critical for implementation of the VBL-QP controller, specifically for solving the unconstrained OCP (3). This section explains the criteria and steps required to apply VBL to a nonlinear system, and then carries out those steps for the reduced-order dynamic model of the Mini Cheetah (12).

A. ERROR PARAMETERIZATION ON MANIFOLDS

Linearization along a reference trajectory on a manifold involves taking the variation of the state with respect to that reference trajectory. The variation can roughly be considered as a local approximation of the displacement between two points on a manifold. On the manifold $SO(3)$, the manifold of interest for the Mini Cheetah's orientation dynamics, the variation with respect to a reference trajectory R_b^d is given by [36]

$$\delta R_b = \frac{d}{d\epsilon} \Big|_{\epsilon=0} R_b^d \exp(\epsilon^B \hat{\eta}) = R_b^d \delta^B \hat{\eta}, \quad (17)$$

where $^B\eta \in \mathbb{R}^3$ is an approximation of the angle-axis error describing the rotation necessary to achieve the desired orientation. The exponential map $\exp : \mathfrak{so}(3) \rightarrow SO(3)$ maps a skew-symmetric matrix to a rotation matrix such that

$$\exp(\hat{\eta}) = I_3 + \frac{\hat{\eta}}{\|\eta\|} \sin \|\eta\| + \frac{\hat{\eta}^2}{\|\eta\|^2} (1 - \cos \|\eta\|). \quad (18)$$

The subsequent variation in angular velocity, expressed in body coordinates, is given by

$$\delta^B \omega = {}^B \hat{\omega}^d {}^B \eta + {}^B \dot{\eta}. \quad (19)$$

For sufficiently close R_b and R_b^d , the variations ${}^B \eta$ and $\delta^B \omega$ can be treated as a linear approximation of the exact error, e , between the desired and actual state defined by

$$\begin{aligned} e_R &= \frac{1}{2} ((R_b^d)^T R_b - R_b^T R_b^d)^\vee, \\ e_\omega &= {}^B \omega - (R_b^T R_b^d) {}^B \omega^d. \end{aligned} \quad (20)$$

where the vee map, $\vee : \mathfrak{so}(3) \rightarrow \mathbb{R}^3$, is such that $\hat{x}^\vee = x$, $\forall x \in \mathbb{R}^3$. Additionally, although not developed here, similar parameterizations for variations on the manifold \mathbb{S}^2 also exist [26].

B. VARIATION DYNAMICS

For a given dynamically feasible reference trajectory $x^d(t)$ and the nominal control input $u^d(t)$ to follow this trajectory, the linear variation dynamics for the system can be found in the following manner [26]. This technique is valid only when the system is control affine with respect to u and when the dynamic model of the system consists only of vector addition, dot product, cross product, and matrix multiplication with a matrix or vector. The first step is to take the variation on both sides of the dynamic equations, which involves recursively applying the following formulae

$$\begin{aligned} \delta(x + y) &= \delta x + \delta y, \\ \delta(x \times y) &= \delta x \times y_d + x_d \times \delta y, \\ \delta(x \cdot y) &= \delta x \cdot y_d + x_d \cdot \delta y, \\ \delta(R_1 x) &= \delta R_1 x_d + R_{1,d} \delta x, \\ \delta(R_1 R_2) &= (\delta R_1) R_{2,d} + R_{1,d} (\delta R_2) \end{aligned} \quad (21)$$

where $x, y \in \mathbb{R}^3$, $R_1, R_2 \in SO(3)$, and δ represents the variation. For all systems satisfying the given criteria, the variation will be linear with respect to the state error s and the variational control input δu . These terms can then be rearranged into a linear system that forms the variation-based dynamics (1).

All of the above linearization steps, as well as the VBL-QP controller in general, are applicable to both time-varying and time-invariant systems. The time dependence of the system becomes especially relevant when dealing with the solution to the unconstrained OCP (3). In the time-invariant infinite-horizon case, the UCTG is computed via the form of the continuous-time algebraic Riccati equation given by (4). In the finite-horizon case, the OCP can still be solved with a differential Riccati equation, but some additional measures are required. For a prediction horizon T , a terminal condition $P(t = T) = P_f$, where $P_f = P_f^T \geq 0 \in \mathbb{R}^{n \times n}$, must be specified for the terminal state $s(t = T)$. This terminal condition is used to solve the following Riccati equation in backwards in time

$$\begin{aligned} -\dot{P}(t) &= Q - P(t)B(t)R^{-1}B(t)^T P(t) \\ &\quad + P(t)A(t) + A(t)^T P(t). \end{aligned} \quad (22)$$

Depending on the problem, numerical solutions to this equation may be significantly less computationally expensive than conventional finite-horizon optimal control solutions in existing MPC approaches, both linear and nonlinear, which would allow the VBL-QP controller to predict over longer horizons.

C. APPLICATION TO MINI CHEETAH DYNAMICS

These principles of variations along a reference trajectory are next applied to the reduced-order dynamics of the Mini Cheetah in order to complete the linearization. Using the variation formulae in (17) and (19), the approximate linear error $s \in \mathbb{R}^{12}$ for the state of the reduced-order model of the quadruped (10) is given by

$$s := \begin{bmatrix} \delta p_c \\ \delta \dot{p}_c \\ {}^B\eta \\ {}^B\omega \end{bmatrix} \approx \begin{bmatrix} e_p \\ e_v \\ e_R \\ e_\omega \end{bmatrix} = \begin{bmatrix} p_c - p_c^d \\ \dot{p}_c - \dot{p}_c^d \\ \frac{1}{2}((R_b^d)^T R_b - R_b^T R_b^d)^\vee \\ {}^B\omega - (R_b^T R_b^d) {}^B\omega^d \end{bmatrix}. \quad (23)$$

Because the variations δp_c and $\delta \dot{p}_c$ remain strictly on the linear manifold \mathbb{R}^3 , these variations actually match the exact errors e_p and e_v regardless of proximity to the reference trajectory. The next step is to recursively apply the variation formulae (21) to the reduced order dynamics (12). The resulting variation dynamics take the form

$$\frac{d}{dt} \begin{bmatrix} \delta p \\ \delta \dot{p} \\ {}^B\eta \\ {}^B\omega \end{bmatrix} = \begin{bmatrix} \delta \dot{p} \\ \frac{1}{m} \sum \delta f_i - g \\ -{}^B\omega^d {}^B\eta + \delta {}^B\omega \\ {}^B I^{-1} (\delta R_b^T \sum \tau_i^d + (R_b^d)^T \sum \delta \tau_i) - c \end{bmatrix}, \quad (24)$$

where the following terms can be rewritten as

$$\delta u = [\delta f_1^T \quad \delta f_2^T \quad \delta f_3^T \quad \delta f_4^T]^T, \quad (25)$$

$$\sum \delta f_i = \sum_{i=1}^4 \delta f_i, \quad (26)$$

$$\delta R_b^T = -{}^B\hat{\eta}(R_b^d)^T, \quad (27)$$

$$\sum \tau_i^d = \sum_{i=1}^4 (\hat{r}_i^d f_i^d), \quad (28)$$

$$\sum \delta \tau_i = \sum_{i=1}^4 \hat{f}_i^d \delta p + \sum_{i=1}^4 (\hat{r}_i^d \delta f_i), \quad (29)$$

$$c = {}^B I^{-1} (\widehat{{}^B I} {}^B \omega^d - \widehat{{}^B \omega^d} {}^B I) \delta {}^B \omega. \quad (30)$$

In order to express the dynamics in the linear form offered by (1), the variation dynamics are factored with respect to the error state s and the variational control input δu . The resulting state-space matrices A and B , which depend only

on the reference trajectory, are given by

$$A = \begin{bmatrix} 0_3 & I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & -\widehat{{}^B \omega^d} & I_3 \\ A_{4,1} & 0_3 & A_{4,3} & A_{4,4} \end{bmatrix}, \quad (31)$$

where

$$A_{4,1} = {}^B I^{-1} (R_b^d)^T \sum \hat{f}_i^d, \quad (32)$$

$$A_{4,3} = {}^B I^{-1} (\widehat{(R_b^d)^T} \sum \tau_i^d), \quad (33)$$

$$A_{4,4} = {}^B I^{-1} (\widehat{{}^B I} \widehat{{}^B \omega^d} - \widehat{{}^B \omega^d} {}^B I), \quad (34)$$

and

$$B = \begin{bmatrix} 0_3 & \dots & 0_3 \\ \frac{1}{m} I_3 & \dots & \frac{1}{m} I_3 \\ 0_3 & \dots & 0_3 \\ {}^B I^{-1} (R_b^d)^T \hat{r}_1^d & \dots & {}^B I^{-1} (R_b^d)^T \hat{r}_4^d \end{bmatrix}. \quad (35)$$

For cases where a leg is taken out of contact with the ground and therefore no longer capable of contributing to the net wrench on the body, the corresponding columns of B are set to zero.

D. REFERENCE CONTROL INPUT

The reference forces f_i^d that appear in the A matrix describe the ground reaction forces necessary to achieve the reference motion x^d and are related to u^d via

$$u^d = [(f_1^d)^T \quad (f_2^d)^T \quad (f_3^d)^T \quad (f_4^d)^T]^T. \quad (36)$$

For the balance control application, the trajectory of the robot is treated as a series of discrete reference states. To realize the desired accelerations for each reference state, a reference input u^d is found such that

$$\underbrace{\begin{bmatrix} I_3 & I_3 & I_3 & I_3 \\ \hat{r}_1^d & \hat{r}_2^d & \hat{r}_3^d & \hat{r}_4^d \end{bmatrix}}_{H^d} u^d = \underbrace{\begin{bmatrix} m(\ddot{p}^d + g) \\ I_g \dot{\omega}^d + \hat{\omega}^d I_g \omega^d \end{bmatrix}}_{b^d}, \quad (37)$$

where $H^d u^d$ is the wrench produced by the ground reaction forces, b^d is the desired spatial acceleration, and I_g is the inertia tensor about the CoM expressed in the world frame. To find forces that best satisfy this equation, the following friction constrained QP is formulated:

$$\begin{aligned} \min_{u^d} (H^d u^d - b^d)^T S (H^d u^d - b^d) \\ \text{s.t. } \underline{d} < C u^d < \bar{d} \end{aligned} \quad (38)$$

where $S \in \mathbb{R}^{6 \times 6}$ is a positive-definite weighting matrix and $\underline{d}, C, \bar{d}$ comprise the matrix form of (14). This paper assumes static reference states with no desired linear or angular accelerations of the body. Consequently, the optimization simply involves finding the normal forces $f_{i,z}^d$ that support the robot's weight without generating moments about its CoM. This optimization need only be carried out when the desired $x - y$ position of the CoM changes.

TABLE 1. Mini cheetah controller parameters.

Parameter	Value	Units
m	9.0	kg
I_{xx}	0.025	$\text{kg} \cdot \text{m}^2$
I_{yy}	0.15	$\text{kg} \cdot \text{m}^2$
I_{zz}	0.18	$\text{kg} \cdot \text{m}^2$
μ	0.6	
f_{min}	5	N
f_{max}	150	N

V. RESULTS AND DISCUSSION

This section presents the results from simulation and hardware experiments of the Mini Cheetah using the proposed balance controller. The results demonstrate the comparative benefits of VBL-QP over MPC-based approaches as well as the broader advantages of weighing short- and long-term costs via approximation of the CCTG. Additional results are presented to validate that the approximation holds across the range of terrains the robot can expect to see as well across the robot's kinematic workspace.

A. SIMULATOR/EXPERIMENTAL SETUP

The balance control algorithm proposed in this paper is implemented both on the Mini Cheetah hardware [32] as well as in simulation via the open-source dynamic simulation software for the MIT Mini Cheetah developed in collaboration between MIT and Notre Dame.¹ For hardware experimentation, high-level control is executed on an UP Board low-power single-board computer with a quad-core Intel Atom CPU and 4 GB RAM, running Linux with the CONFIG_PREEMPT_RT patch for soft-realtime operation. Communication between the computer and actuators is carried out using a custom quad CAN bus interface. The actuator communication loops, as well as the control and state estimation loops run at 500 Hz.

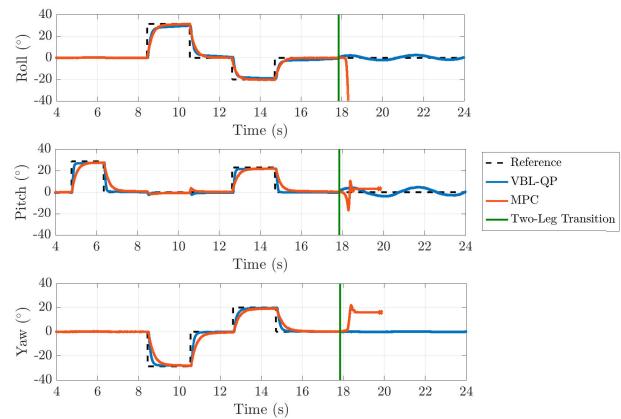
The dynamic simulation software simulates the full-body dynamics of the quadruped and is meant to mimic real-world experimentation as closely as possible. Similar to the hardware, information about the robot's state comes from a state estimator that decouples the estimation of position and velocity from orientation [33]. Uniformly distributed noise is introduced to the accelerometer, gyroscope, and quaternion readings with variances based on the specifications of the IMU. The simulator uses a time-stepping scheme with a hard contact model for the ground and treats the robot's feet as point contacts. The results presented in this paper include actuator models that account for motor inertia and torque/speed limits. The system parameters used in simulation are given by Table 1. The LQR gains used by the VBL-QP controller for all simulation and hardware testing are given by Table 2.

B. COMPARISON TO MPC

The primary advantage of the VBL-QP controller lies in its compact formulation. Tasks such as underactuated balancing

TABLE 2. LQR weighting matrices. Matrix Q corresponds to components of the state and matrix R to x, y, z components for each foot.

Parameter	Weight Vector	Units
Q_p	20 20 10000	$(\frac{1}{\text{m}})^2$
$Q_{\dot{p}}$	1 1 10	$(\frac{\text{s}}{\text{m}})^2$
Q_R	1100 1600 500	$(\frac{1}{\text{rad}})^2$
Q_{ω}	10 5 1	$(\frac{\text{s}}{\text{rad}})^2$
R	1.25 1.25 0.5	$(\frac{1}{\text{N}})^2$
β	2.25	$(\frac{1}{\text{N}})^2$

**FIGURE 4.** Simulated reference trajectory tracking for the VBL-QP and MPC controllers. The robot starts on four legs and after tracking orientation changes attempts to transition to two-leg balance.

on two legs are achievable using VBL-QP, but to the authors' best knowledge have never been demonstrated via conventional MPC-based approaches. The challenges to MPC-based control are twofold. The small base of support on two legs along with uncertainty from state estimation and imperfect modeling necessitates a sufficiently high control frequency to handle these uncertainties. Furthermore, underactuation requires suitably long prediction horizons to be considered to ensure that balance can be sustained indefinitely. Even when a reduced-order model is used, computational cost prevents solving the OCP (5) over suitably long prediction horizons in the requisite amount of time.

The compact formulation of the VBL-QP allows the controller to negotiate these challenges with greater success than MPC-based approaches. The VBL-QP controller was compared to the MIT Mini Cheetah's standard linear MPC locomotion controller [18]. The simulation results in Fig. 4 show that while both control approaches are able to successfully track reference trajectories on four legs, only the VBL-QP controller can handle the transition to two legs. Furthermore, Fig. 5 shows a comparison of the solve times corresponding to the motion in Fig. 4. Only solve times for the stable, four-leg portion of the MPC trajectory are shown in this figure. The transition from four legs to two, demonstrated on the hardware, is depicted in Fig. 6.

¹<https://github.com/mit-biomimetics/Cheetah-Software>

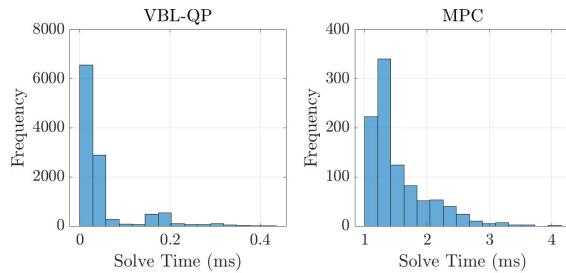


FIGURE 5. Solve times for the VBL-QP and MPC controller as the robot tracks a reference trajectory.

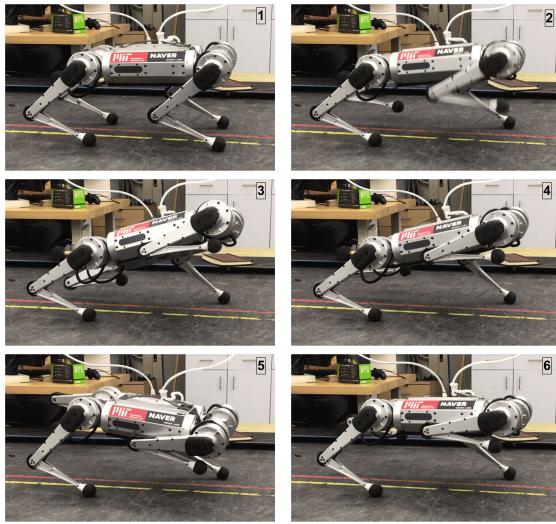


FIGURE 6. Transition from four-leg to two-leg balance using the VBL-QP controller.

C. REFERENCE TRACKING

A comparison between the ability of a conventional QP-based balance controller [25] and the VBL-QP controller to track reference trajectories on four legs illustrates the versatility of the controller, showing its effectiveness on four legs as well as two. The VBL-QP controller is able to track trajectories with comparable fidelity to the conventional controller [25]. Fig. 7 compares the set of achievable states as well as the position tracking error to reach those states for the two controllers following a range of step-input change commands for the desired position. The predictive nature of the VBL-QP controller affords it greater consistency over the range of desired states. To account for the sensitivity of the conventional controller to gain tuning, the results present two different sets of gains: (A), which maximizes the set of achievable states, and (B), which minimizes tracking error.

The set of achievable states for the two controllers is evaluated by measuring the approximate area of safely reachable $x - y$ positions for each controller. For the given motion, the approximate areas of the conventional (A), conventional (B), and VBL-QP were 5.65 cm^2 , 1.66 cm^2 , and 3.77 cm^2 , respectively. The larger area offered by the conventional (A) comes with the drawbacks of significantly higher tracking error as well as inconsistent performance. The overall

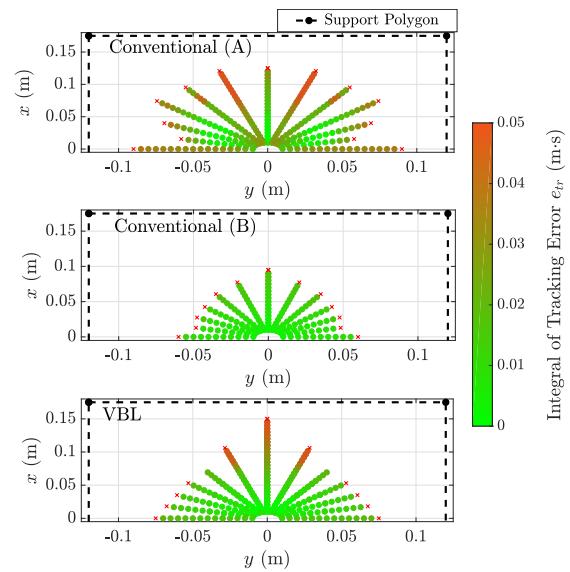


FIGURE 7. Points represents a desired change in xy position. Desired orientation of the body involved a 10° pitch and a yaw in the direction of the movement, max 17.5° .

position tracking error for states within 5cm of the starting position, measured according to

$$e_{tr} = \int_0^{t_{\text{settle}}} \|p - p^d\| dt, \quad (39)$$

was $1.39 \text{ cm}\cdot\text{s}$ for conventional (A), compared $0.64 \text{ cm}\cdot\text{s}$ for conventional (B) and $0.59 \text{ cm}\cdot\text{s}$ for VBL.

In addition to the increased error, the inconsistent performance of conventional (A) is highlighted by how the error does not strictly increase with displacement and how the tracked set points do not form the the neat half-oval that the other two do. The conventional (B) controller is able to improve consistency and lower the error, but at the cost of a 56% reduction in the size of achievable sets. So although gain dependencies make overall performance comparison between the controllers difficult, these results indicate that the VBL-QP is generally able to handle the trade-off between tracking error and achievable positions better than any set of gains attempted for the conventional QP.

D. OVERCOMING UNDERACTUATION

In addition to the challenge of maintaining balance over a dramatically reduced base of support, two-leg balance is further complicated by loss of actuation, which reduces set of external wrenches the robot can produce via ground reaction forces at the feet. Due to this loss of actuation, pure rolling and pitching are not possible for the robot's base. The predictive nature of the controller, however, is able to exploit the dynamics of the system, specifically by sacrificing short-term orientation, and achieve these desired orientation changes. The simulation and hardware results in Fig. 8 illustrate the response of the controller to a reference state calling for pure rolling of the body, and Fig. 9 shows the robot achieving desired pitches and rolls.

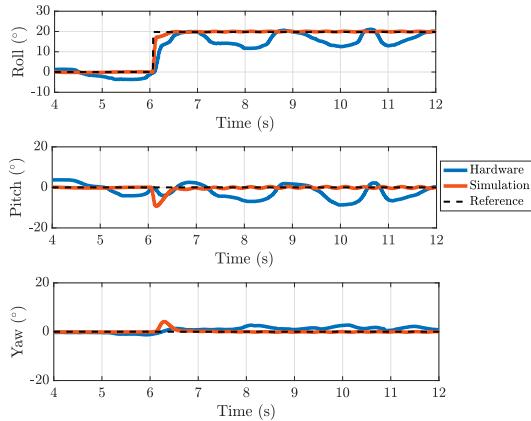


FIGURE 8. Simulation and hardware responses to desired 20° roll on two legs.

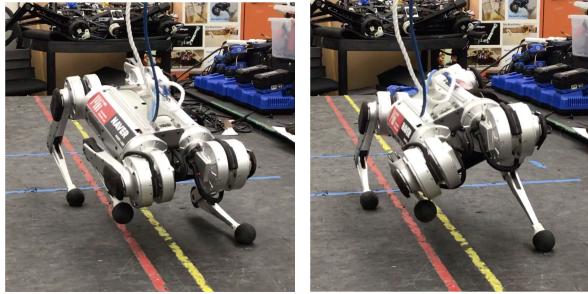


FIGURE 9. VBL-QP controller achieving desired pitching (left) and rolling (right) of the robot on two legs.

The results demonstrate the ability of the controller to handle underactuation. For example, when a pure rolling of the body is commanded, the controller accepts initial, short-term errors in pitch and yaw before settling the body into its desired state. This subtlety is best observed in the simulation results, which lack the noise seen in the hardware implementation. Nevertheless, for both simulation and hardware, the inclusion of the cost-to-go term in the optimization (8) makes these orientation changes possible.

E. PUSH DISTURBANCE RECOVERY

Robustness to perturbations is crucial for a successful balance controller. The optimal-control-based approach of the VBL-QP controller equips the robot to handle significant perturbations not only on four legs, but also on two legs. To measure the controller's disturbance robustness, the maximum instantaneous change in velocity that the robot could withstand without falling over was determined for each direction of velocity perturbation. The results of these simulation experiments are illustrated in Fig. 10.

In the direction of the FL and BR feet, the feet in contact when on two legs, the controller can handle roughly the same magnitude of kick on four legs and on two. In the direction of the FR and BL feet, however, the maximum tolerable kick magnitude when on two legs is 85% smaller

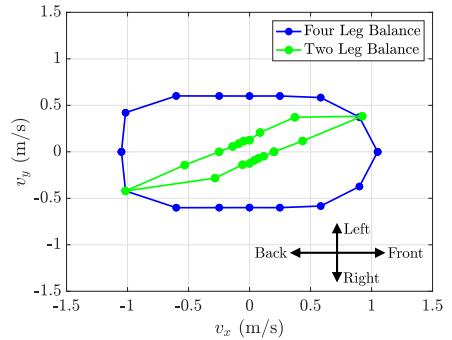


FIGURE 10. Maximum instantaneous change in velocity the robot can experience in each direction without falling.

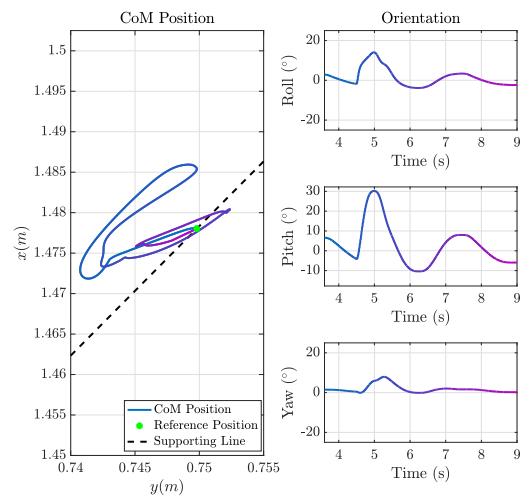


FIGURE 11. Mini Cheetah's response to 0.15 m s^{-1} change in x velocity and 0.05 m s^{-1} change in y while balancing on two legs. (Top) 2D projection of the CoM trajectory onto the ground plane. (Bottom) Corresponding orientation error as the robot recovers from push.

than for the four-leg case. Considering that any movement in this direction is purely outside of and away from the stable region, this basin of attraction still represents significant recovery capability. To more closely highlight the challenge of recovering from outside the stable region, Fig. 11 shows simulation results for the response of the controller to a single push disturbance on two legs. Notice that after the CoM is perturbed from the supporting line, the orientation of the robot is sacrificed in the short term so that the robot can return to a stable position. This ability to recover from unstable states is made possible by the controller's consideration of both immediate and future effects of control inputs. This consideration effectively allows the controller to recognize and avoid the control inputs that will lead to instability.

This robustness to perturbations was also demonstrated on the hardware. The controller was able to stabilize the robot after a series of manual pushes applied to different parts of the robot and in different directions. The results in Fig. 12 show the controller's response to one such push recovery test, and Fig. 13 provides a visualization of the robot recovering from a push.

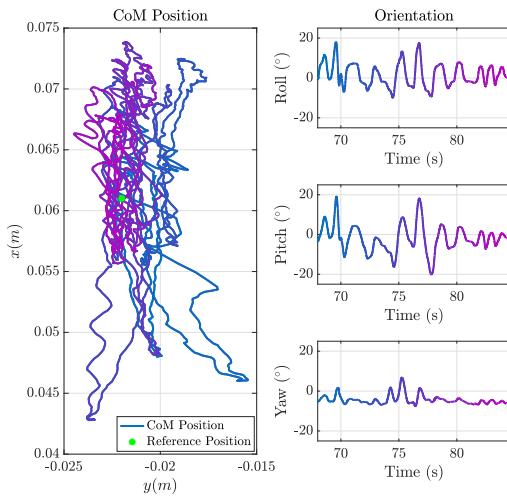


FIGURE 12. Mini Cheetah's response to two push disturbances (at $t = 68$ s and $t = 74$ s) while balancing on two legs. (Left) 2D projection of the CoM trajectory onto the ground plane. (Right) Corresponding angular velocity of the robot as it recovers from the pushes.

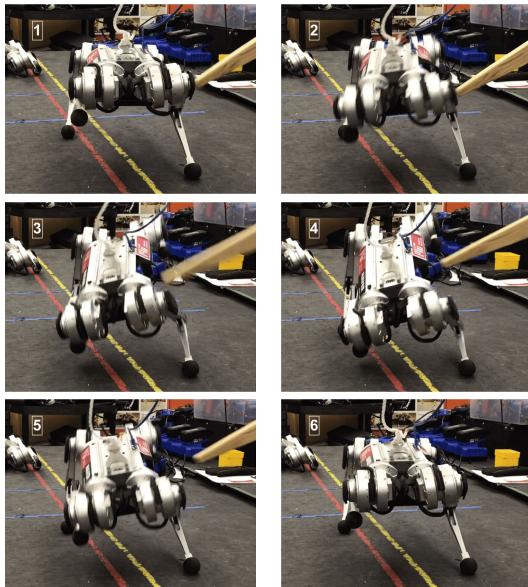


FIGURE 13. Recovery from push disturbance.

The hardware results are not able to as clearly show the controller's ability to sacrifice short orientation error to ensure stability because, unlike in simulation, it is difficult to impart a pure change in the linear velocity of the robot without also affecting its angular velocity. Regardless, these results show that with the VBL-QP controller implemented on the robot, it can withstand significant push disturbances while balancing on two legs.

F. REGION OF ATTRACTION

A critical component of the proposed balance controller is the QP implementation of the HJB minimization step that approximates the optimal control policy. This modified

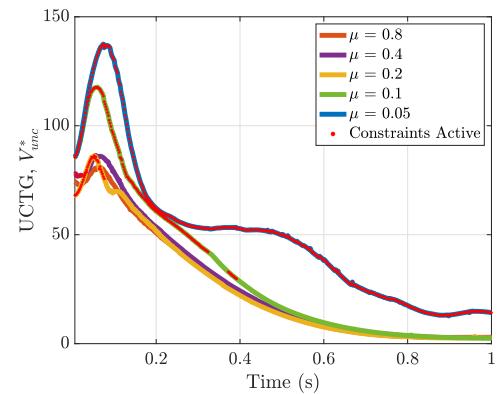


FIGURE 14. Simulated UCTG response to 20° roll and pitch and -20° yaw on two legs for various static coefficients of friction. Red dots indicate active friction constraints.

implementation is made possible by the substitution of the CCTG with the UCTG, a quantity easily computed by solving an algebraic Riccati equation. In the case of two-leg balance, the viability of this optimal value approximation is challenged by friction constraints and the limits of the dynamic linearization. In theory, one or both of these factors can prevent the UCTG from being a good predictor of long-term costs of staying balanced, rendering the controller ineffective. In practice however, we find that the approximated solution to the HJB equation used in the VBL-QP control law is robust enough that for friction coefficients as low as 0.05 (10% of nominal) and for nearly all references states within the robot's kinematic workspace, the controller is able to maintain balance.

We evaluate the impact of friction constraints by observing the response of the UCTG to a change in the desired reference state. In Fig. 14, the UCTG response to a desired roll, pitch, and yaw of the body with various coefficients of friction is presented. In the case of perfect modeling and an exact solution to the constrained OCP (5), the “ideal” cost-to-go would strictly decrease with time. The results in Fig. 14 show that as the coefficient of friction decreases and friction constraints are more frequently active throughout the robot's motion, the response deviates from this ideal response. However, despite the spikes in UCTG and the slower rates of decrease, the VBL-QP controller maintains the ability to reach the desired reference state.

In Section II.B, we briefly discussed the concept of region of attraction (ROA) as it relates to the linearization of the dynamics about a reference trajectory. We here present the effective region of attraction (ROA*) of the controller and demonstrate that it sufficiently covers the set of feasible reference states the robot can expect to encounter. The ROA* differs from the ROA in that the ROA* considers the effect of friction constraints and kinematic limits on the set of reachable states. In other words, the ROA provides the states about which the system's dynamics can be linearized with sufficient accuracy, while the ROA* provides the states that 1) can be linearized about with sufficient accuracy, 2) are

TABLE 3. Effective region of attraction, ROA*, for two-leg balance using VBL-QP controller.

	Kinematic Workspace		ROA* ($\mu = 0.6$)		ROA* ($\mu = 0.1$)	
	Min	Max	Min	Max	Min	Max
Roll	-43°	63°	-43.0°	63°	-43.0°	54.4°
Pitch	-31.5°	25.8°	-31.5°	25.8°	-31.5°	25.8°
Yaw	-48.7°	31.5°	-48.7°	28.6°	-48.7°	31.5°
Roll, Pitch	75%	80%	75%	75%	75%	70%
Roll, Yaw	85%	85%	85%	75%	85%	75%
Pitch, Yaw	75%	85%	75%	80%	75%	80%

kinematically feasible, and 3) can be reached in the presence of friction constraints starting from the base state. The size of the ROA* validates that the approximations used in the VBL-QP controller, specifically the linearized dynamics and the cost-to-go approximation, are viable.

The ROA* for two-leg balance of the Mini Cheetah was evaluated in simulation for two different friction coefficients, $\mu = 0.6$ and $\mu = 0.1$. To reiterate, a reference state is considered to be in the ROA* if the robot can start at its base position, linearize about that desired reference state, and then successfully reach the reference state. Simple orientation changes like roll, pitch, and yaw are quantified in terms of degrees, while combinations of the rotations (roll+pitch, roll+yaw, pitch+yaw) are quantified by percent of maximum rotation. For example, saying that the ROA* for maximum roll and pitch is 75% is equivalent to saying the robot can successfully perform a combined roll and pitch of 75% its maximum kinematically feasible roll and 75% its maximum kinematically feasible pitch. The results presented in Table 3 show that the ROA*'s for both the nominal and reduced friction case span well over the majority of the kinematic workspace.

VI. CONCLUSION

We have presented a novel variational-based optimal controller for underactuated dynamic balancing of a quadruped and demonstrated its viability in simulation and experiments through two-leg balance on point feet. The application of VBL to quadruped dynamics enabled the use of optimal control strategies for computing the UCTG of the system. The inclusion of this UCTG in the objective function of a QP that replicates the minimization step of the HJB equation sufficiently captures future effects of present control inputs. This crucial predictive element of the controller allows it to navigate complex trade-offs between short-term tracking and long-term balance. The power of this approach is demonstrated through experiments of two-leg balance that show robustness to perturbations and the ability to overcome underactuation via exploitation of angular momentum. Finally, an analysis of the cost-to-go response for various environmental conditions and desired motions shows that the the linearized model and HJB minimization are valid for the terrains and tasks that the Mini Cheetah can expect to encounter. More broadly, the extension of our approaches to other legged systems, such as biped and humanoids, presents

future opportunity to address the coordination of center of mass and body posture as a mechanism to enhance balance control.

REFERENCES

- [1] Y. Abe, M. Da Silva, and J. Popović, “Multiobjective control with frictional contacts,” in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, Aug. 2007, pp. 249–258.
- [2] J. Park, J. Haan, and F. C. Park, “Convex optimization algorithms for active balancing of humanoid robots,” *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 817–822, Aug. 2007.
- [3] P. M. Wensing and D. E. Orin, “Generation of dynamic humanoid behaviors through task-space control with conic optimization,” in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 3103–3109.
- [4] A. Escande, N. Mansard, and P.-B. Wieber, “Hierarchical quadratic programming: Fast online humanoid-robot motion generation,” *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 1006–1028, Jun. 2014.
- [5] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, “Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 981–988.
- [6] C. Ott, M. A. Roa, and G. Hirzinger, “Posture and balance control for biped robots based on contact force optimization,” in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, Oct. 2011, pp. 26–33.
- [7] B. J. Stephens and C. G. Atkeson, “Dynamic balance force control for compliant humanoid robots,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 1248–1255.
- [8] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, “Optimal distribution of contact forces with inverse-dynamics control,” *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 280–298, Mar. 2013.
- [9] M. Hutter, M. A. Hoepflinger, C. Gehring, M. Bloesch, C. D. Remy, and R. Siegwart, “Hybrid operational space control for compliant legged systems,” in *Robotics: Sci. Syst.*, 2008.
- [10] 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, Jun. 2000.
- [11] M. W. Spong, “The swing up control problem for the acrobot,” *IEEE Control Syst. Mag.*, vol. 15, no. 1, pp. 49–55, Feb. 1995.
- [12] Y. Tassa, T. Erez, and E. Todorov, “Synthesis and stabilization of complex behaviors through online trajectory optimization,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4906–4913.
- [13] M. Neunert, F. Farshidian, A. W. Winkler, and J. Buchli, “Trajectory optimization through contacts and automatic gait discovery for quadrupeds,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1502–1509, Jul. 2017.
- [14] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 3346–3351.
- [15] Y. Tassa, N. Mansard, and E. Todorov, “Control-limited differential dynamic programming,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 1168–1175.
- [16] G. I. Boutselis and E. Theodorou, “Differential dynamic programming on lie groups: Derivation, convergence analysis and numerical results,” 2018, *arXiv:1809.07883*. [Online]. Available: <http://arxiv.org/abs/1809.07883>
- [17] M. Diehl, H. J. Ferreau, and N. Haverbeke, “Efficient numerical methods for nonlinear MPC and moving horizon estimation,” in *Nonlinear model Predictive Control*. Cham, Switzerland: Springer, 2012, pp. 391–417.
- [18] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–9.
- [19] Y. Ding, A. Pandala, and H.-W. Park, “Real-time model predictive control for versatile dynamic motions in quadrupedal robots,” in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 8484–8490.
- [20] G. Bledt, P. M. Wensing, and S. Kim, “Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the MIT cheetah,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 4102–4109.
- [21] S. Kuindersma, F. Permenter, and R. Tedrake, “An efficiently solvable quadratic program for stabilizing dynamic locomotion,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2589–2594.

[22] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, Nov. 2001, pp. 239–246.

[23] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Biped walking pattern generation by using preview control of zero-moment point,” in *Proc. IEEE Int. Conf. Robot. Autom.*, Taipei, Taiwan, Sep. 2003, pp. 1620–1626.

[24] P.-B. Wieber, “Trajectory free linear model predictive control for stable walking in the presence of strong perturbations,” in *Proc. 6th IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2006, pp. 137–142.

[25] M. Focchi, A. del Prete, I. Havoutis, R. Featherstone, D. G. Caldwell, and C. Semini, “High-slope terrain locomotion for torque-controlled quadruped robots,” *Auto. Robots*, vol. 41, no. 1, pp. 259–272, Jan. 2017.

[26] G. Wu and K. Sreenath, “Variation-based linearization of nonlinear systems evolving on $SO(3)$ and S^2 ,” *IEEE Access*, vol. 3, pp. 1592–1604, 2015.

[27] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, no. 2. Belmont, MA, USA: UAthena scientific, 1995.

[28] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[29] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit linear quadratic regulator for constrained systems,” *Automatica*, vol. 38, no. 1, pp. 3–20, Jan. 2002.

[30] P. O. M. Scokaert and J. B. Rawlings, “Constrained linear quadratic regulation,” *IEEE Trans. Autom. Control*, vol. 43, no. 8, pp. 1163–1169, Aug. 1998.

[31] P. Grieder, F. Borrelli, F. Torrisi, and M. Morari, “Computation of the constrained infinite time linear quadratic regulator,” *Automatica*, vol. 40, no. 4, pp. 701–708, Apr. 2004.

[32] B. Katz, J. D. Carlo, and S. Kim, “Mini cheetah: A platform for pushing the limits of dynamic quadruped control,” in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6295–6301.

[33] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, “MIT cheetah 3: Design and control of a robust, dynamic quadruped robot,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 2245–2252.

[34] P. M. Wensing, A. Wang, S. Seok, D. Otten, J. Lang, and S. Kim, “Proprioceptive actuator design in the MIT cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots,” *IEEE Trans. Robot.*, vol. 33, no. 3, pp. 509–522, Jun. 2017.

[35] R. Featherstone, *Rigid Body Dynamics Algorithms*. New York, NY, USA: Springer, 2014.

[36] T. Lee, M. Leok, and N. H. McClamroch, “Stable manifolds of saddle equilibria for pendulum dynamics on S^2 and $SO(3)$,” in *Proc. IEEE Conf. Decis. Control Eur. Control Conf.*, Dec. 2011, pp. 3915–3921.



MATTHEW CHIGNOLI received the bachelor’s degree in mechanical engineering from the University of Notre Dame, in 2019. He is currently a Graduate Student with the Mechanical Engineering Department, Massachusetts Institute of Technology. His undergraduate research in the Robotics, Optimization, and Assistive Mobility (ROAM) Lab focused on balance control of the MIT Cheetah 3 quadruped. His research in the MIT Biomimetic Robotics Lab is aimed at developing control strategies for highly dynamic locomotion of quadruped robots.



PATRICK M. WENSING (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from The Ohio State University, Columbus, OH, USA, in 2009, 2013, and 2014, respectively. He is currently an Assistant Professor with the Department of Aerospace and Mechanical Engineering, University of Notre Dame, where he directs the Robotics, Optimization, and Assistive Mobility (ROAM) Lab. Before joining the University of Notre Dame, he was a Postdoctoral Associate at MIT working on control system design for the MIT Cheetah robots. His research interests focus on the intersection of dynamics, optimization, and control as applied to agile, intelligent, and physically interactive robotic systems. He received the NSF Graduate Research Fellowship for his dissertation work on balance control strategies for humanoid robots. He also serves as an Associate Editor for the IEEE TRANSACTIONS ON ROBOTICS and a Co-Chair for the IEEE RAS Technical Committee on Model-Based Optimization for Robotics.

• • •