Enhanced Balance for Legged Robots Using Reaction Wheels

Chi-Yen Lee[1*], Shuo Yang[1*], Benjamin Bokser[2], and Zachary Manchester[1]

***Abstract*— We introduce a reaction wheel system that enhances the balancing capabilities and stability of quadrupedal robots during challenging locomotion tasks. Inspired by both the standard centroidal dynamics model common in legged robotics and models of spacecraft commonly used in the aerospace community, we model the coupled quadruped-reaction-wheel system as a gyrostat, and simplify the dynamics to formulate the problem as a linear discrete-time trajectory optimization problem. Modifications are made to a standard centroidal model-predictive control (MPC) algorithm to solve for both stance foot ground reaction forces and reaction wheel torques simultaneously. The MPC problem is posed as a quadratic program and solved online at 1000 Hz. We demonstrate improved attitude stabilization both in simulation and on hardware compared to a quadruped without reaction wheels, and perform a challenging traversal of a narrow balance beam that would be impossible for a standard quadruped. A video of our experiments is available online[1].**

I. Introduction

The core design of quadrupedal robots has largely converged over the past decade: High-performance contemporary designs share many similarities, including a rigid torso, four three-degree-of-freedom (3-DOF) legs, and a rounded "point foot" at the end of each leg. While simple and effective, this configuration is highly underactuated during locomotion [11]. Quadrupedal robots lose rotational control authority entirely during flight phases, and around the line of support when only two feet are in contact with the ground. With this configuration, large body orientation errors can only be eliminated by switching stance-foot configurations [11, 2], and a quadrupedal robot becomes especially vulnerable to impact and disturbance during the two-foot standing phase.

In contrast, terrestrial animals use a multitude of strategies to perform inertial stabilization as they perform dynamic movements: Humans heavily regulate their angular momentum during locomotion through arm movements [22], cheetahs have been observed using their tails during high-speed chases and turning maneuvers [21], and falling cats are able to adjust their attitude during falls using their highly flexible spines [17]. To bring similar capabilities to quadrupedal robots with a minimum of additional hardware and control complexity, we take inspiration from the aerospace industry: Reaction wheel actuators (RWAs) are widely used on satellites to perform attitude control [8]. We develop a proof-of-concept RWA module that can be attached to the back of a standard Unitree A1 robot to provide additional angular momentum control. The 4.3-kg module, shown in Figure 1, is compact, reusable, and has high control bandwidth.

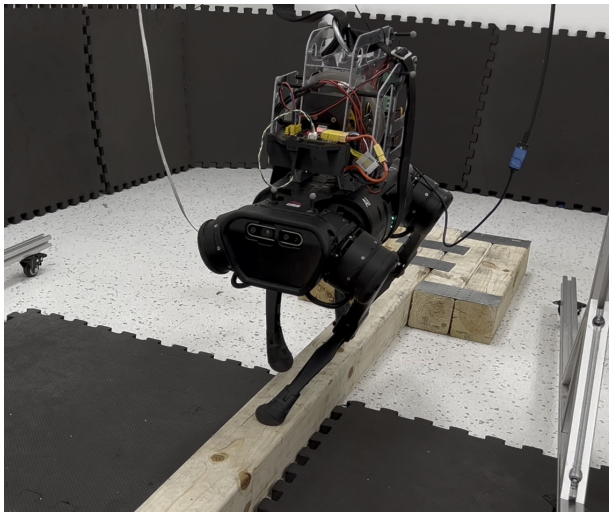


Fig. 1: The Unitree A1 Quadruped walking on a six-centimeter-wide beam with the assistance of our reaction wheel actuator system and gyrostat MPC controller.

RWAs offer a number of advantages over other mechanical appendages such as tails or multi-link limbs: Controllers do not need to consider the non-trivial collision-avoidance problem between appendages and the robot chassis. And, perhaps most importantly, RWAs lead to linearized dynamics that can be easily integrated into the standard centroidal model commonly used in MPC controllers for quadrupeds [2, 8]. One major drawback of RWAs is their reliance on wheel acceleration to provide body torque control, which can lead to saturation of rotor speed. In this paper, we show that this limitation can be handled effectively by adding a set of linear inequality constraints in the MPC formulation, and a linear feedback term on reaction wheel momentum.

The contributions of this work include:

- Design and construction of a two-axis RWA system for quadrupeds
- A convex gyrostat MPC formulation that leverages the RWAs to improve disturbance rejection
- Demonstration of enhanced stability and new locomotion capabilities using RWAs on a Unitree A1 robot
- The first demonstration of narrow-beam walking performed by a quadruped robot

The paper proceeds as follows: Section II discusses previous works related to inertial appendages and reaction wheels; Section III talks about the background ideas that we build

Authors are with [1] the Robotics Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA; and [2] the Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA

*These authors contributed equally to this paper.

[1] https://youtu.be/UroSaUg3Q6k

upon in this paper; Section IV presents the mechanical design of the RWA module; Section V introduces the gyrostat MPC algorithm; and Section VI presents our simulated and hardware results.

II. Related Work

In this section, we discuss some previous research results on quadruped balance control, related works on inertial appendages, and existing works on integrating RWAs into legged robot locomotion.

A. Legged Robot Balance Strategies

Without inertial appendages, the classic optimal control approach uses MPC to perform body balancing using only ground reaction forces [7, 2, 1, 18, 6]. While these approaches have demonstrated incredible dynamic hardware behavior (including running, galloping, and jumping), quasi-static motion such as balancing in a highly under-actuated pose remains difficult. In [5], the authors balance a quadruped in a two-leg under-actuated pose. In [10], the authors demonstrate the same behavior with careful modeling of the robot in the balance pose and demonstrate narrow-beam walking in simulation. However, to the best of our knowledge, no hardware demonstration of continuous locomotion on a narrow beam — where the support polygon is nearly empty — has been achieved before. In addition to the control problem itself, hardware challenges such as deformable feet make this difficult to realize in practice [25].

B. Tail Appendages

Inertial appendages have been widely explored for improving the stability of legged robots. Tails, which can allow robots to stabilize their attitude in mid-air without ground contact, have received significant attention [16]. In [13, 21], the authors designed tails for hexapod and wheeled robots to perform aerial reorientation maneuvers. In [19], the authors leveraged the aerodynamics of tails. In [26], the authors introduced a sequential distributed MPC framework to stabilize legged robot locomotion with tails in simulation. In comparison to RWAs, tails can be more lightweight and energetically efficient [16]. However, RWAs offer simpler linearized dynamics and, unlike tail-based systems, do not encounter self collisions or joint limits.

C. Reaction Wheels

There are numerous examples of adding RWAs to robots. Arguably one of the most well-known examples is the Cubli robot [9], which solely uses reaction wheels to achieve a variety of dynamic behaviors. RWAs have also been installed on bipeds to improve walking efficiency [3, 4] and to stabilize pitch during high-speed running [20]. On quadrupeds, RWAs have been studied for potential use in microgravity on the moon to perform attitude stabilization during jumping [15]. Most prior work on RWAs use simple proportional-derivative (PD) control laws to perform attitude stabilization, and do not reason about cross-coupling between the leg actuators and RWAs. In contrast, we present a new convex MPC algorithm that simultaneously optimizes both leg and RWA control inputs.

III. Background

This sections describes the Euler angle attitude representation that we use in this paper, the classic centroidal MPC framework for quadrupeds that we use as a starting point for our MPC formulation, and the gyrostat model that we use to incorporate RWA dynamics into the controller.

A. Euler Angles

In this paper, we parameterize the robot's orientation using the Tait–Bryan angles. Specifically, we use $\theta = [\theta_r; \theta_p; \theta_y]$ to represent orientation, where θ_r, θ_p, θ_y are commonly referred as roll, pitch, and yaw angles, respectively. The rotation matrix R can defined as

$$\mathrm{R} = \mathrm{R}_z(\theta_y)\mathrm{R}_y(\theta_p)\mathrm{R}_x(\theta_r), \tag{1}$$

where $\mathrm{R}_i(\alpha)$ represents a rotation of α degrees about the i-axis. The time derivative of the Euler angles $\dot{\theta}$ is related to the robot body angular velocity ${}^B\omega$ through the following relationship

$$\dot{\theta} = \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_p \\ \dot{\theta}_y \end{bmatrix} = \Omega(\theta)\,{}^B\omega = \begin{bmatrix} c_y/c_p & s_y/c_p & 0 \\ -s_y & c_y & 0 \\ (c_y s_p)/c_p & (s_y s_p)/c_p & 1 \end{bmatrix} {}^B\omega, \tag{2}$$

where we denote $\cos(\theta_i) = c_i$ and $\sin(\theta_i) = s_i$ for $i \in \{r, p, y\}$. Under stable walking conditions, we assume that c_p will never reach zero. Assuming the pitch and roll angles are small, $\Omega(\theta)$ simplifies to

$$\mathrm{R}_z(\theta) = \begin{bmatrix} c_y & s_y & 0 \\ -s_y & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

We note that the control approach in this paper is not restricted to Euler angles, and could be implemented with any other attitude representation, including quaternions [12].

B. Centroidal MPC

Our controller is built on the widely used convex centroidal MPC framework [2, 7]. This approach decouples the controller into two sub-problems: contact planning and body control. The contact planning problem calculates the the foot step location $p_i \in \mathcal{R}^{3\times 1}$ for foot $i \in 1 \ldots n$. In our work, the desired foot step locations p_i^d are determined with the Raibert heuristic [23] such that

$$p_i^d = \bar{p}_i + {}^N v \Delta t, \tag{4}$$

where $\bar{p}_i$ is the nominal foot position i on the ground beneath the corresponding hip i, Δt is the time that the foot will spend on the ground, and ${}^N v \in \mathcal{R}^{3\times 1}$ is the center of mass velocity in the inertial frame. Foot position p_i is then tracked to p_i^d with an end-effector PID controller [7].

The body control problem uses an MPC controller with a simplified model to achieve fast solution times. This simplified model, called the *centroidal model*, neglects the inertia of the robot's legs and treats the robot as a single

rigid body subjected to ground reaction forces produced by the feet. Given a body mass m and body moment of inertia expressed in the body frame ${}^{B}\mathrm{J}_b \in \mathcal{R}^{3\times3}$, the governing equations of the centroidal model are

$$\dot{x} = \begin{bmatrix} \dot{r} \\ \dot{\theta} \\ {}^{N}\ddot{r} \\ {}^{B}\dot{\omega} \end{bmatrix} = \begin{bmatrix} {}^{N}v \\ \Omega(\theta)\,{}^{B}\omega \\ \frac{1}{m}\,{}^{N}F - g \\ ({}^{B}\mathrm{J}_b)^{-1}({}^{B}\tau_f - {}^{B}\omega \times {}^{B}\mathrm{J}_b\,{}^{B}\omega) \end{bmatrix}, \quad (5)$$

where the state of the system includes of the center of mass position $r \in \mathcal{R}^{3\times1}$, Euler angle representation of the body attitude $\theta \in \mathcal{R}^{3\times1}$, inertial-frame linear velocity ${}^{N}v \in \mathcal{R}^{3\times1}$, and body-frame angular velocity ${}^{B}\omega \in \mathcal{R}^{3\times1}$. The input of the system contains an inertial frame force input ${}^{N}F \in \mathcal{R}^{3\times1}$ and a body frame torque input ${}^{B}\tau_f \in \mathcal{R}^{3\times1}$ due to ground reaction forces. The input forces and torques are mapped from the ground reaction forces $f_i = [f_x, f_y, f_z]^T$ generated by each foot at position p_i through the following relationship:

$$u = \begin{bmatrix} {}^{N}F \\ {}^{B}\tau_f \end{bmatrix} = \begin{bmatrix} I_3 & \dots & I_3 \\ \mathrm{R}^T[p_1]^{\times} & \dots & \mathrm{R}^T[p_n]^{\times} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}. \quad (6)$$

where $[*]^{\times}$ is the skew-symmetric cross-product matrix, and $I_n \in \mathcal{R}^{n\times n}$ denotes an n by n identity matrix. The ground reaction forces are further subjected to friction-cone constraints to prevent foot slip. Often, the friction-cone constraint is approximated as a pyramid, which enables the constraints to be expressed as a set of linear inequalities:

$$\begin{aligned} -\mu f_z \leq & f_x \leq \mu f_z \\ -\mu f_z \leq & f_y \leq \mu f_z. \end{aligned} \quad (7)$$

With the model described in (5) and (6), the MPC controller enforces the dynamics as a set of constraints in a trajectory optimization problem that is solved online. In this paper, the problem is convexified and formulated as a quadratic program (QP) similar to [7].

C. Gyrostat Dynamics

The dynamics of a rigid body with RWAs can be modeled as a gyrostat [8]. A gyrostat is a system of coupled rigid bodies whose relative motions do not change the total inertia tensor of the system. A set of k reaction wheels with perfect static and dynamic balance have a constant inertia in the body frame of the robot that can be expressed as

$${}^{B}\mathrm{J}_r = \sum_{i=1}^{k} {}^{B}\mathrm{L}_i \quad (8)$$

where ${}^{B}\mathrm{L}_i \in \mathcal{R}^{3\times3}$ is the inertia of the i-th wheel expressed in the robot's body frame. The total angular momentum of the wheels in the body frame of the robot is

$$\mathrm{h}_r = \sum_{i=1}^{k} {}^{B}\mathrm{L}_i({}^{B}\omega + {}^{B}\psi_i), \quad (9)$$

where ${}^{B}\omega \in \mathcal{R}^{3\times1}$ is the body angular velocity of the robot and ${}^{B}\psi_i \in \mathcal{R}^{3\times1}$ is the angular velocity of wheel i in the robot's body frame. The total angular momentum of the robot ${}^{B}\mathrm{H}$ is then

$$\begin{aligned} {}^{B}\mathrm{H} &= {}^{B}\mathrm{J}_\mathrm{b}\,{}^{B}\omega + \mathrm{h}_r & (10) \\ &= ({}^{B}\mathrm{J}_\mathrm{b} + {}^{B}\mathrm{J}_\mathrm{r})^{B}\omega + \sum_{i=1}^{k} {}^{B}\mathrm{L}_i\,{}^{B}\psi_i. & (11) \end{aligned}$$

Taking the derivative of the angular momentum yields the equation of motion for the gyrostat,

$${}^{B}\tau_f = {}^{B}\mathrm{J}\,{}^{B}\dot{\omega} + {}^{B}\omega \times ({}^{B}\mathrm{J}\,{}^{B}\omega + \sum_{i=1}^{k} {}^{B}\mathrm{L}_i\,{}^{B}\psi_i) + {}^{B}\,\tau_r, \quad (12)$$

where ${}^{B}\mathrm{J} \in \mathcal{R}^{3\times3}$ is the sum of ${}^{B}\mathrm{J}_r$ and ${}^{B}\mathrm{J}_b$, ${}^{B}\tau_f \in \mathcal{R}^{3\times1}$ is the net body-frame external torque on the robot, and ${}^{B}\tau_r \in \mathcal{R}^{3\times1}$ is the net body-frame torque exerted by the RWAs. The wheel-frame angular momentum $\rho \in \mathcal{R}^{k\times1}$ can be mapped to this equation with a constant jacobian matrix $\Lambda_r \in \mathcal{R}^{k\times3}$ and the input torques along the wheel axis $u_r \in \mathcal{R}^{k\times1}$ such that

$$\begin{aligned} \sum_{i=1}^{k} {}^{B}\mathrm{L}_i\,{}^{B}\psi_i &= \Lambda_r^T\rho & (13) \\ {}^{B}\tau_f &= {}^{B}\mathrm{J}\dot{\omega} + {}^{B}\omega \times ({}^{B}\mathrm{J}\,{}^{B}\omega + \Lambda_r^T\rho) + \Lambda_r^T u_r. & (14) \end{aligned}$$

This can be written in the standard manipulator form,

$$M\dot{v} + C(q, v) = B(q)u \quad (15)$$

where the mass matrix M, dynamic bias $C(q, v)$, and control mapping $B(q)$ for configuration q, velocity v, and control u are defined as

$$\underbrace{\begin{bmatrix} {}^{B}\mathrm{J} & 0 \\ 0 & I_k \end{bmatrix}}_{M} \begin{bmatrix} {}^{B}\dot{\omega} \\ \dot{\rho} \end{bmatrix} + \underbrace{\begin{bmatrix} {}^{B}\omega \times ({}^{B}\mathrm{J}\,{}^{B}\omega + \Lambda_r^T\rho) \\ 0 \end{bmatrix}}_{C(q,v)} = \underbrace{\begin{bmatrix} \Lambda_r^T \\ I_k \end{bmatrix}}_{B(q)} \begin{bmatrix} u_r \end{bmatrix}. \quad (16)$$

IV. Hardware Design

This section presents the specifics of the mechanical design of the RWA module, which is pictured in Fig. 2. We use only two RWAs, one for the pitch axis and one for the roll axis. An RWA for the yaw axis is not included as yaw control is much less important for stability of the robot. The RWAs are driven by two permanent-magnet synchronous motors, each with a continuous maximum current draw of 60A, a maximum speed of 1900 RPM, and a maximum torque output of 4.7 Nm. The chassis is composed of polycarbonate plates and 3D-printed polylactic acid (PLA). The overall system has dimensions of 100 × 210 × 300 mm and a total mass of 4.3 kg, including a 2200 mAh lithium polymer battery pack. Variables such as efficiency, weight, and dimension were not optimized for this prototype. As demonstrated in Section VI, despite the increase in total mass and the change in inertial properties caused by the addition of the RWA module, the module still significantly improves the stabilizing capabilities of the robot.

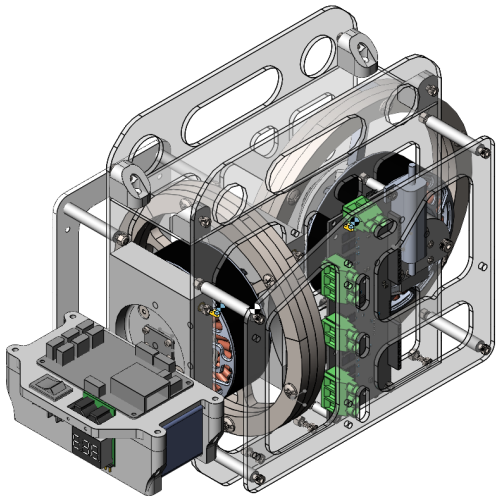


Fig. 2: Proof-of-concept RWA system that mounts directly to the back of a Unitree A1 robot. The system includes two permanent-magnet synchronous motors that drive two high-inertia flywheels along the roll and pitch axes up to a maximum speed of 1900 RPM. The system contains its own battery and is capable of generating a maximum continuous torque of 5 Nm along each axis.

V. Gyrostat MPC

This section describes an MPC framework that jointly solves for RWA torques and ground-reaction forces for a quadruped equipped with our RWA module. We incorporate the RWA dynamics into the centroidal quadruped controller using the gyrostat model, and we linearize the model to use it as a part of a convex discrete-time trajectory optimization problem.

A. Gyrostat Quadruped Dynamics

Our RWA module adds two RWAs that provide control over the body angular momentum in the pitch and roll axes. Combining the centroidal dynamics from (5) and the RWA rotational dynamics from (16), we obtain the following equations of motion:

$$\dot{x} = \begin{bmatrix} \dot{r} \\ \dot{\theta} \\ {}^N\ddot{r} \\ {}^B\dot{\omega} \\ \dot{\rho} \end{bmatrix} = \begin{bmatrix} {}^N v \\ \Omega(\theta)\,{}^B\omega \\ \frac{1}{m}\,{}^N F - g \\ ({}^B\mathrm{J})^{-1}({}^B\tau_f + \Lambda_r^T u_r - {}^B\omega \times ({}^B\mathrm{J}\omega + \Lambda_r^T \rho)) \\ u_r \end{bmatrix}, \tag{17}$$

where, in addition to the original centroidal model states, we introduce the RWA momentum state vector $\rho \in \mathcal{R}^{2\times 1}$ and torque input vector $u_r \in \mathcal{R}^{2\times 1}$. The control input vector $u \in \mathcal{R}^{14\times 1}$ for the whole system is now expressed as:

$$u = \begin{bmatrix} {}^N F \\ {}^B\tau_f \\ u_r \end{bmatrix} = \begin{bmatrix} I_3 & \dots & I_3 & 0_{32} \\ \mathrm{R}^T[p_1]_\times & \dots & \mathrm{R}^T[p_n]^\times & 0_{32} \\ 0_{23} & \dots & 0_{23} & I_2 \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ u_r \end{bmatrix}. \tag{18}$$

where 0_{nm} represents a $n \times m$ matrix with all zero elements, and 0_n represents a $n \times n$ square matrix with all zero elements. We now linearize (17) for a convex MPC formulation assuming that attitude displacement, body angular velocity, and RWA velocities are small during normal operating conditions. This allows us to parameterize rotation using (3) and simplify the rotational kinematics and dynamics from (17) to

$$\begin{aligned} \dot{\theta} &\approx \mathrm{R}_z^T(\theta_y)\,{}^B\omega \\ {}^B\dot{\omega} &\approx ({}^B\mathrm{J})^{-1}({}^B\tau_f + u_r). \end{aligned} \tag{19}$$

Finally, we convert the angular velocities into the inertial frame, and the resulting linearized dynamics for the gyrostat quadruped model becomes

$$\frac{d}{dt}\begin{bmatrix} r \\ \theta \\ {}^N\dot{r} \\ {}^N\omega \\ \dot{\rho} \end{bmatrix} = \begin{bmatrix} 0_3 & 0_3 & I_3 & 0_3 & 0_{32} \\ 0_3 & 0_3 & 0_3 & \mathrm{R}_z^T(\theta_y) & 0_{32} \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_{32} \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_{32} \\ 0_{23} & 0_{23} & 0_{23} & 0_{23} & 0_2 \end{bmatrix} \begin{bmatrix} r \\ \theta \\ {}^N\dot{r} \\ {}^N\omega \\ \rho \end{bmatrix} + \begin{bmatrix} 0_{31} \\ 0_{31} \\ g \\ 0_{31} \\ 0_{21} \end{bmatrix} + \begin{bmatrix} 0_3 & \dots & & 0_{32} \\ 0_3 & \dots & & 0_{32} \\ \frac{I_3}{m} & \dots & \frac{I_3}{m} & 0_{32} \\ ({}^N\mathrm{J})^{-1}[p_1]_\times & \dots & ({}^N\mathrm{J})^{-1}[p_n]_\times & ({}^N\mathrm{J})^{-1}\mathrm{R}^T\Lambda_r^T \\ 0_{23} & \dots & & I_2 \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ u_r \end{bmatrix}, \tag{20}$$

where $g \in \mathcal{R}^{3\times 1}$ represents the gravity vector. The dynamics is now in a linearized form parameterized by the yaw angle θ_y and foot positions p_i such that

$$\dot{x}(t) = A(\theta_y)x(t) + B(p_1, \dots, p_n, \theta_y)u(t). \tag{21}$$

B. Trajectory Optimization Problem

The problem is now linearized with the continuous-time transition and control matrices A and B. We convert those matrices into discrete-time $\mathcal{A}$ and $\mathcal{B}$ matrices. This control problem is then posed as a discrete-time convex quadratic program as follows:

$$\min_{x,u} \quad \sum_0^{l-1} \|x_{i+1}^d - x_{i+1}\|_{Q_i}^2 + \|u_i\|_{R_i}^2 \tag{22a}$$

$$\text{subject to} \quad x_{i+1} = \mathcal{A}_i x_i + \mathcal{B}_i u_i, i = 0 \dots l-1 \tag{22b}$$

$$\underline{c}_i \le C_i u_i \le \bar{c}_i, i = \dots l-1 \tag{22c}$$

$$D u_i = 0, i = 0 \dots l-1, \tag{22d}$$

where $x_i, u_i \in \mathcal{R}^{14\times 1}$, $Q_i, R_i \in \mathcal{R}^{14\times 14}$ are the state of the robot, control inputs to the robot, and cost matrices for state and control inputs at time step i, respectively. The matrices C_i in (22c) are used to enforce linearized friction cone constraints from (7). We also regularize the speed of the RWAs inside the dynamics penalty term and constrain the RWA torques and speeds with the affine constraints

$$\underline{u}_r \le u_r \le \bar{u}_r \tag{23}$$

$$\underline{\rho} \le \rho_i \le \bar{\rho}, i = \dots l-1, \tag{24}$$

where $\bar{u}_r$, $\underline{u}_r$, $\bar{\rho}_r$, $\underline{\rho}_r$ represents the upper bound and lower bound of the RWAs control torque and momentum, respectively. The equality constraints in (22d) are used to constrain foot forces to be zero when a foot is in the swing phase. The solution to the QP (22) returns the ground reaction forces for each of the feet in contact with the ground and the torques to be applied to the RWAs. We convert the ground reaction forces into joint torques using,

$$\tau_i = \Lambda_i^T \mathrm{R}(\theta)^T f_i, \tag{25}$$

where $\tau_i \in \mathcal{R}^{3\times1}$, $\Lambda_i \in \mathcal{R}^{3\times3}$, and $f_i \in \mathcal{R}^{3\times1}$ are the joint torques, forward kinematic jacobian, and ground reaction forces for leg i, respectively.

C. Angular Momentum Error Feedback

During online implementation, we introduce an error feedback term on the angular momentum of the roll RWA such that

$$\theta_r^d = \bar{\theta}_r^d + k_r \rho_r, \tag{26}$$

where $\bar{\theta}_r^d$ is the nominal roll angle that is usually set to 0, k_r is the feedback gain for the roll RWA momentum, ρ_r is the roll RWA momentum, and θ_r^d is the final desired roll angle for the robot. We found that this feedback term is essential to avoid RWA rotor speed saturation during the hardware experiments. There are a number of factors that contribute to RWA saturation during the hardware experiments, including center-of-mass mismatch between simulated and physical model, and biases in the state estimation pipeline. This feedback term, along with the inequality constraints in (22c), keeps the RWA from saturating during the balance-beam walking experiment described in Section VI.

VI. Experiments and Results

We now present the simulation and hardware experiment results for the gyrostat MPC on a Unitree A1 robot equipped with our RWA module. In simulation, we tested the system's disturbance rejection and aerial reorientation abilities. On hardware, we tested the system's balancing capability through a beam walking experiment. To the best of our knowledge, the experiment presented in this work is the first successful hardware demonstration of a beam walk done by a quadruped robot, where the robot has to continuously balance itself with a near-empty support polygon.

A. Hardware/Simulation Setup

We built the gyrostat MPC controller on top of an open-source convex MPC implementation for the Unitree A1 robot[2]. The MPC problem from (22) is solved using OSQP [24] online at 1000Hz on a computer equped with an AMD Threadripper 3 CPU. The MPC employs a 20-step horizon and a 0.05 s time step, and we design a controller stack that uses the same MPC code to control either a hardware robot or a simulated robot. On hardware, the control torque command is sent to the robot joint motors and the RWA drivers via Ethernet. In the simulation, the controller interfaces with the Gazebo simulator [14]. To reduce the sim-to-real gap, the Gazebo simulation uses the precise model mass and inertial parameters from the robot manufacturer. It also uses a motor dynamics simulation, a fine-tuned soft foot contact model, and accurate sensor noise injections according to sensor data sheets.

[2]https://github.com/ShuoYangRobotics/A1-QP-MPC-Controller

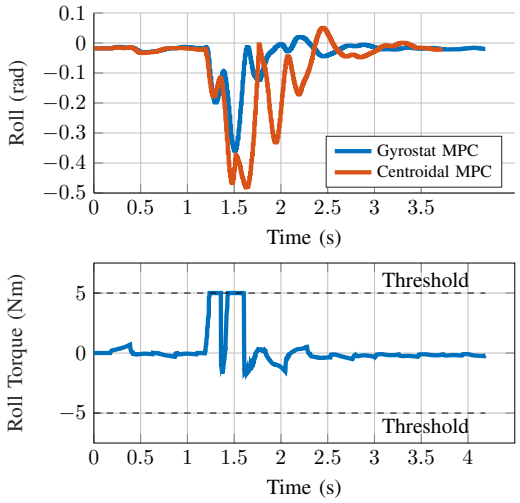
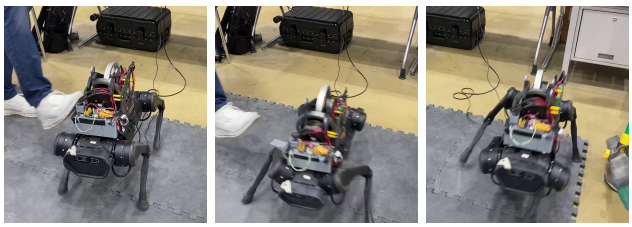


Fig. 3: Roll error (top) and RWA torque responses (bottom) to a 650 N impulse applied to the robot's body y-axis at $t = 1.2$ seconds.

B. Locomotion Disturbance Rejection

In the simulated disturbance rejection tests, we applied a 600 N, 650 N, and 700 N force to the y-axis of the robot body while trotting. The impulse used in our experiment is defined as a continuous force disturbance that lasts 0.05 s, and the disturbance is applied at precisely the same gait phase for each test. The same impulse is applied to a robot equipped with the RWA module but running the base centroidal MPC controller [7] and the same robot running the gyrostat MPC controller. Figure 3 shows the roll error response of both robots during the 650 N impulse test in Gazebo. The



(a) (b) (c)

Fig. 4: Hardware impulse test where we provide an impulse force on the robot during locomotion with a kick. 4a shows the robot performing stable trotting when the impulse is applied. 4b shows the robot losing balance on the footholds while maintaining a stable attitude as it eventually recovers from the impulse in 4c.

maximum roll error is reduced by 26% with faster steady-state convergence. The experiments demonstrate an enhanced ability to recover from sudden impacts due to better inertial stabilization when the robot is airborne. During the 700 N impulse experiments, the gyrostat MPC controller consistently recovers from the impact while the base centroidal MPC controller fails. Figure 4 demonstrates this disturbance rejection behavior on hardware, though without precisely quantified impulses.

C. Aerial Re-orientation

We also tested the aerial reorientation capability of the gyrostat MPC controller. By locking the joints of the robot and solely relying on the torques from the RWAs, we dropped the robot from a height of 0.5 m with a 0.6 radian initial roll error, as shown in Figure 5. The RWAs were able to correct the robot's orientation in before touchdown. This experiment verifies that the RWAs are able to quickly correct large orientation errors in mid-air.

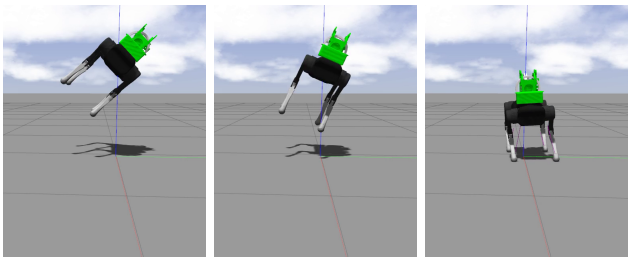


Fig. 5: A drop test sequence where the robot reorients itself with the torques from the reaction wheels. The robot is initially positioned 50 cm off the ground with a 0.6 radian roll error. The RWA controller is set to turn on immediately after release to correct the attitude error.

D. Balance-Beam Walking

To demonstrate the full capability of the gyrostat MPC controller on hardware, we perform a traversal of a narrow wooden beam, as shown in Figure 1 and the supplementary video. Since the support polygon is always close to a line, it is almost impossible for a standard quadruped to perform balancing and locomotion simultaneously on the beam. In contrast, our robot with the RWA module and the gyrostat MPC controller is able to maintain a stable roll angle to keep itself from falling.

Hardware demonstration with the Unitree A1 poses a major challenge — the robot's deformable rubber feet cause unwanted vibration at higher walking frequency. To achieve stable beam walking, we lengthen the gait cycle from 1s to 3s. This significantly reduced the robot's walking speed and largely eliminated the vibration. We also fuse the robot's proprioceptive sensor data with an external motion-capture system in a Kalman Filter to achieve sub-centimeter position estimation so the robot can place foothold locations with sufficient accuracy to stay on the beam, which is less than 6 cm wide. Figure 6 shows the roll angle, roll RWA torque, and roll RWA velocity collected during the beam walking experiment. The RWA speed and torque are effectively kept under the threshold limits (200 rad/s and 5 Nm respectively) by the MPC constraints.

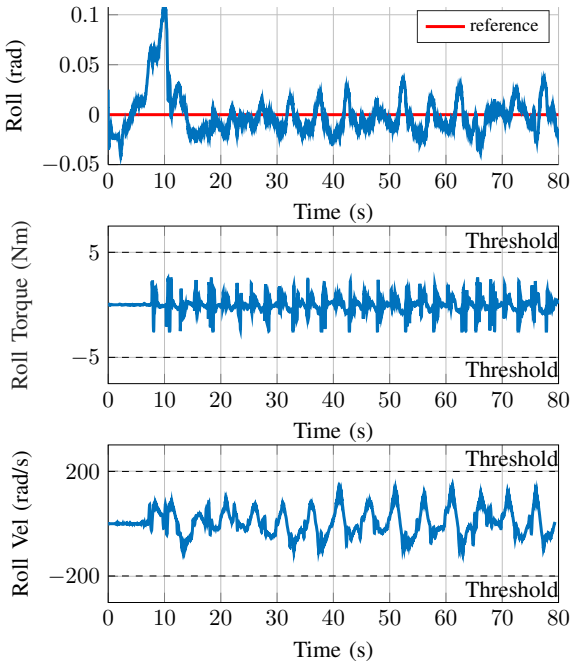


Fig. 6: Roll angle, roll RWA torque, and roll RWA velocity during the hardware beam walking experiment.

VII. Conclusions and Future Work

We have presented a reaction-wheel actuator system that enhances quadrupedal robots' balancing and stabilization abilities during challenging locomotion tasks. We have shown that RWAs can be integrated into a state-of-the-art MPC framework with a few relatively minor modifications, and we have demonstrated our proposed gyrostat MPC controller in a series of hardware and simulation experiments. Our simulated experiments demonstrate that the RWA module helps the quadruped handle larger disturbances and gives it self-righting capabilities in mid-air. On hardware, we have successfully demonstrated the first narrow-beam walking performed by a quadruped. In the future, we believe that RWA modules like ours can be better optimized for lower power consumption and weight, and be integrated into many legged robot designs for improved robustness.

References

[1] Gerardo Bledt and Sangbae Kim. "Implementing Regularized Predictive Control for Simultaneous Real-Time Footstep and Ground Reaction Force Optimization". In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019, pp. 6316–6323. DOI: 10.1109/IROS40897.2019.8968031.

[2] Gerardo Bledt et al. "MIT Cheetah 3: Design and control of a robust, dynamic quadruped robot". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 2245–2252.

[3] Travis Brown and James Schmiedeler. "Energetic effects of reaction wheel actuation on underactuated biped robot walking". In: May 2014, pp. 2576–2581. DOI: 10.1109/ICRA.2014.6907228.

[4] Travis L. Brown and James P. Schmiedeler. "Reaction Wheel Actuation for Improving Planar Biped Walking Efficiency". In: *IEEE Transactions on Robotics* 32.5 (2016), pp. 1290–1297. DOI: 10.1109/TRO.2016.2593484.

[5] Matthew Chignoli and Patrick M. Wensing. "Variational-Based Optimal Control of Underactuated Balancing for Dynamic Quadrupeds". In: *IEEE Access* 8 (2020), pp. 49785–49797. DOI: 10.1109/ACCESS.2020.2980446.

[6] Simon Le Cleac'h et al. *Fast Contact-Implicit Model-Predictive Control*. 2021. DOI: 10.48550/ARXIV.2107.05616. URL: https://arxiv.org/abs/2107.05616.

[7] Jared Di Carlo et al. "Dynamic Locomotion in the MIT Cheetah 3 Through Convex Model-Predictive Control". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid: IEEE, Oct. 2018, pp. 1–9. ISBN: 978-1-5386-8094-0. DOI: 10.1109/IROS.2018.8594448. URL: https://ieeexplore.ieee.org/document/8594448/.

[8] Mason A. Peck Frederick A. Leve Brian J. Hamilton. *Spacecraft Momentum Control Systems*. Springer, 2015.

[9] Mohanarajah Gajamohan et al. "The Cubli: A cube that can jump up and balance". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012). Vilamoura-Algarve, Portugal: IEEE, Oct. 2012, pp. 3722–3727. ISBN: 978-1-4673-1736-8 978-1-4673-1737-5 978-1-4673-1735-1. DOI: 10.1109/IROS.2012.6385896. URL: http://ieeexplore.ieee.org/document/6385896/.

[10] Carlos Gonzalez et al. "Line Walking and Balancing for Legged Robots with Point Feet". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 3649–3656. DOI: 10.1109/IROS45743.2020.9341743.

[11] Marco Hutter et al. "Quadrupedal locomotion using hierarchical operational space control". In: *The International Journal of Robotics Research* 33.8 (2014), pp. 1047–1062.

[12] Brian E. Jackson, Kevin Tracy, and Zachary Manchester. "Planning With Attitude". In: *IEEE Robotics and Automation Letters* 6.3 (2021), pp. 5658–5664. DOI: 10.1109/LRA.2021.3052431.

[13] Aaron M. Johnson et al. "TAIL ASSISTED DYNAMIC SELF RIGHTING". In: A K M Azad et al. *Adaptive Mobile Robotics*. WORLD SCIENTIFIC, Sept. 2012, pp. 611–620. ISBN: 978-981-4415-94-1 978-981-4415-95-8. DOI: 10.1142/9789814415958_0079. URL: http://www.worldscientific.com/doi/abs/10.1142/9789814415958_0079.

[14] N. Koenig and A. Howard. "Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator". In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*. Vol. 3. Sendai, Japan: IEEE, 2004, pp. 2149–2154. ISBN: 978-0-7803-8463-7. DOI: 10.1109/IROS.2004.1389727. URL: http://ieeexplore.ieee.org/document/1389727/ (visited on 07/25/2022).

[15] Hendrik Kolvenbach et al. "Towards Jumping Locomotion for Quadruped Robots on the Moon". In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019, pp. 5459–5466. DOI: 10.1109/IROS40897.2019.8967552.

[16] Thomas Libby et al. "Comparative Design, Scaling, and Control of Appendages for Inertial Reorientation". In: *IEEE Transactions on Robotics* 32.6 (Dec. 2016), pp. 1380–1398. ISSN: 1552-3098, 1941-0468. DOI: 10.1109/TRO.2016.2597316. URL: http://ieeexplore.ieee.org/document/7562541/.

[17] Richard Montgomery. "Gauge theory of the falling cat". In: *Fields Inst. Commun.* 1 (July 1993). DOI: 10.1090/fic/001/09.

[18] Michael Neunert et al. "Whole-Body Nonlinear Model Predictive Control Through Contacts for Quadrupeds". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 1458–1465. DOI: 10.1109/LRA.2018.2800124.

[19] Joseph Norby et al. "Enabling Dynamic Behaviors With Aerodynamic Drag in Lightweight Tails". In: *IEEE Transactions on Robotics* 37.4 (Aug. 2021). Conference Name: IEEE Transactions on Robotics, pp. 1144–1153. ISSN: 1941-0468. DOI: 10.1109/TRO.2020.3045644.

[20] Jongwon Park et al. "Raptor: Fast bipedal running and active tail stabilization". In: *2014 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. 2014, pp. 215–215. DOI: 10.1109/URAI.2014.7057424.

[21] Amir Patel and M. Braae. "Rapid turning at high-speed: Inspirations from the cheetah's tail". In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2013 IEEE/RSJ International

Conference on Intelligent Robots and Systems. ISSN: 2153-0866. Nov. 2013, pp. 5506–5511. DOI: 10.1109/IROS.2013.6697154.

[22] Marko Popovic, Andreas Hofmann, and Hugh Herr. "Angular Momentum Regulation during Human Walking: Biomechanics and Control." In: vol. 3. Jan. 2004, pp. 2405–2411. DOI: 10.1109/ROBOT.2004.1307421.

[23] Marc H. Raibert and Ernest R. Tello. "Legged Robots That Balance". In: *IEEE Expert* 1.4 (1986), pp. 89–89. DOI: 10.1109/MEX.1986.4307016.

[24] B. Stellato et al. "OSQP: an operator splitting solver for quadratic programs". In: *Mathematical Programming Computation* 12.4 (2020), pp. 637–672. DOI: 10.1007/s12532-020-00179-2. URL: https://doi.org/10.1007/s12532-020-00179-2.

[25] Shuo Yang, Howie Choset, and Zachary Manchester. "Online Kinematic Calibration for Legged Robots". In: *IEEE Robotics and Automation Letters* 7.3 (2022), pp. 8178–8185. DOI: 10.1109/LRA.2022.3186501.

[26] Yanhao Yang et al. "Improving Tail Compatibility Through Sequential Distributed Model Predictive Control". In: *RSS Workshop on Software Tools for Real-Time Optimal Control*. July 2021.