

Terrestrial Locomotion of PogoX: From Hardware Design to Energy Shaping and Step-to-step Dynamics Based Control

Yi Wang*, Jiarong Kang*, Zhiheng Chen*, and Xiaobin Xiong

Abstract—We present a novel controller design on a robotic locomotor that combines an aerial vehicle with a spring-loaded leg. The main motivation is to enable the terrestrial locomotion capability on aerial vehicles so that they are carrying heavy loads: heavy enough that flying is no longer possible, e.g., when the thrust-to-weight ratio (TWR) is small. The robot is designed with a pogo-stick leg and a quadrotor, and thus it is named as PogoX. We show that with a simple and lightweight spring-loaded leg, the robot is capable of hopping with $TWR < 1$. The control of hopping is realized via two components: a *vertical height control* via control Lyapunov function-based energy shaping, and a step-to-step (S2S) dynamics based *horizontal velocity control* that is inspired by the hopping of the Spring-Loaded Inverted Pendulum (SLIP). The controller is successfully realized on the physical robot, showing dynamic terrestrial locomotion of PogoX which can hop at variable heights and different horizontal velocities with robustness to ground height variations and external pushes.

I. INTRODUCTION

Flying robots, such as unmanned aerial vehicles (UAVs) with quadrotors as an iconic example, are increasingly popular in modern society [1]. The prevalence of UAVs can be attributed largely to the fact that they can be agile and autonomous. Such favorable characteristics have rendered UAVs with a wide range of applications, including but not limited to search and rescue [2], package delivery [3], agricultural applications [4], and exoplanet exploration [5].

Despite the successes and great potential in these fields, current quadrotors have their own limitations in terms of energy consumption, short operation time, human-robot-interaction, and most importantly payload weight. Under heavy load when the thrust-to-weight ratio (TWR) is less than 1, flying is no longer feasible. Moreover, flying robots have difficulties at near ground operation and are not able to interact with the ground as they can only be stabilized in mid-air. In contrast, legged robots naturally can carry heavy loads and perform terrestrial locomotion by interacting with the ground with their feet [6], [7], [8]. This motivates us to append legs to UAVs to increase their payload capacity by performing terrestrial locomotion. Thus, in this paper, we present an effective robot design by adding a light weighted spring-leg appendage to a custom designed FPV type quadrotor, namely PogoX, and, most importantly, a controller design with $TWR < 1$ to realize dynamic hopping on this robot with different horizontal velocities and hopping height.

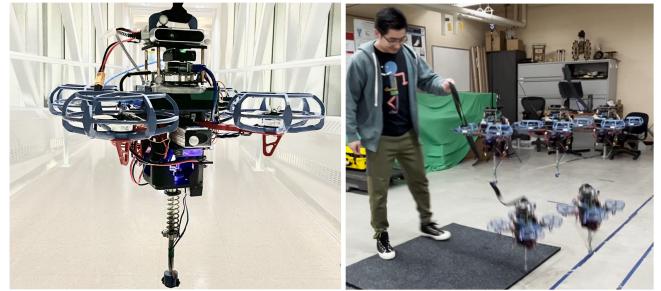


Fig. 1. PogoX (left) and its hopping behavior (right).

Robotic systems that combine legs or wheels [9], [10], [11], [12], [13] with propellers have existed in the literature. Most work mainly focused on creating multi-modal locomotion capability rather than dynamic hopping with heavy loads, which is our focus. For instance, [12] appended wheels on drones that combine flying and wheeled locomotion. [11] combined a pair of bipedal legs with propellers on its torso, and [9] appended a bipedal mechanism under a quadrotor; both produced flying and walking on their robots. [10] appended a spring under a miniature quadrotor that produced hopping; however the TWR of their system appears to be much bigger than one, and versatile and robust repetitive hopping behaviors have not been shown.

With $TWR < 1$, the robot becomes highly *underactuated* as directly controlling the Cartesian coordinates in the continuous phases is not possible, which prevents the direct applications of existing control frameworks [14], [10] on quadrotors. To address this challenge of underactuation, we draw inspiration from the legged robotic community to utilize the Spring-loaded Inverted Pendulum (SLIP) [15], [16], [17], [18], [19], [20], [21] to control balancing in the horizontal directions, along with which is a Lyapunov based controller for energy stabilization in the vertical direction.

The energy controller is driven by a control Lyapunov function-based quadratic program (CLF-QP) [22] to stabilize the vertical energy to a desired level, which enables periodic hybrid dynamic behaviors of hopping. Then, under this energy controller, a SLIP model is used off-line to numerically identify periodic orbits with its step-to-step (S2S) dynamics [23], [24], [25] for feedback stabilization. The combination of the two controllers can thus realize periodic hopping on the robot with onboard computation. We realize this controller design both in simulation and hardware, showing dynamic, versatile, and robust hopping behaviors on the custom-designed robotic system of PogoX with $TWR < 1$.

* The authors contribute equally to this work.

¹The authors are with the Wisconsin Expedited Legged Locomotion (WELL-Lab) at the University of Wisconsin-Madison. Corresponding to xiaobin.xiong@wisc.edu

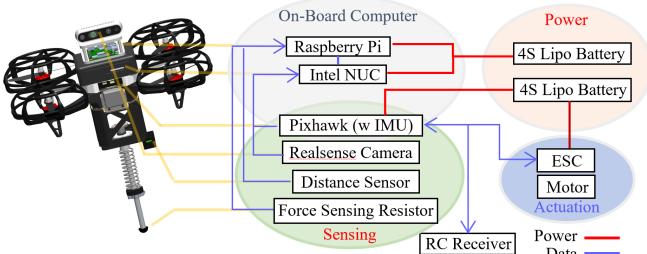


Fig. 2. Illustration of the hardware system of PogoX.

II. HARDWARE SYSTEM DESIGN

In this section, we briefly describe the hardware design of the robot, PogoX. The design architecture aims towards fully autonomous operation of this robot in the field. Besides the goal of terrestrial locomotion with $TWR < 1$, this design integration also can expand the operational envelope of conventional flying robots, leveraging the advantages of both flight and hopping to navigate in complex environments.

A. Mechanical Design

On the mechanical design front, we select an off-the-shelf quadrotor frame that has a diagonal wheelbase at 330mm and tri-blade polycarbonate propellers that are 5.1-inch in diameter; the sizes are chosen so that various sensors, batteries, and computational resources can be mounted on the center of the body, and also that the propellers can provide sufficient thrusts. The propellers are covered by 3D printed guards to provide safety to operations with human in close proximity. The terrestrial mobility is realized via a pogo-stick leg, which can store potential energy for subsequent highly energetic maneuvers. The leg is connected through a custom-designed hollow support structure shown in Fig. 2, which is made from 3D-printed PLA-CF materials. The hollow structure provides a maximum 10cm displacement for the spring. The pogo stick itself consists of an aluminum tube and a helical spring with a natural length of 8cm and stiffness at 4848.5N/m. Linear bearings and clamping hubs are installed in place to guide the linear motion of the leg along the body. All the components are either off-the-shelf or 3D printed. The leg appendage is of low cost and lightweight, making it straightforward to be deployed on regular quadrotors.

B. Electronics and Control Systems

The electrical system of the robot is designed with the goal that the robot can perform autonomous hopping and hybrid locomotion using all onboard sensors, computational resources, and power. For the motors, we select T-Motor Velox (V2 2550KV), which has a maximum power output of 545W with a weight of 34.3g, engineered to provide the force needed for complex locomotion. A Pixhawk 6C Mini that runs the open-source PX4 software is used for motor control. To detect foot-ground contact, a FlexiForce A101 sensor is integrated with a custom-designed foot that is then mounted at the end of the leg. A Garmin LIDAR-Lite

v4 LED Distance Measurement Sensor with an accuracy of $\pm 1\text{cm}$ is installed on the body to detect the ground height and measure the spring displacement. We also use an Intel RealSense Depth Camera D435if to enhance perception, which in combination with an internal IMU inside the Pixhawk 6C Mini is used to provide visual-inertia odometry (VIO) for state estimation. To manage these multiple sensory inputs, a Raspberry Pi 4b is used to communicate these sensors and send the data through an Ethernet cable to an Intel NUC 13 Pro onboard computer, which performs the state estimation and optimization-based control realization. Two 4S-100C Lipo batteries with 3300mAh capacity, each weighing 12.52 ounces, are used to provide power to all the sensors, motors, and computers. With all the components installed, the robot weighs 2.5kg with a height of 0.45m.

III. HYBRID DYNAMICS OF HOPPING

The hopping of PogoX is modeled as a dynamic system of two rigid bodies connected by a prismatic joint. The first rigid body is the floating base, consisting of the drone body, and the second rigid body is the lower part of the leg. The spring force acts directly on the prismatic joint. In total, the system has 7 degrees of freedom (DoFs), with 6 DoFs from the floating base, and another DoF from the prismatic joint. Thus, the configuration of the robot is $\mathbf{q} \in SE(3) \times \mathbb{R}$.

The dynamics of periodic hopping is composed of two phases: an aerial phase and a stance phase. The transition from the aerial phase to the stance phase is a discrete impact map, whereas the transition from the stance phase to the aerial phase is smooth. The hybrid dynamics structure is shown in Fig. 3.

We assume the hopping behaviors of the robot are controlled at relatively low speeds ($\leq 5\text{m/s}$), thus aerodynamic drag forces are neglected. Additionally, with the presence of the leg, the propellers on the body are relatively far away from the ground, and consequently the ground effects of the propellers are considered negligible.

A. Aerial Phase Dynamics

In the aerial phase, PogoX is actuated via its propellers. The dynamics are derived from Euler-Lagrangian Equations:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\dot{\mathbf{q}}, \mathbf{q}) = \mathbf{B}\mathbf{F}_s + \mathbf{J}^T(\mathbf{q})\mathbf{F}, \quad (1)$$

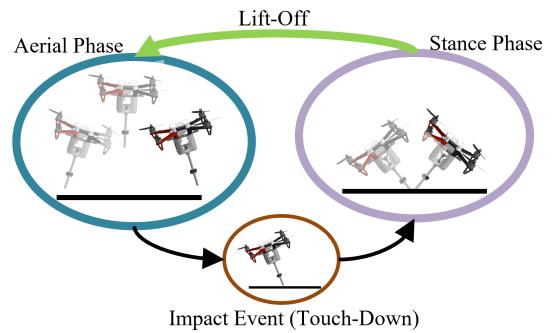


Fig. 3. Illustration of the hybrid dynamics model of PogoX.

where \mathbf{M} is the mass matrix, \mathbf{H} is the Coriolis, centrifugal and gravitational forces, \mathbf{B} is a constant mapping matrix, F_s is the spring force, \mathbf{F} represents the thrusts and moments from the propellers, and \mathbf{J} is their corresponding Jacobians.

B. Stance Phase Dynamics

In the stance phase, we assume the contact between the foot and the ground is non-slipping, which introduces holonomic constraints to the dynamics. The constraints are:

$$\mathbf{J}_f \dot{\mathbf{q}} = 0, \quad \mathbf{J}_f \ddot{\mathbf{q}} + \dot{\mathbf{J}}_f \dot{\mathbf{q}} = 0 \quad (2)$$

where \mathbf{J}_f is the Jacobian of the foot position. Combining (1) and (2) yields the differential equations that govern the stance phase dynamics:

$$\begin{bmatrix} \mathbf{M} & -\mathbf{J}_f^T \\ \mathbf{J}_f & \mathbf{0}_{3 \times 3} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{F}_{GRF} \end{bmatrix} = \begin{bmatrix} \mathbf{Bu} + \mathbf{J}^T \mathbf{F} - \mathbf{H}(\dot{\mathbf{q}}, \mathbf{q}) \\ -\dot{\mathbf{J}}_f \dot{\mathbf{q}} \end{bmatrix} \quad (3)$$

where \mathbf{F}_{GRF} is the ground reaction force on the foot.

C. Impact Mapping at Touch-Down

We assume the impact between the foot and the ground at touch-down is purely plastic [26]. The post-impact state satisfies the holonomic constraints, and the impulse of the impact creates a change of momentum.

$$\begin{bmatrix} \mathbf{M} & -\mathbf{J}_f^T \\ \mathbf{J}_f & \mathbf{0}_{3 \times 3} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}^+ \\ \mathbf{F}_{imp} \end{bmatrix} = \begin{bmatrix} \mathbf{M}\dot{\mathbf{q}}^- \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (4)$$

where \mathbf{F}_{imp} is the impulse force from the ground, and $+$ and $-$ represent the states at *post-* and *pre-*impact, respectively.

D. Control Inputs for Hopping

With the derivations of the dynamics in the individual domains and their transitions, we have a hybrid controlled system with the control inputs being \mathbf{F} , i.e., the thrusts and moments of the propellers. Based on the propeller models in [27], we assume linear relationships between the motor torque and ω^2 , and ω^2 and propeller thrust force, where ω is the rotational speed of the propeller. Since the electric motors are controlled by ESC with sensorless Field Oriented Control (FOC), the realizable ω is in a certain range $[\omega_{min}, \omega_{max}]$. As a result, the propeller moment is bounded, i.e., $\tau \in [\tau_{min}, \tau_{max}]$ and so is the thrust force.

With an eye towards the control of hopping, we configure the “flight controller” of the hardware to its “off-board” model, in which we can command desired target total thrust forces F_t with desired roll, pitch, and yaw angles $[r, p, y]$; the “flight controller” then try to realize these commands via PID controllers, the associated feedback gains of which are custom tuned. With the orientation being controlled by the flight controller, we then only need to utilize the four inputs $[F_t, r, p, y]$ to realize hopping. In order to identify the input bound of F_t , a linear program was formulated to solve for the minimal thrust required by the “flight controller”:

$$F_t^{min} = \min_{\tau \in \mathbb{R}^4} \sum k_t \tau_i$$

$$\text{s.t. } \mathbf{A}_m \boldsymbol{\tau} = \mathbf{M}_b$$

$$\tau_{max} \geq \tau_i \geq \tau_{min}, \forall \tau_i \in \boldsymbol{\tau}$$

where $\boldsymbol{\tau}$ represents the torques generated by all propellers, and k_t is the constant ratio from the propeller moment to thrust force. $\mathbf{M}_b \in \mathbb{R}^3$ is the moment required by the flight controller represented in the body frame to realize orientation control. $\mathbf{A}_m \in \mathbb{R}^{3 \times 4}$ stands for the constant linear relationship between $\boldsymbol{\tau}$ and \mathbf{M}_b . The maximum total thrust force can be identified similarly with setting an additional constraint that $F_t^{max} < rmg$, where m is the total robot mass, and $r < 1$ represents the thrust-to-weight ratio (TWR).

IV. CONTROL SYNTHESIS

In this section, we present the controller design for realizing hopping on this hybrid system. As the orientation of the robot can be controlled directly, we mainly focus on the control of the center of mass (COM) through the hybrid behavior of hopping. The springy leg provides a desired passive dynamics of hopping. The propellers of the quadrotor can be used to inject or dissipate the energy of the whole system during hopping. Additionally, through orientation control using the propellers, the leg angles can be modulated to change the locomotion behavior.

The control of hopping is decoupled into a vertical energy shaping part for realizing a desired apex height, and a horizontal stabilization part for achieving a target forward velocity. In the later subsections, for the vertical control, we formulate a quadratic programs based controller to compensate for vertical energy loss due to impact and spring damping; then we describe a step-to-step (S2S) dynamics [28], [21], [29] based horizontal velocity controller, which is inspired by Spring loaded Inverted Pendulum (SLIP) [30], [24] to achieve robust stepping.

A. Energy Shaping for Vertical Control

Rather than directly controlling the vertical height, we consider driving the vertical total energy to a desired value as it allows the natural flow between potential and kinetic energy during hopping. Additionally, as the ground phase is very short, we only apply the control in the aerial phase.

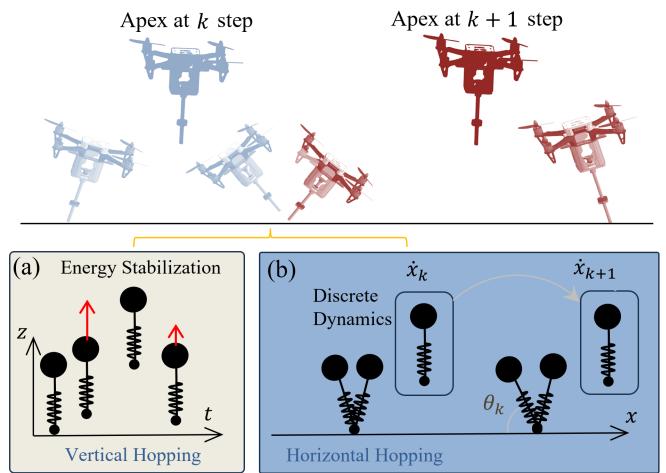


Fig. 4. Illustration of the synthesis of decoupled controllers: (a) vertical energy stabilization and (b) horizontal balancing via SLIP model.

Thus, the output of the control system is defined as:

$$\eta = E - E_d, \quad (5)$$

where E_d is the desired constant energy level corresponding to the desired apex position. E is the vertical total energy, which is the sum of the vertical kinetic energy and the potential energy coming from the gravitational force and spring force. The spring potential energy is neglected in the aerial phase. Thus $E = \frac{1}{2}m\dot{z}^2 + mgz$, where m is the total mass of the system, z is the vertical position,

Differentiating the output yields the output dynamics:

$$\dot{\eta} = \dot{E} = f_\eta + g_\eta F_t \quad (6)$$

where F_t is the magnitude of the total thrust force of the quadrotor. $f_\eta = 0$ in this case. A feedback linearizing controller can be designed for output stabilization: $F_t = \frac{1}{g_\eta} K_p \eta$, where $K_p < 0$ is the feedback gain. This yields a stable closed-loop dynamics:

$$\dot{\eta} = g_\eta F_t = K_p \eta := A_{cl} \eta. \quad (7)$$

A Lyapunov function can thus be synthesized based on the closed-loop output dynamics: $V_\eta = P\eta^2$, with $P > 0$ satisfies the Lyapunov equation $2A_{cl}P = -Q$, where $Q > 0$ is a user-defined variable [31], [32]. Note that \dot{V} is affine w.r.t. the input F_t : $\dot{V}(\eta, F_t) = 2P\eta\dot{\eta} = 2P\eta g_\eta F_t$. Then V is thus a control Lyapunov function (CLF), which motivates the use of the CLF inequality condition on the input F_t to stabilize the output η . The system will be exponentially stable when enforcing:

$$\dot{V}(\eta, F_t) \leq -\gamma V(\eta, F_t), \quad (8)$$

with $\gamma > 0$. This inequality is affine w.r.t. the input:

$$A_{CLF}F_t \leq b_{CLF}, \quad (9)$$

where $A_{CLF} = 2P\eta g_\eta$, and $b_{CLF} = -\gamma P\eta^2$. Thus a **control Lyapunov function based quadratic program (CLF-QP)** can be formulated for optimizing F_t , subject to (9) and input bounds:

$$(F_t, \delta) = \underset{(F_t, \delta) \in \mathbb{R}^2}{\operatorname{argmin}} pF_t^2 + \delta^2 \quad (10)$$

$$\text{s.t. } A_{CLF}F_t \leq b_{CLF} + \delta$$

$$F_t^{\min} \leq F_t \leq F_t^{\max}$$

where δ is the relaxation term to ensure feasibility, and p is a tuning coefficient on the cost.

Remark 1: In practice, F_t is constantly bounded below by F_t^{\min} to avoid reversing motor rotation and maintain system operation. It is equivalent to adding a constant anti-gravitational force. Thus, we redefine the vertical energy:

$$E(t) = mge_z + \frac{m\dot{z}^2}{2}, \quad (11)$$

using equivalent gravity constant $g_e = g - \frac{F_t^{\min}}{m}$. The resulting control F_t will have approximately equivalent effect to the case when the minimal bound equals 0.

Remark 2: The inequality CLF condition in (8) enforces a convergence rate at no less than γ . Thus, the CLF-QP

formulation can yield excessively large convergence rate that can lead to problematic aggressive behaviors in real systems. Inspired by task-space QP based controllers [33], [23], [34], [35], we encourage a constant convergence rate of the Lyapunov function by changing the inequality in (8) to an equality with a relaxation. The final QP becomes:

$$(F_t, \delta) = \underset{(F_t, \delta) \in \mathbb{R}^2}{\operatorname{argmin}} p(F_t - F_t^{pre})^2 + \delta^2 \quad (12)$$

$$\text{s.t. } A_{CLF}F_t = b_{CLF} + \delta$$

$$F_t^{\min} \leq F_t \leq F_t^{\max}$$

Minimizing the relaxation term is equivalent to minimizing the $|\dot{V} - \dot{V}^d|^2$ with the desired $\dot{V}^d = -\gamma V(\eta, F_t)$, which then is the cost in standard task-space control based QP formulations [21]. Additionally, instead of penalizing the magnitude of the input F_t , we try to reduce the instantaneous changes of it by adding $(F_t - F_t^{pre})^2$ in the cost, where F^{pre} is the applied force from the previous control sample.

B. S2S Dynamics based Horizontal Stabilization

Now we present the horizontal state control based on a energy controlled SLIP model. The goal is to control the apex velocity in the horizontal plane. The control task is first solved in the planar case, and then applied respectively in its sagittal and lateral planes to stabilize the system in 3D.

SLIP Dynamics: We consider to use the Spring-loaded Inverted Pendulum (SLIP) model to approximate the COM dynamics of the robot, since the spring dynamics contributes heavily to the hopping behavior. Damping is added to the spring, the coefficient of which comes from the spring on the robot. Moreover, the vertical energy controller is applied on the SLIP so that the vertical energy conserves. Its dynamics can be compactly represented by $m\ddot{\mathbf{p}} = \mathbf{F}_t + \mathbf{F}_s + mg$, where \mathbf{p} represent mass position. \mathbf{F}_t and \mathbf{F}_s are the thrust and spring forces, respectively, and they are in the same direction. $|\mathbf{F}_s| = k_s s + d_s \dot{s}$ in the ground phase with k_s , d_s , and s being the stiffness, damping, and deformation of the spring, respectively, and $\mathbf{F}_s = 0$ with the robot is in the air. The magnitude of \mathbf{F}_t is calculated from the vertical controller and its direction aligns with the leg angle. The leg is assumed massless, and thus the leg angle θ can be directly assigned in the aerial phase. For the canonical SLIP model [15], θ can be set to the touchdown angle in the the aerial phase. As for our SLIP model, swing leg trajectory is part of the vertical energy-shaping controller, so a Bézier curve is used to construct a smooth swing leg trajectory $\theta(t)$ from the angle at lift-off to the desired touch down angle.

Periodic Orbit and Return Map: To realize periodic hopping on the robot, we first identify periodic orbits of the SLIP. Given a desired vertical height and a target horizontal velocity, a periodic orbit with a desired touch down angle θ^{td} can be numerically identified with a given Bézier curve coefficients and the gains in the energy controller. Then we can numerically derive its return map.

Note that for the canonical SLIP with running behavior, the dimension of the state space of the Poincaré map is reduced from four to two: the CoM height and CoM forward

velocity [24], [36]. With the aforementioned vertical energy controller, our SLIP maintains a constant apex height. As a result, the Poincaré map of our SLIP has only one state variable: the CoM forward velocity \dot{x} . Taking the touch down angle as the input, the Poincaré map is:

$$\dot{x}_{k+1} = \mathcal{P}(\dot{x}_k, u_k), \quad (13)$$

where \dot{x}_k, \dot{x}_{k+1} are the horizontal velocities at the apex of consecutive steps, and $u_k = \theta_k^{td}$ denotes the control input.

S2S Dynamics for Stabilization: The return map is nonlinear in nature. To utilize it for control, we derive its first-order numerical approximation:

$$\dot{x}_{k+1} - \dot{x}^* = A(\dot{x}_k - \dot{x}^*) + B(u_k - u^*) + \delta, \quad (14)$$

where $A = \frac{\partial \mathcal{P}}{\partial \dot{x}}$, $B = \frac{\partial \mathcal{P}}{\partial u}$ are the Jacobian of Poincaré map evaluated at (\dot{x}^*, u^*) , and δ is the linear approximation error here, which will also incorporate the modeling from the SLIP to the robot. We denote (14) as the Step-to-step (S2S) dynamics [23]. The control object of driving \dot{x}_{k+1} to \dot{x}^* can be achieved by selecting:

$$u_k = u^* + K(\dot{x}_k - \dot{x}^*), \quad (15)$$

which yields the closed-loop S2S error dynamics [23]:

$$e_{k+1} = (A + BK)e_k + \delta \quad (16)$$

with $e_k = \dot{x}_k - \dot{x}^*$. K is chosen based on $(A + BK) = 0$ so that the error is stabilized in one step. In practice with δ being small and bounded, the errors converge to an error invariant set \mathcal{E} [23], yielding bounded velocity tracking error.

V. RESULTS

Now we present the results of our controller implementation in both simulation and experiment. We also utilize our simulation to verify our design choices and compare energy consumption between flying and hopping for robots with different TWRs. Then, we focus on realizing different hopping behaviors on the custom-designed hardware.

A. Simulation

The simulation is realized using MATLAB based on the hybrid dynamical model in Section III. The dynamics are integrated via ODE45 with event triggering for domain transitions. The physical parameters of the SLIP, i.e., the mass and the stiffness, are selected to be similar to those of the robot. The periodic orbits and their S2S dynamics of the SLIP model are offline identified as a *Priori* based on the selected hopping height and forward velocity. The leg angle controller in (15) is realized in the aerial phase to decide the target touch-down angle, which is mapped to the desired roll and pitch angles of the body. The QP-based controller in (12) is realized using qpOASES [37] at 200Hz to calculate the desired thrust force F_t . Finally, the desired control inputs of hopping $[F_t, r, p, y]$ are realized by PID controllers by sending desired rotational velocities to the propellers.

Periodic Hopping: We show the realized periodic hopping with a desired velocity of 0.5m/s and a desired COM apex height of 0.5m. Both the horizontal velocity and the

vertical energy are stabilized as expected. We also verify that both parameters of the desired hopping behavior can be changed as we search for a different periodic orbit of the SLIP model and their associated S2S dynamics. Additionally, the implemented controller is shown to have robustness to external push disturbances. Fig. 5 shows the trajectories of a simulated periodic hopping of the robot, and that under an external push force at 10N for 0.1s.

Design Analysis: In order to obtain the design principle of system parameters (spring stiffness and maximum thrusts) with a given robot weight and target jumping height, we utilize the SLIP model to analyze the vertical hopping of PogoX. To simplify the dynamics, we assume it only hops in the vertical direction and the control is bang-bang (maximum thrust in the ascending phase and minimum thrust in the descending phase); then the dynamics become linear in all phases. Closed-form solutions are thus solved to obtain the relations between the jumping height, maximum thrust, and spring stiffness. Fig. 6 (a,b) shows the selection of the spring stiffness can affect the maximum thrust needed to realize a certain height for different weights, and the relation between apex height v.s. TWR.

Cost of Transport: We compare the cost of transport (COT) of the same robot under different payloads between different locomotion modes: periodic hopping v.s. flying.

The COT is defined as: $\frac{1}{Lmg} \sum_{i=1}^4 \int_{t_0}^{T_f} |F_i| dt$, where L is the total horizontal distance traveled by the robot for a time period T_f . The energy consumption is calculated based on the integration of the applied thrust forces. Without losing generality, we control the robot to realize a 1m/s horizontal velocity in both flying and hopping; for hopping, the height is chosen to be 1m, and the minimum thrust force is set to 0 for simplification. For a fixed robot weight, we gradually decrease the maximum motor thrust forces, which equivalently decreases the TWR. Fig. 6 (c) shows the comparison. For flying, the COT is a finite number only

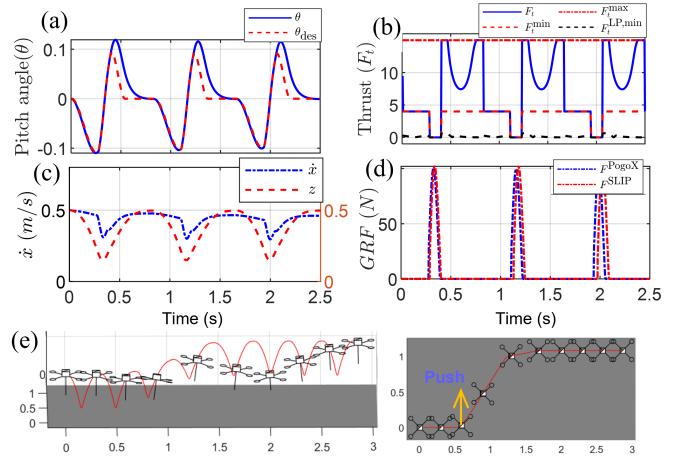


Fig. 5. Simulation results: trajectories of the leg pitch angles (a), the applied thrust forces and the bounds (b), the horizontal velocity and vertical height (c), and GRFs (d) of a periodic hopping of PogoX with a desired forward velocity of 0.5m/s; (e) shows the trajectory under an external push force.

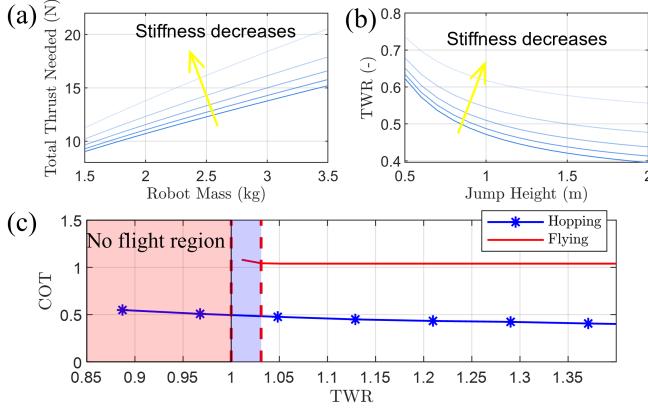


Fig. 6. (a, b) Illustration of the effects of spring stiffness on periodic hopping behaviors: (a) the maximum thrust needed for different robot weights to achieve the jump height at 1.3m; (b) the TWR required to realize certain hopping height. The stiffness is increased from $2e+3\text{N/m}$ to $1e+4\text{N/m}$ with $2e+3\text{N/m}$ increments, and the damping coefficient is assumed to be 15Ns/m . (c) Comparison of the COT of hopping v.s. flying with a 1m/s horizontal velocity under different TWR.

when $\text{TWR} > 1$. As the TWR decreases close to 1 (blue region), the controller struggles to maintain flying. At $\text{TWR} \leq 1$, flying is no longer feasible, producing zero velocity and thus an infinity on the COT; in reality, the TWR for realizing flying is typically chosen to be 2 or larger. For jumping, with $\text{TWR} > 0.8$, the robot is able to hop to the desired height, thus producing a finite valued COT. It is thus clear that hopping is necessary with $\text{TWR} \leq 1$ in order to locomote, and if the horizontal velocity can be realized on hopping, hopping is more efficient than flying at $\text{TWR} > 1$ as well. Note that aerial dynamics effects are neglected as we assume low-speed locomotion.

B. Hardware

The same controller is realized on the hardware. As the robot weighs 2.5kg, the maximum thrust is manually set to 22N so that $\text{TWR} < 1$. The control is implemented in ROS running at 200Hz. Visual-inertia-odometry (VIO) is realized using VINS-Fusion [38] to estimate the velocity and orientation of the robot during hopping. For indoor testing where the VIO alone lacks robustness due to the scenes in the environment, we use a motion-capture system to get measurements of the global position and velocity for control implementation. Since the thrust force comes from the propeller rotation, it naturally cannot deviate rapidly from the applied value at previous time stamp; we set $p = 0$ in (12) and the resultant QP has a closed-form solution. In the experiment, the human operator holds a tether to only provide safety for the robot when it is E-stopped. Results can be seen in the video <https://youtu.be/XmN6uNW69H4> [39].

Periodic Hopping: The controller is primarily implemented on the hardware to produce periodic hopping with different horizontal velocities. The desired apex height of the main body is set to 1.1m, which corresponds to a desired equivalent energy level at 10.2J. The robot is able to stabilize in place and also to desired target velocities.

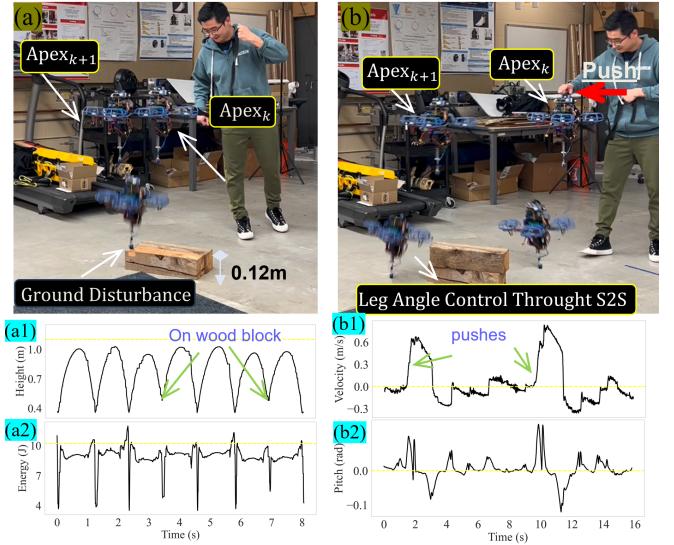


Fig. 7. (a) The robot hops from the level ground onto a wood block that is 0.12m tall with the apex height remaining roughly the same: (a1) the vertical height of the robot; (a2) the energy level. (b) The robot is pushed in the air: (b1) the horizontal velocity stabilization; (b2) the pitch angle of the robot. The desired values are denoted in yellow.

Robustness to Pushes and on Uneven Terrain: We also manually provide external pushes from the human and place wood blocks on the floor in the experiments, which represent unmodelled disturbances. It is shown that the controller can stabilize the system quickly. With the elevation from wood blocks, the hopping height remains the same due to the vertical energy controller. With the pushes, the horizontal velocities are stabilized with one step, which validates our S2S dynamics based deadbeat controller. Fig. 7 shows the time-lapsed images of the experiments and their associated trajectories. Both the vertical energy and horizontal velocities are stabilized with adequate robustness.

VI. CONCLUSIONS AND FUTURE WORK

To conclude, we present a controller design to enable terrestrial hopping on the robot PogoX, which is a quadrotor with a spring-leg appendage at the bottom. The key advantage of the novel system with the controller design is to enable locomotion capability on flying robots with a thrust-to-weight ratio (TWR) of less than 1. The controller is an effective combination of control Lyapunov function (CLF) driven synthesis and a Spring-loaded Inverted Pendulum (SLIP) inspired step-to-step (S2S) dynamics control. Dynamic hopping behaviors with robustness are realized in both simulation and experiment.

In the future, we are interested in enabling PogoX to locomote outdoors in the natural environment, which will first require methodically unifying all the available advancing sensing technologies for accurate state estimation. We are also interested in exploring the optimal operations of PogoX with its flying and hopping modalities to navigate in complex and challenging environments with the goal of increasing its energy efficiency and payload capacity.

REFERENCES

- [1] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, “A survey on aerial swarm robotics,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.
- [2] D. C. Schedl, I. Kurmi, and O. Bimber, “An autonomous drone for search and rescue in forests using airborne optical sectioning,” *Science Robotics*, vol. 6, no. 55, p. eabg1188, 2021.
- [3] A.-a. Agha-mohammadi, N. K. Ure, J. P. How, and J. Vian, “Health aware stochastic planning for persistent package delivery missions using quadrotors,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3389–3396.
- [4] T. Elmokadem, “Distributed coverage control of quadrotor multi-uav systems for precision agriculture,” *IFAC-PapersOnLine*, vol. 52, no. 30, pp. 251–256, 2019.
- [5] B. Balaram, T. Canham, C. Duncan, H. F. Grip, W. Johnson, J. Maki, A. Quon, R. Stern, and D. Zhu, “Mars helicopter technology demonstrator,” in *2018 AIAA atmospheric flight mechanics conference*, 2018, p. 0023.
- [6] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.
- [7] B. Siciliano, O. Khatib, and T. Kröger, *Springer handbook of robotics*. Springer, 2008, vol. 200.
- [8] M. Chignoli, D. Kim, E. Stanger-Jones, and S. Kim, “The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors,” in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2021, pp. 1–8.
- [9] C. J. Pratt and K. K. Leang, “Dynamic underactuated flying-walking (duck) robot,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3267–3274.
- [10] B. Zhu, J. Xu, A. Charway, and D. Saldaña, “Pogodrone: Design, model, and control of a jumping quadrotor,” in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2031–2037.
- [11] K. Kim, P. Spieler, E.-S. Lupu, A. Ramezani, and S.-J. Chung, “A bipedal walking robot that can fly, slackline, and skateboard,” *Science Robotics*, vol. 6, no. 59, p. eabf8136, 2021.
- [12] H. T. Suh, X. Xiong, A. Singletary, A. D. Ames, and J. W. Burdick, “Energy-efficient motion planning for multi-modal hybrid locomotion,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 7027–7033.
- [13] D. Pucci, S. Traversaro, and F. Nori, “Momentum control of an underactuated flying humanoid robot,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 195–202, 2017.
- [14] S. Bouabdallah and R. Siegwart, “Full control of a quadrotor,” in *2007 IEEE/RSJ international conference on intelligent robots and systems*. Ieee, 2007, pp. 153–158.
- [15] H. Geyer, A. Seyfarth, and R. Blickhan, “Compliant leg behaviour explains basic dynamics of walking and running,” *Proc. Royal Soc. B*, vol. 273, no. 1603, pp. 2861–2867, 2006.
- [16] M. Shahbazi, R. Babuška, and G. A. Lopes, “Unified modeling and control of walking and running on the spring-loaded inverted pendulum,” *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1178–1195, 2016.
- [17] C.-J. Hu, C.-K. Huang, and P.-C. Lin, “A torque-actuated dissipative spring loaded inverted pendulum model with rolling contact and its use as the template for design and dynamic behavior generation on a hexapod robot,” in *2015 IEEE Int. Conf. on Rob. and Autom. (ICRA)*, pp. 5177–5183.
- [18] P.-B. Wieber, R. Tedrake, and S. Kuindersma, “Modeling and control of legged robots,” in *Springer handbook of robotics*. Springer, 2016, pp. 1203–1234.
- [19] X. Xiong and A. D. Ames, “Bipedal hopping: Reduced-order model embedding via optimization-based control,” in *IEEE/RSJ Int. Conf. on Intell. Rob. and Sys. (IROS)*, 2018, pp. 3821–3828.
- [20] W.-S. Yang, W.-C. Lu, and P.-C. Lin, “Legged robot running using a physics-data hybrid motion template,” *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1680–1695, 2021.
- [21] X. Xiong and A. D. Ames, “Dynamic and versatile humanoid walking via embedding 3d actuated slip model with hybrid lip based stepping,” *IEEE Rob. and Automation Letters*, vol. 5, no. 4, pp. 6286–6293, 2020.
- [22] A. D. Ames, K. Galloway, K. Sreenath, and J. Grizzle, “Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics,” *IEEE Trans. on Autom. Control*, vol. 59, no. 4, pp. 876–891, 2014.
- [23] X. Xiong and A. Ames, “3-d underactuated bipedal walking via h-lip based gait synthesis and stepping stabilization,” *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2405–2425, 2022.
- [24] J. E. Seipel and P. Holmes, “Running in three dimensions: Analysis of a point-mass sprung-leg model,” *The International Journal of Robotics Research*, vol. 24, no. 8, pp. 657–674, 2005.
- [25] P. A. Bhounsule, J. Cortell, A. Grewal, B. Hendriksen, J. D. Karssen, C. Paul, and A. Ruina, “Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge,” *The Int. J. of Rob. Res.*, vol. 33, no. 10, pp. 1305–1321, 2014.
- [26] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, “Models, feedback control, and open problems of 3d bipedal robotic walking,” *Automatica*, vol. 50, no. 8, pp. 1955–1988, 2014.
- [27] T. Luukkonen, “Modelling and control of quadcopter,” *Independent research project in applied mathematics, Espoo*, vol. 22, no. 22, 2011.
- [28] X. Xiong and A. Ames, “Slip walking over rough terrain via h-lip stepping and backstepping-barrier function inspired quadratic program,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2122–2129, 2021.
- [29] G. Colin, J. Byrnes, Y. Sim, P. Wensing, and J. Ramos, “Whole-body dynamic telelocomotion: A step-to-step dynamics approach to human walking reference generation,” *arXiv preprint arXiv:2307.10533*, 2023.
- [30] M. Hutter, C. D. Remy, M. A. Höpflinger, and R. Siegwart, “Slip running with an articulated robotic leg,” in *2010 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys. (IROS)*, pp. 4934–4939.
- [31] A. D. Ames and M. Powell, “Towards the unification of locomotion and manipulation through control lyapunov functions and quadratic programs,” in *Control of Cyber-Physical Systems*. Springer, 2013, pp. 219–240.
- [32] H. K. Khalil, “Nonlinear systems,” *Prentice-Hall, New Jersey*, 1996.
- [33] Y. Wang, N. Dehio, A. Tanguy, and A. Kheddar, “Impact-aware task-space quadratic-programming control,” *arXiv preprint arXiv:2006.01987*, 2020.
- [34] P. M. Wensing and D. E. Orin, “Generation of dynamic humanoid behaviors through task-space control with conic optimization,” in *IEEE Int. Conf. on Rob. and Autom. (ICRA)*, 2013, pp. 3103–3109.
- [35] J. Lee, E. Bakolas, and L. Sentis, “Hierarchical task-space optimal covariance control with chance constraints,” *IEEE Control Systems Letters*, vol. 6, pp. 2359–2364, 2022.
- [36] P. M. Wensing and D. E. Orin, “High-speed humanoid running through control with a 3d-slip model,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5134–5140.
- [37] H. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, “qpOASES: A parametric active-set algorithm for quadratic programming,” *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [38] T. Qin, S. Cao, J. Pan, and S. Shen, “A general optimization-based framework for global pose estimation with multiple sensors,” *arXiv preprint arXiv:1901.03642*, 2019.
- [39] Simulation Video: [On-line] 2023.09. Available: <https://youtu.be/XmN6uNW69H4>.