SCIENCE ROBOTICS | RESEARCH ARTICLE

HOPPING LOCOMOTION

An agile monopedal hopping quadcopter with synergistic hybrid locomotion

Songnan Bai[1]†, Qiqi Pan[2,3]†, Runze Ding[1]†, Huaiyuan Jia[1], Zhengbao Yang[2,3], Pakpong Chirarattananon[1,3,4]*

Nature abounds with examples of superior mobility through the fusion of aerial and ground movement. Drawing inspiration from such multimodal locomotion, we introduce a high-performance hybrid hopping and flying robot. The proposed robot seamlessly integrates a nano quadcopter with a passive telescopic leg, overcoming limitations of previous jumping mechanisms that rely on stance phase leg actuation. Based on the identified dynamics, a thrust-based control method and detachable active aerodynamic surfaces were devised for the robot to perform continuous jumps with and without position feedback. This unique design and actuation strategy enable tuning of jump height and reduced stance phase duration, leading to agile hopping locomotion. The robot recorded an average vertical hopping speed of 2.38 meters per second at a jump height of 1.63 meters. By harnessing multimodal locomotion, the robot is capable of intermittent midflight jumps that result in substantial instantaneous accelerations and rapid changes in flight direction, offering enhanced agility and versatility in complex environments. The passive leg design holds potential for direct integration with conventional rotorcraft, unlocking seamless hybrid hopping and flying locomotion.

INTRODUCTION

Many organisms in nature achieve hybrid locomotion through the integration of jumping and flying behaviors. Arboreal foraging birds such as parrotlets and sparrows exhibit high agility by combining jumps between branches with intermittent flights (*1*). Certain species of widowbirds perform ritualized display jumps for advertising fitness and defending territories to prospective mates during breeding seasons (*2*). Insects such as locusts, froghoppers, and fleas integrate jumping and flying behaviors to navigate complex environments and evade predators (*3*–*5*). Flying squirrels (*6*) and Draco lizards (*7*) also perform hybrid jumping and gliding to achieve long jump distances. These examples demonstrate the potential benefits of hybrid jumping-flying locomotion for improving mobility, agility, efficiency, and versatility.

Developing a hybrid robot that can achieve continuous jumping and flying is challenging, particularly when considering the limitations and complexity of conventional jumping mechanisms. Although a few combustion-driven robotic jumpers have been reported (*8*–*11*), previous robots capable of jumping are primarily based on either latched (*12*–*30*) or unlatched (*31*–*37*) actuation mechanisms. The former uses a catch mechanism to enable rapid energy release from mounted elastomer, making varying jump height and continuous hopping difficult. When the explosive energy release is passively triggered (via cam escapement or pawl-and-rachet mechanisms, for instance) (*12*–*16*, *20*, *23*–*25*, *27*–*30*), the jump height is fixed according to the predetermined tension, limiting the robot's capability. On the other hand, active triggering permits adjustable jump heights, but continuous hopping remains elusive because precisely timing the release to coincide with landing introduces a stringent coordination requirement (*18*, *19*, *21*, *26*).

In contrast, jumpers with unlatched actuation mechanisms can directly drive their legs with actuators (*31*, *38*–*40*), compatible with continuous hopping with variable jump height. However, without energy storage, they cannot generate sufficient instantaneous power for high jumps. To address this limitation, elastomers are used in parallel (*35*, *37*) or series-elastic (*32*–*34*) mechanisms to temporarily store the energy produced by the actuator and amplify the output power during the stance phase. With carefully designed linkages and nonlinear spring stiffness, these robots benefit from variable mechanical advantage, demonstrating agile continuous hopping with superior vertical jumping agility (*22*, *32*–*34*). However, they necessitate a relatively complex mechanical structure, higher actuator power, and non-negligible stance time (usually more than 57 ms; see table S1), restricting hopping agility because the robot must spend a certain amount of time in stance for the actuator to inject energy into the system.

Previous attempts to create robots capable of both jumping and flight have directly combined jumping mechanisms with aerial platforms (*16*–*19*, *41*–*43*). This allows the robots to rapidly gain altitude through jumping-assisted takeoff. However, most cannot achieve continuous hopping, fine-tune jump height, or sustain flight.

In this article, we present Hopcopter, a hybrid hopping-flying robot that augments a micro–aerial vehicle with a passive elastic telescopic leg (Fig. 1A and Movie 1). This results in a mechanically simple and efficient design that directly leverages the existing rotors for jumping. The passive elastic leg differentiates Hopcopter from previous jumping robots. The absence of direct leg actuation reduces stance duration, leading to higher jumping frequencies and agility. To showcase this, we constructed a 35-g prototype and identified and characterized the robot's hopping dynamics. On the basis of the modeling results, we developed a model-based hopping controller to stabilize the robot during hopping

---

[1]Department of Biomedical Engineering, City University of Hong Kong, Tat Chee Avenue, Hong Kong SAR, China. [2]Department of Mechanical and Aerospace Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong SAR, China. [3]Department of Mechanical Engineering, City University of Hong Kong, Tat Chee Avenue, Hong Kong SAR, China. [4]Centre for Nature-inspired Engineering, City University of Hong Kong, Tat Chee Avenue, Hong Kong SAR, China.
*Corresponding author. Email: pakpong.c@cityu.edu.hk
†These authors contributed equally to this work.

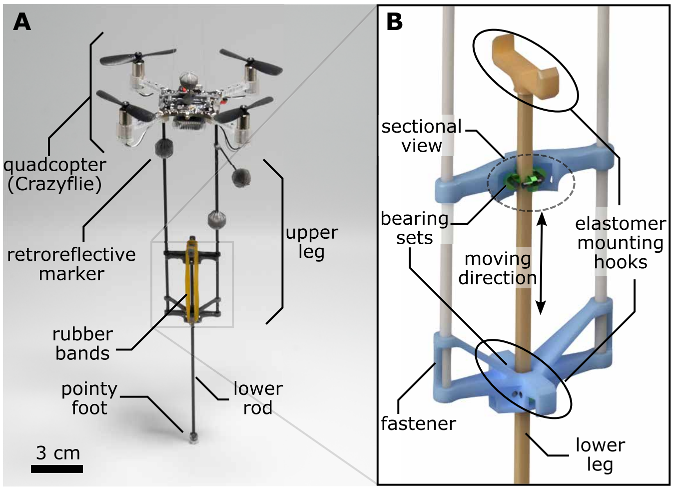
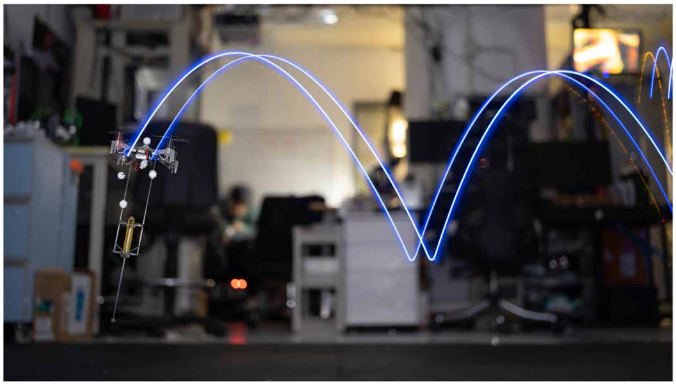


Fig. 1. Hopcopter prototype. (**A**) Photograph of the hybrid flying-hopping robot. The robot consists of a micro-quadcopter and a passive telescopic leg. (**B**) A 3D drawing illustrating the retractable elastic leg mechanism. The sectional view illustrates the configuration of the bearing set (green components) as the support for the lower leg. The rubber bands are not shown in (B).



Movie 1. Our hybrid hopping and flying robot seamlessly transitions between aerial and terrestrial locomotion modes. The hybrid locomotion offers the robot boosted agility and endurance. When equipped with the aerodynamic stabilizer, the robot is capable of performing outdoor operations without requiring position feedback.

using offboard position measurements. Experimental results show that the robot accurately tracked a reference trajectory, validating our three-dimensional (3D) hopping model and illustrating that the Hopcopter's average vertical hopping speeds approach the theoretical ballistic limit, surpassing state-of-the-art jumping robots, including Salto-1P (*33*). Intermittent hops during flight enhance instantaneous accelerations for rapid and tight turns. This multimodal locomotion capability has the potential to expand the range of environments in which robots can operate effectively and efficiently, from cluttered indoor spaces to rugged outdoor terrains. To attain stable hopping using onboard sensors only, we outfitted the robot with an active aerodynamic stabilizer. These surfaces dynamically influence landing attitude, stabilizing lateral hopping speed and attitude without external feedback or vision.

RESULTS

Hopcopter platform

Hopcopter is a hybrid hopping and flying vehicle. The robot comprises a micro-quadcopter (27 g; Bitcraze, Crazyflie 2.1) and a telescopic leg (Fig. 1A). The telescopic leg features a rigidly fixed upper section, a sliding rod as a lower section, and rubber bands acting as the elastic element. The upper leg is made of two carbon fiber rods (diameter: 1.8 mm) rigidly mounted below the quadcopter via 3D printed fasteners (Formlab Form 3, Tough 2000). The lower leg (yellow in Fig. 1B), made of a single carbon-fiber rod (2 mm by 2 mm square cross section), a pointy foot, and 3D printed hooks (for mounting the rubber bands), constitutes the main moving components of the hopping mechanism. The lower leg connects to the upper part via two sets of bearings (VXB 681 with 1-mm inner diameter), restricting motion to vertical translation and minimizing sliding friction. In this configuration, the axis of translation passes through the center of mass (CoM) of the robot. Two rubber bands (diameter ≈ 1.5 mm and ring radius ≈ 20 mm) anchor between the upper leg and the tip of the lower leg as the elastic element. The rubber bands are installed such that they are prestretched (even without leg contraction), and they further elongate when the leg contracts. In addition, four reflective spherical markers are affixed on the robot to obtain the pose measurements from a motion capture system. The entire Hopcopter weighs 34.8 g, less than the maximum takeoff weight of 42 g.

Notably, our robot is actuated in the aerial phase rather than the stance phase, enabling continuous hopping and adjustable jump height. During each hop, the passive telescopic leg absorbs and restores kinetic energy through elastic recoil, as the thrust-based actuation compensates for energy loss and stabilizes attitude and velocity. This distinguishes our approach from those relying on latched elastic actuation, which imposes limitations on continuous hopping and variable jump heights, and from those dependent on unlatched series elastic actuation, which require a longer stance time. Although a similar design has been proposed (*44*), the robot therein was limited to isolated jumps performed intermittently between flights and did not display continuous hopping locomotion.

Our design incorporates two key features to simplify the stance phase dynamics. First, the long telescopic leg (22 cm from the foot to the CoM of the robot) is more than twice the width of the airframe. Second, the rubber bands are prestretched, enabling the elastic force to overpower the robot's weight [the prestretch is absent from the leg of the quadrotor capable of intermittent jumps reported in (*44*)]. These properties allow the robot to be treated as a point mass during the stance phase and decouple the dynamics from the gravity's orientation as detailed below.

Hopping dynamics

In flight, Hopcopter's dynamics are characterized by propelling torque (roll, pitch, and yaw) and collective thrust, indistinguishable from a regular rotorcraft (*45*–*47*). However, when functioning as a monopedal hopper, the robot is a hybrid dynamical system, with one jumping cycle alternating between the aerial and terrestrial stages, separated by the ground contact (*48*–*50*). For modeling purposes, the movement of the CoM of the hybrid robot in each jumping cycle is divided into aerial (AP) and stance (SP) phases, separated by two key time stamps: landing (t_{LD}) and takeoff (t_{TO}) as illustrated in Fig. 2A.

In AP, the robot shares the same flight dynamics with a regular quadcopter with six degrees of freedom in the form of rotational and translational motion. Upon landing, the foot collision marks the transition between AP and SP. Under the condition of no slippage between the foot and the ground (*48*, *49*), the attitude (described by a rotation matrix $\mathbf{R}$) and position $\mathbf{p}$ of the robot are physically coupled. As a spring-loaded inverted pendulum (SLIP) system (*49*, *51*) (with the robot abstracted as a point mass instead of a rigid body), the ground contact point acts as a pivot for the body attitude, with the distance to the CoM varying with the leg contraction as illustrated in Fig 2 (A and B).

Thanks to the prestretched rubber bands, the elastic force (described by Eq. 7) passes through the CoM and dominates the stance dynamics. In the absence of active propelling torque, the angular momentum of the robot is conserved throughout the SP. Thus, the robot rotates about a particular axis $\mathbf{e}_\psi$ (defined by Eq. 11) determined by the landing velocity $\dot{\mathbf{p}}(t_{\mathrm{LD}})$ and landing attitude $\mathbf{z}_b(t_{\mathrm{LD}})$, where $\mathbf{z}_b$ denotes the primary axis of the robot, as depicted in Fig. 2B.

By separately modeling the angular and axial dynamics of the robot, the rotational angle ψ (measured with respect to $\mathbf{e}_\psi$) and the leg contraction of the robot from landing to takeoff can be evaluated according to the landing state: landing speed $\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$ and $\mathbf{z}_b(t_{\mathrm{LD}})$ (in the form of θ_{LD}, the angle between vectors $-\dot{\mathbf{p}}(t_{\mathrm{LD}})$ and $\mathbf{z}_b(t_{\mathrm{LD}})$) as shown in Fig. 2B. This enables the takeoff state to be predicted from the landing state as detailed in Materials and Methods. The takeoff state is described by $\Delta\psi$, θ_{TO}, and $\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\|$. $\Delta\psi$ denotes the angular displacement during the entire stance duration, measured with respect to $\mathbf{e}_\psi$; θ_{TO} is the angle between the takeoff velocity $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ and attitude $\mathbf{z}_b(t_{\mathrm{TO}})$ as seen in Fig. 2B; and $\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\|$ is the takeoff speed. After parameter identification (detailed in Materials and Methods), the mapping from $\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$ and θ_{LD} to $\Delta\psi$, θ_{TO}, and $\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\|$ is presented in Fig. 2C. This relationship completely defines the state transition from landing to takeoff. The maps suggest that for a given $\dot{\mathbf{p}}(t_{\mathrm{LD}})$, $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ can be directly manipulated by modifying θ_{LD} in the form of $\mathbf{z}_b(t_{\mathrm{LD}})$. This forms the basis of the model-based hopping controller.

Model-based hopping controller

Similar to other spring-mass monopeds, the passive jumping dynamics of the Hopcopter is unstable and lossy. Stabilizing the attitude of the robot in the aerial phase to guarantee an upright landing attitude is insufficient to prevent the horizontal speed from diverging after consecutive jumps (*48*, *52*, *53*). Concomitantly, energy must be injected into the system to compensate for viscous losses in both stance and aerial phases. However, unlike monopedal hoppers with direct leg actuation [for instance, linear hydraulic and hip actuators in (*54*) or a series-elastic actuator in (*33*)], the stance phase of the Hopcopter is completely passive. This hopping mechanism demands a dedicated control framework.

The proposed controller for the Hopcopter differs from widely used policies (*33*–*35*, *53*) based on Raibert's hopping machine (*54*) and other jumping controllers (*35*, *37*, *39*, *48*, *52*, *55*). Our controller allows the robot to take off in a desired direction in three dimensions. This capacity enables the robot to track trajectories while hopping and facilitates smooth transitions between aerial and terrestrial

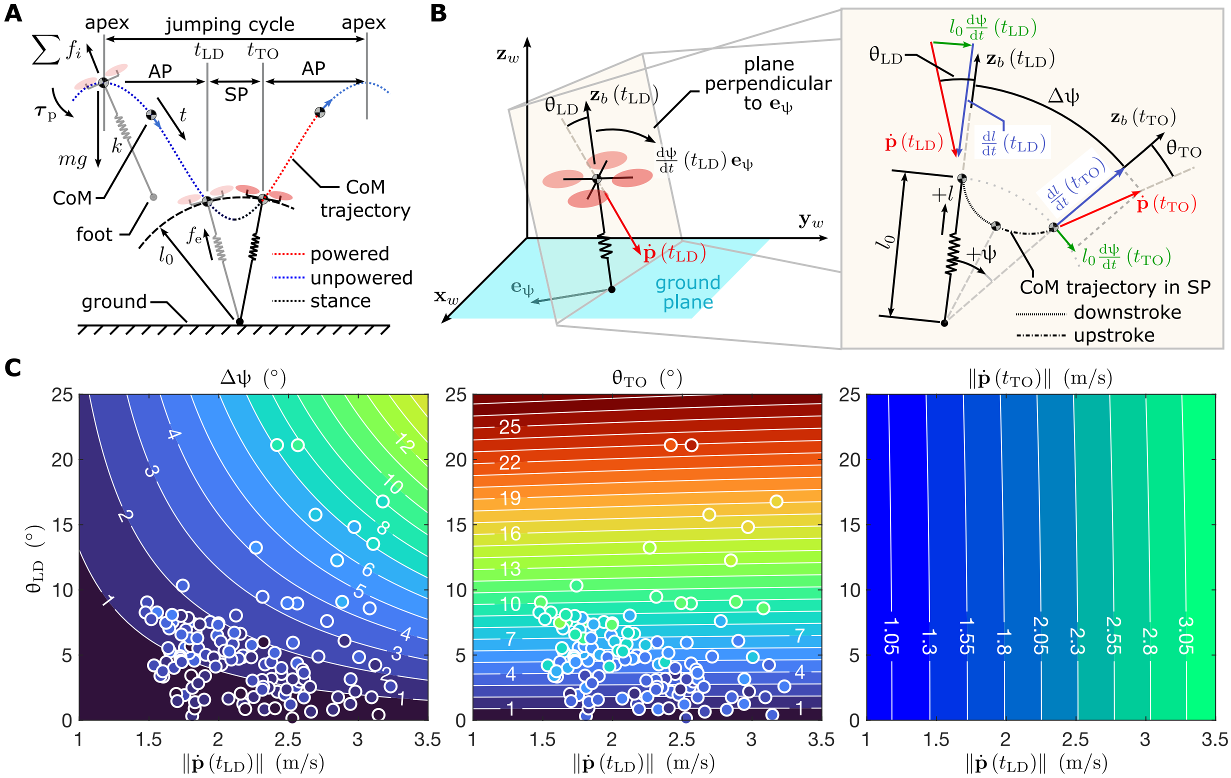


Fig. 2. Hopping dynamics of the robot. (**A**) A schematic diagram illustrating the dynamics during the aerial and stance phases of a complete jumping cycle. (**B**) Illustration of the stance phase dynamics from the moment of landing (t_{LD}) to the point of takeoff (t_{TO}), showing the geometric relationships between the landing velocity, landing attitude, takeoff attitude, and takeoff velocity. (**C**) Takeoff state prediction based on the identified stance phase dynamics. The 130 data points show the measured landing and takeoff states of the robot from the controlled hopping experiment, with the color depth representing the values of these states.

modes, enabling complex hybrid locomotion. For instance, during high-speed flight, Hopcopter leverages the ground contact or environment to generate an instantaneous burst of acceleration and hop to decelerate, turn, or accelerate, notably improving agility beyond flight alone.

The control strategy takes advantage of the derived model of the stance dynamics to decrease the control effort and decouple the altitude control problem from the landing control problem as much as practically possible as detailed in Materials and Methods. As a result, the trajectory of the robot in the aerial phase is mostly ballistic, and the position control is accomplished through the control of $\mathbf{z}_b(t_{\mathrm{LD}})$, in a similar manner to the control of the touchdown angle in (*33*). This differs from a previous jumping quadrotor (*44*), in which the position control is directly implemented during the aerial phase using a flight controller designed for a rotorcraft. Our framework renders the hopping locomotion markedly more efficient than flying. In addition, the elastic force during the stance phase overpowers the robot's weight by an order of magnitude (see fig. S1D) and surpasses the propulsive thrust. Exploiting this considerable elastic force through the leg allows superior position control and jumping ability.

Hopping tests and model verification

To validate the identified parameters and the derived dynamic model, we performed a preliminary controlled hopping experiment with the developed controller. The robot was commanded to hop at varying altitude set points in the range of 0.5 to 0.7 m and translate laterally up to 1 m per step for approximately 90 s. According to the recorded landing and takeoff states, the values of $\Vert\dot{\mathbf{p}}(t_{\mathrm{LD}})\Vert$, θ_{LD}, $\Delta\psi$, and θ_{TO} from 130 consecutive jumps were computed as detailed in Materials and Methods. We used the devised model to predict the takeoff states in terms of $\Delta\psi$ and θ_{TO} and compared the predictions with the measurements as presented in Fig. 2C [note that $\Vert\dot{\mathbf{p}}(t_{\mathrm{TO}})\Vert$ cannot be accurately measured because of the implementation of powered climbing]. The measurements are highly aligned with the predictions, with the root mean square errors (RMSEs) of only 1.0° and 1.6° for $\Delta\psi$ and θ_{TO}. The consistency between the model predictions and measurements confirms that the derived equations and associated assumptions are suitable for capturing the stance dynamics of the Hopcopter.

Next, the tracking performance was evaluated using circular and step trajectories. For the circular trajectory, the robot was instructed to hop at a constant altitude (0.60 m) with a radius of 1.20 m at an average speed of 0.2 m/s. The step trajectory demanded the robot to maintain a jump height of 0.75 m and laterally translate by up to 2 m several times in 40 s. Three repeated trials were conducted for each trajectory (movie S1). The resultant trajectories are shown in Fig. 3 (A to C). The tracking performance was determined by calculating the RMSEs of three repeated tests: The RMSE of the landing position is 16 cm for the circular trajectory and 40 cm for the step trajectory. For the jump height, the RMSEs are 4.1 and 4.8 cm, respectively. Figure 3B suggests that the amplified error of the step trajectory is due to the inability to respond promptly to sudden changes of the set point position. These results demonstrate that the proposed controller can achieve accurate trajectory tracking with low altitude error and moderate landing error, even with large displacements between steps and the absence of full flight control during the aerial phase.

Hopping agility

Next, we conducted an experiment to investigate the relationship between the hopping height h and the frequency f of the robot and to calculate its hopping agility, defined as the cycle-averaged vertical hopping speed of the robot: $\nu = 2hf$. The definition of hopping agility ν presented here deviates from the vertical jumping agility defined in (*32*); the latter refers to the vertical climbing speed. The robot was commanded to hop in place for more than 60 s using the developed position controller. The test was divided into seven segments, each corresponding to a different hopping height ranging from 0.59 to 1.63 m. In each segment, the robot hopped at a constant height for at least six cycles (fig. S2 and movie S2). We then measured and plotted the period of one hopping cycle for each segment and plotted it against the corresponding height (see Fig. 4A). This was compared with the period predicted by the model, which can be broken down into $t_{\mathrm{SP}} = t_{\mathrm{TO}}$

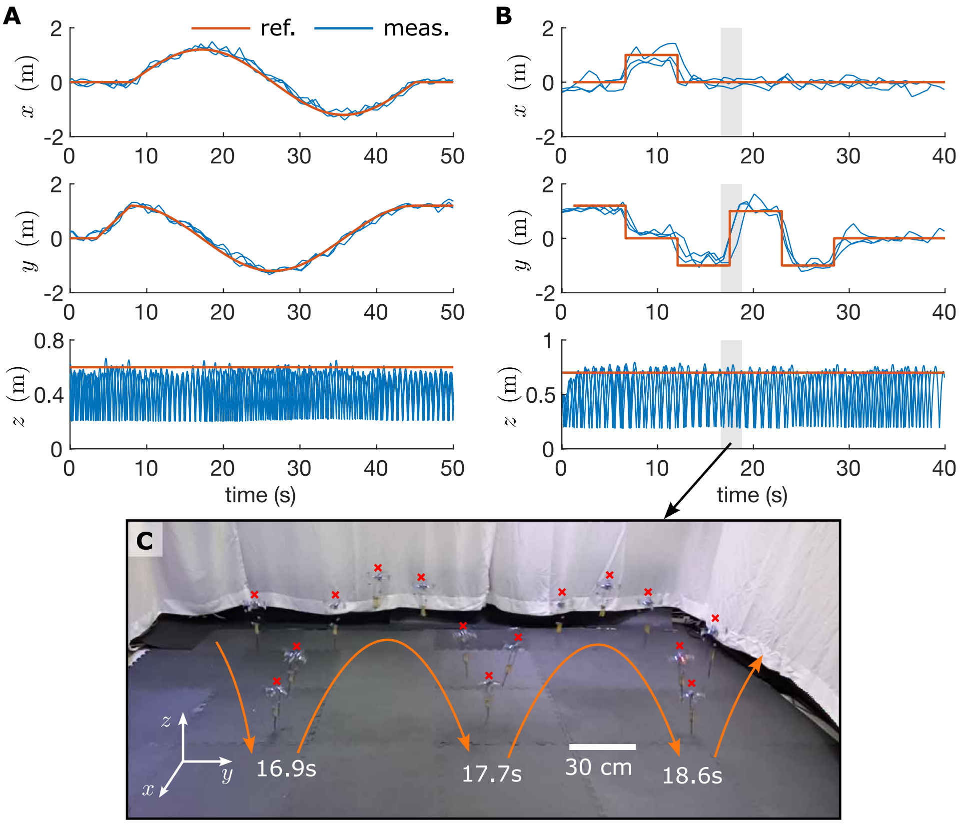


Fig. 3. Trajectory tracking experiments in the hopping mode. (**A**) Reference and realized circular trajectories. (**B**) Reference and realized step trajectories. For each experiment, three repeated trials were conducted and included. (**C**) A composite photo showing the trajectory of the robot from 16.6 to 18.8 s while tracking the step trajectory.

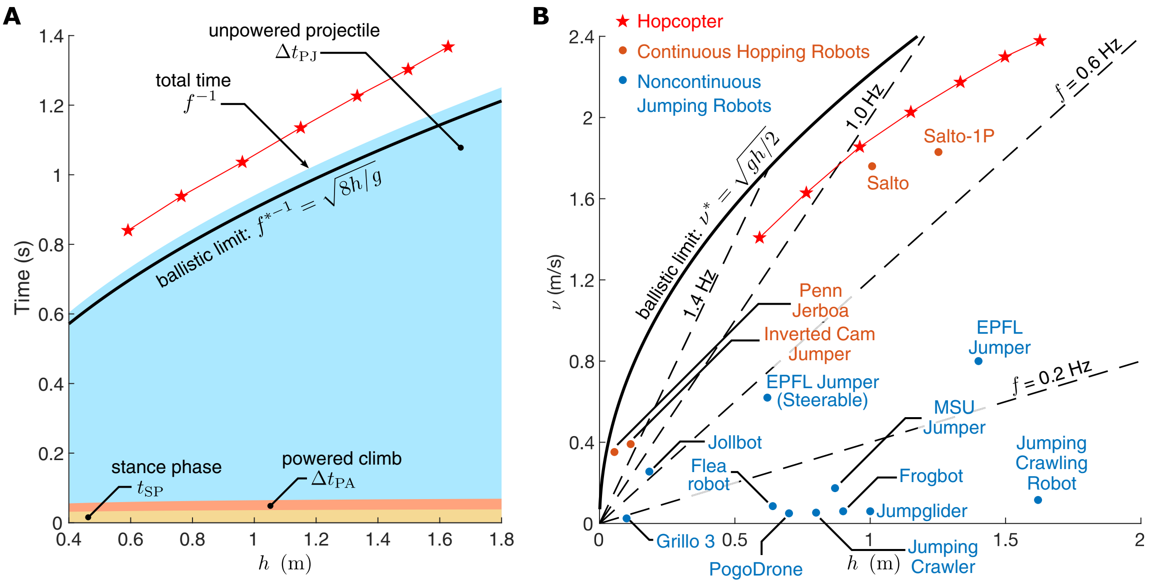


Fig. 4. Jumping height, frequency, and hopping agility. (**A**) Plot of the predicted and measured hopping period against hopping height. The period consists of the stance phase, the powered climb, and the unpowered projectile segment. (**B**) Hopping agility versus hopping height. The plot includes the Hopcopter and other jumping robots.

$- t_{\text{LD}}$ (as defined in eq. S18 in Supplementary Methods), Δt_{PA} (as defined in Eq. 27), and Δt_{PJ} (as defined in Eq. 29). In addition, we identified the ballistic limit $f^{*-1} = \sqrt{8h/g}$ (*32*), which occurs when the robot spends negligible time in the stance phase and the entire flight phase is ballistic, where g is the gravitational acceleration.

As seen in Fig. 4A, the model prediction of the jumping period is approximately 40 ms longer than the ballistic limit, which approximately corresponds to the time the robot spends in the stance phase. In addition, the measured periods are approximately 130 ms longer than the model prediction. This discrepancy is mainly attributed to the fact that the model neglects the air drag and thrust generated for the attitude control effort when the robot is descending. These prolong the time in the aerial phase. Therefore, we can infer that these factors collectively lengthen the robot's hopping period compared with the theoretical limit.

We computed the hopping agility as the product of the height and frequency. Because of the short stance time (<45 ms; refer to Supplementary Methods), the Hopcopter achieved a hopping agility of 2.38 m/s when $h = 1.63$ m, which is 30% higher than 1.83 m/s for 1.25 m attained by Salto-1P (with a stance time of 57 ms for a lower jump height) (*32*, *33*) and other jumping robots (Fig. 4B and table S1). All in all, the Hopcopter exploits the natural dynamics of a springy telescopic leg with high stiffness to minimize the ground contact time (see Supplementary Methods), which enables high agility because of the reduced stance time during hopping.

Synergistic hybrid locomotion for agile maneuvers

Strategically combining hopping and flying can enhance maneuverability and agility because the hopping mode can generate bursts of acceleration for flight. By momentarily switching to hopping mode during flight for a single jump, the monopedal robot can generate an instantaneously large acceleration from the ground reaction force. Using the developed hopping model, the takeoff direction is regulated through the landing attitude. This operation assists the robot in decelerating, accelerating from hover, or turning tightly (movie S3), providing improved agility despite the limited thrust-to-weight ratio of only 1.2.

The first maneuver demonstrates the use of elastic leg and ground reaction force to accelerate and then decelerate the robot in flight. As captured in Fig. 5 (A and B), the robot was initially commanded to hover. To accelerate forward, the hopping mode was activated and the robot entered a free fall (red shading in Fig. 5B). Once landed at a pitch angle of 14°, the passive stance dynamics induced an impulsive takeoff with the forward speed of 1.14 m/s with less than 45 ms spent in the SP, producing the lateral SP acceleration of 26.0 m/s^2. The landing and takeoff velocities, as well as the SP acceleration for this and subsequent experiments, were calculated on the basis of an estimated SP duration of 45 ms, as detailed in Supplementary Methods. Including the fall period (0.50 s as indicated by shadings in Fig. 5B and fig. S3), the average lateral acceleration was 2.2 m/s^2. After the liftoff, the robot continued its flight and reached the forward speed of more than 1.99 m/s before executing the second jump to land at a pitch angle of 19° and deftly decelerate to lateral speed of 0.05 m/s. The sudden stop resulted in a lateral SP deceleration of 43.5 m/s^2 and an average deceleration of 3.3 m/s^2 (including the falling period of 0.52 s as marked in Fig. 5B and fig. S3).

Next, the agility is illustrated through rapid and tight turns. The robot leveraged the ground to swiftly change the travel direction by 90° as depicted in Fig. 5 (C and D). Beginning with the flight speed of 1.9 m/s, the leg-assisted right turn altered the flight course in the space of 0.54 s (from falling to completion of the SP). The robot landed with the lateral speed of 1.63 m/s and resulted in the post jump speed of more than 1.23 m/s as shown in fig. S4. This equates to the lateral SP acceleration of 44.4 m/s^2 (in 45 ms) and average acceleration of 4.2 m/s^2. The large SP acceleration resulted in a sharp right-angle trajectory instead of a curve as captured by Fig. 5C. An even higher acceleration was achieved when the hybrid locomotion was implemented for suddenly reversing the flight direction (fig. S5). The jump with a pitch angle of 28° at landing entirely turned the flight around in 0.55 s (from falling to completion of SP), from the lateral

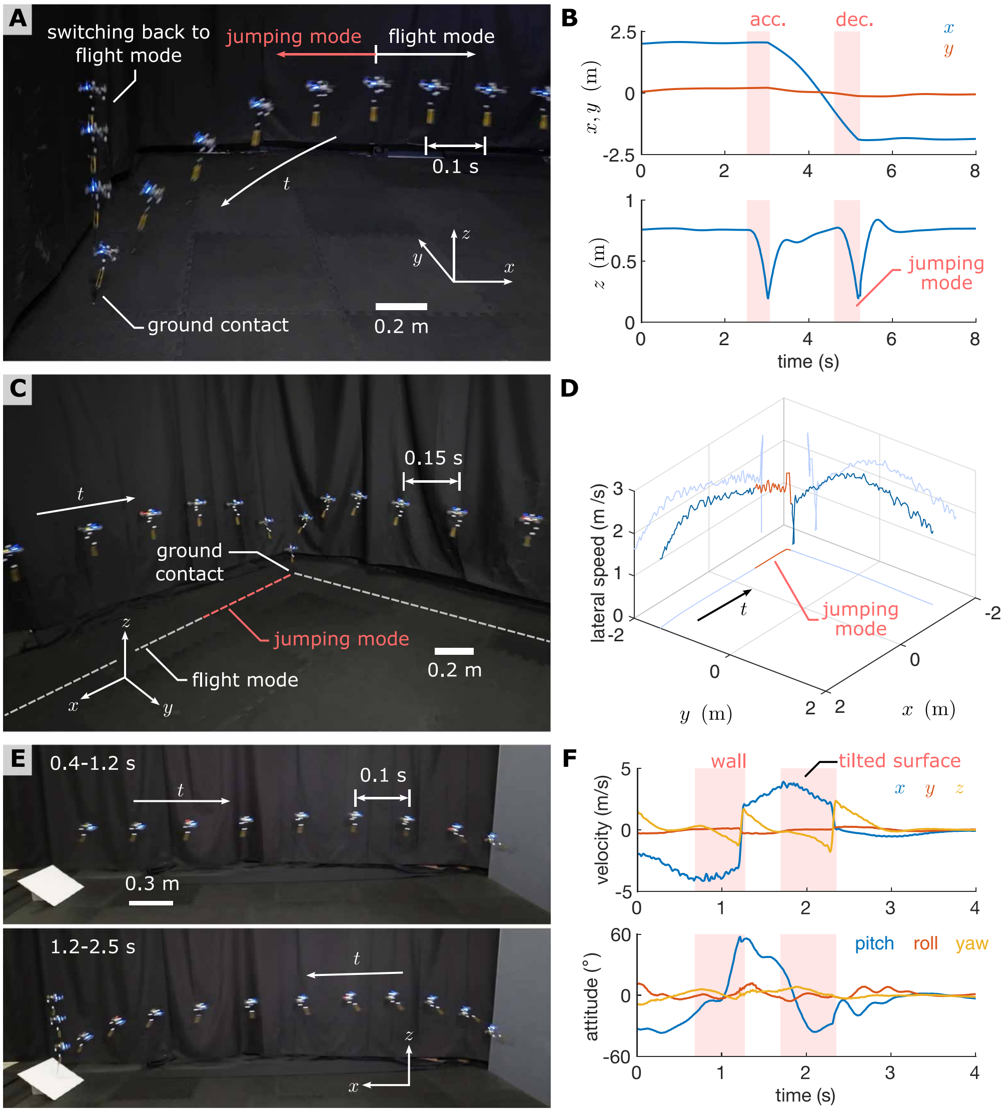


Fig. 5. Demonstration of hybrid jumping and flying locomotion. (**A** and **B**) A composite image and the trajectory of the robot showing the robot making use of the ground to accelerate from hover and decelerate from lateral translation. Panel (A) only demonstrates the deceleration from 4.4 to 5.5 s. (**C** and **D**) A composite image and flight data showing the robot using ground contact to turn 90°. Panel (D) demonstrates that the turning maneuver does not require prior deceleration. (**E** and **F**) Two sequential composite images and data showing the robot jumping on a wall and a tilted surface.

landing velocity of 1.79 m/s to the post jump speed of 1.46 m/s in the opposite direction, equating to a lateral SP acceleration of 72.3 m/s^2 (in 45 ms) and an average lateral acceleration of 6.9 m/s^2.

To further push the agility limit, we realized hybrid jumps on a wall and a tilted surface. Nonhorizontal surfaces extend the range of nonslip landing and takeoff angles (see Supplementary Methods). This effectively raises the achievable lateral acceleration. In the experiment, the robot flew toward a wall at a speed more than 4 m/s. Within 0.54 s or 1.8 m before reaching the surface, the attitude controller reoriented the robot for landing, and the robot touched down on the wall with a tilt (pitch) angle of 57° and landing speed of 3.6 m/s (Fig. 5, E and F, and fig. S6). The wall jump reversed the lateral speed of the robot to 2.68 m/s within 45 ms. Immediately after the bounce, the robot spent 0.43 s accelerating toward another inclined surface (30°) and executed the second jump in 0.63 s (from falling to completion of SP) to decelerate the flight from 3.9 m/s to end the maneuver with a hover. The average lateral accelerations in these two periods are 11.4 and 5.7 m/s^2, whereas the lateral SP accelerations reached 139.8 and 66.6 m/s^2.

The results demonstrate that the jump-flying locomotion increases the average and instantaneous lateral accelerations compared with those achieved by flight alone, especially considering the robot's modest thrust-to-weight ratio of only 1.2. The Hopcopter accomplished instantaneous and average accelerations of more than 14g and 1.2g by using the ground reaction force and the elastic leg in intermittent jumps. This is substantially higher than the reported accelerations of other micro aerial robots with similar or higher thrust-to-weight ratios (*46*, *47*, *56*). Hybrid locomotion is particularly advantageous for robots with limited thrust, such as small and lightweight platforms, because it allows them to perform rapid and tight maneuvers that would otherwise be impossible, offering versatility in navigating complex trajectories.

Self-stabilized hopping via active aerodynamic stabilizer

According to the proposed hopping dynamic model, in hopping, the takeoff velocity of the robot is directly dependent on landing attitude and velocity. This implies that to stabilize the robot, a velocity measurement as feedback is necessary. To reduce the robot's dependence on sensors and external measurements (such as GPS or feedback from motion capture cameras), we proposed an actuated aerodynamic stabilizer to interact with the airflow to stabilize the robot. The strategy to enable the robot to hop in place without position or velocity feedback involves the adjustment of θ_{LD} based on the robot's velocity $\dot{\mathbf{p}}$ when falling using both the aerodynamic force and active control. This is by appropriately balancing the attitude controller with the rotational torque created by the stabilizer.

Aerodynamic stabilizer

The 4.9 g of detachable stabilizer for the Hopcopter consists of three horizontally hinged surfaces (Fig. 6A). Each with an area of 39 cm^2 (refer to Supplementary Methods for sizing), these hinged surfaces can be made rigid by pulling on the cables connected to a servomotor (H-King 282AS) or allowed to swing freely in response to airflow

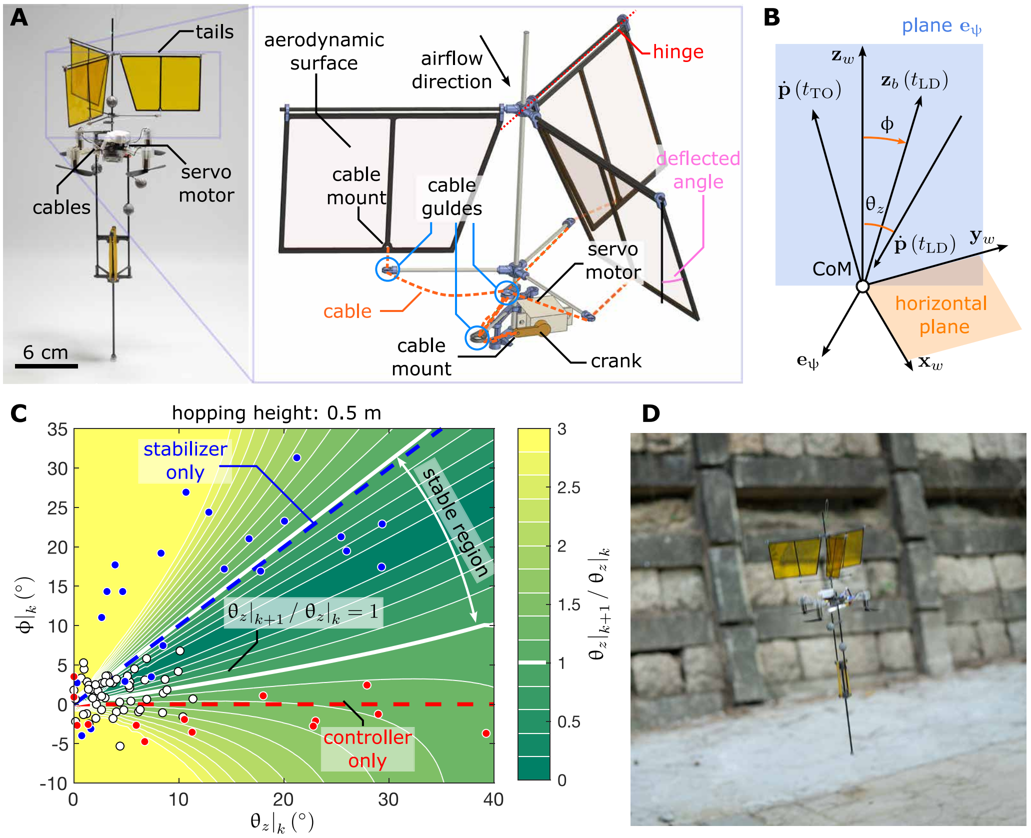


Fig. 6. Detachable active aerodynamic stabilizer and Poincaré map. (**A**) A photo of the robot equipped with the aerodynamic stabilizer and a drawing detailing the structure and actuation of the stabilizer. (**B**) Diagram displaying the relationships between landing velocity, landing attitude, and takeoff velocity. Here, θ_z is the angle between the vertical axis $\mathbf{z}_w$ and the negative landing velocity $-\dot{\mathbf{p}}(t_{\mathrm{LD}})$. The angle ϕ represents the landing attitude defined as the signed angle between the robot's primary axis $\mathbf{z}_b$ and the vertical at landing. The orange arrows illustrates the positive direction of ϕ. (**C**) Poincaré map depicting the evolution of the landing velocity angle θ_z in terms of $\theta_z|_{k+1}/\theta_z|_k$, as a function of $\theta_z|_k$ and $\phi|_k$ for hopping at 0.5 m. The white solid lines indicate the boundaries between stable and unstable regions. Circular dots represent the recorded landing state in the hopping stability test. Red, blue, and white colors represent the measurements when only the attitude controller, aerodynamic stabilizer, or both mechanisms were active. (**D**) Photo of the robot during the outdoor hopping experiment. The aerodynamic surfaces were deflected under aerodynamic load.

when the cables slacken (movie S4). Tightening the cables activates the stabilizer, enabling it to bear aerodynamic load and generate torque to steer the robot's attitude on the basis of airflow. Conversely, loosening the cables deactivates the stabilizer, allowing the surfaces to deflect from the vertical by up to 15° (see fig. S7 and movie S4). In this configuration, the aerodynamic force and torque are substantially diminished unless the robot undergoes aggressive maneuvers. This cable-activated mechanism sets our stabilizer apart from aerodynamic surfaces found in other robotic platforms, which typically are either rigidly affixed (*57*, *58*, *59*, *60*) or actively steerable (*61*). Our design renders the aerodynamic force negligible when the cables are loosened to keep the dynamics during the ascent phase unaffected.

Damper-mediated stability

During hopping, the servo-driven cable is relaxed to disengage the stabilizer and eliminate the aerodynamic damping forces while ascending. This allows the robot to maintain full control over its attitude, aligning itself to an upright orientation for the power climb. The stabilizer is activated during the descent. It uses the upward airflow to create a torque that rotates the robot's attitude and reduces θ_{TD} by directing its major axis to point in the opposite direction of its translational velocity [according to the flat plate theory (*62*–*64*)]: $\mathbf{z}_b \to -\dot{\mathbf{p}}$. In the meantime, the attitude controller applies a propelling torque to make the robot upright $\mathbf{z}_b \to \mathbf{z}_w = \mathbf{e}_3$. As a consequence of the competing efforts, the robot lands with its major axis $\mathbf{z}_b$ pointing between the vertical axis and the velocity vector $-\dot{\mathbf{p}}(t_{\mathrm{LD}})$ (see Fig. 6B). By tuning the control gains and the stabilizer configuration, the hopping dynamics can be stabilized without velocity measurements. That is, the Hopcopter gains the ability to operate in real-world environments using only the feedback from its onboard inertial sensor.

To manifest the damper-assisted stability, we numerically constructed Poincaré maps assuming a constant hopping height of 0.5 m. The landing state was chosen as the fixed point of the limit cycle. Under the influence of the attitude controller and the aerodynamic stabilizer, the vectors $\mathbf{z}_w$, $\mathbf{z}_b$, and $\dot{\mathbf{p}}(t_{\mathrm{LD}})$ are coplanar. The angle between the landing velocity and the vertical (see Fig. 6B) is

$$\theta_z = \arccos\left(-\mathbf{z}_w^{\mathrm{T}}\dot{\mathbf{p}}(t_{\mathrm{LD}}) / \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|\right) \tag{1}$$

The angle quantifies the horizontal speed that should converge to zero over repeated hops in the case of stable hopping. In addition to θ_z, the landing attitude ϕ, defined as the signed angle between $\mathbf{z}_b$ and the vertical at landing, affects the takeoff and subsequent landing states. ϕ is positive, as seen in Fig. 6B, when

$$\arccos(-\mathbf{z}_b^{\mathrm{T}}\dot{\mathbf{p}}(t_{\mathrm{LD}}) / \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|) < \theta_z \tag{2}$$

and negative otherwise. We leveraged the model of the stance dynamics to predict the next landing state for any combination of θ_z and ϕ as described in Materials and Methods.

The Poincaré maps plot $\theta_z|_{k+1}/\theta_z|_k$ for different $\theta_z|_k$ and $\phi|_k$ at a particular hopping height, where k represents the hopping cycle (Fig. 6C and fig. S8). The maps reveal the stable regions ($\theta_z|_{k+1}/\theta_z|_k < 1$), establishing the conditions for converging hopping speed. Notably, the red dashed line in the Poincaré maps corresponds to $\phi|_k = 0$, representing the scenario in which the attitude controller dominates and the robot consistently lands with an upright orientation. On the other hand, the blue dashed line depicts cases where the stabilizer dominates, aligning the landing velocity and attitude ($\theta_z|_k = \phi|_k$). These dashed lines lie near the boundaries of the stable region, indicating that reliance on either strategy alone would not result in stability with sufficient robustness. However, the Poincaré maps exhibit a large overlap between the stable region and the area between the dashed lines. By tuning the control gains and stabilizer configuration, the interplay between the controller and the mechanism stabilizes the hopping speed and attitude over repeated jumps without velocity measurements.

Hopping stability test

To substantiate the insights from the Poincaré analysis, we conducted three sets of experiments evaluating the performance of the hopping robot under each control strategy (attitude control and aerodynamic stabilizer) individually, as well as their combination (movie S5). For each set of experiments, the robot started to hop after a drop from 75 cm. We tracked the hopping trajectory to determine the robot's attitude and velocity in the landing phase.

When using only the attitude controller (red dashed line), the robot was expected to land consistently in an upright orientation or $\phi = 0$ but without any guarantee on the horizontal hopping speed or θ_z. Experimental results confirmed this (Fig. 6C and fig. S9), showing that the robot could hop for three to five cycles before the horizontal speed diverged as $\theta_z|_{k+1}/\theta_z|_k > 1$.

Relying solely on the stabilizer (the attitude controller was only active when the robot is ascending) was predicted to marginalize the difference between θ_z and ϕ but not directly reduce θ_z. Again, experiments validated this (Fig. 6C and fig. S9). Some discrepancies between the model prediction ($\theta_z = \phi$) and the results reflect a limited effectiveness of the aerodynamic stabilizer, which is dependent on the size of the stabilizer and the time the robot spent descending. Overall, the robot displayed poor stability and crashed after 6 to 10 hopping cycles.

In contrast, experiments integrating both the attitude controller and stabilizer yielded highly robust hopping as theorized. The robot retained near-zero horizontal velocity component and near-upright attitude across multiple successive jumps (white dots in Fig. 6C and fig. S9), achieving more than 56 hops in 45 s within a 3 m–by–3 m arena in the absence of position and velocity feedback.

Field hopping demonstration

To further demonstrate the robustness and reliability of the stabilizing strategy, we conducted several field tests in varying environments, including stairs, a narrow corridor, and outdoor terrain (see Fig. 6D and movie S6). In each test, a human operator commanded the desired landing attitude for the controller, although the actual landing posture depended on both the attitude setpoint and aerodynamic stabilizer as previously described. The adjusted desired landing attitude results in a bias that directs the robot to hop toward the prescribed direction. This control scheme allowed the Hopcopter to ascend/descend a flight of stairs, navigate the confined corridor space, and traverse rough terrain. Through these field experiments, we verify that the active stabilizer enables stable and resilient mobility without requiring vision or additional sensors for velocity feedback. Unlike wheeled aerial vehicles that struggle to negotiate rough terrains (*65*–*71*), the hopping ability allows the robot to overcome obstacles and traverse uneven surfaces naturally.

Endurance and power efficiency

Unlike the aerial mode, the Hopcopter generates thrust intermittently rather than continuously during the hopping mode. This mode of operation presents an opportunity to substantially extend the robot's operation time while reducing power consumption. To evaluate the endurance, we conducted a continuous jumping test with a fully charged 250-mAh single-cell Li-ion battery (8.3 g). The robot hopped continuously for more than 1246 s (20.8 min). With a 650-mAh Li-ion battery (16.8 g), the endurance increased to more than 3052 s (50.9 min). However, the total mass of the robot with a high-capacity battery (43.1 g) exceeded the maximum takeoff weight for flight (42.0 g). Battery voltages recorded onboard during the tests are shown in fig. S10B. The terrestrial locomotion shows a four- to eightfold increase in endurance in comparison with the benchmark quadcopter (32.1 g; Crazyflie 2.1 with retroreflective markers but without the leg), which only hovers for 379 s with a 250-mAh battery (movie S7).

The lack of a reliable onboard current sensor and the weight limitation rendered direct measurement of power consumption impractical. Nevertheless, we used duty cycle of the motors as a metric to compare the flight and terrestrial operations. This is because the robot is required to constantly generate $T \approx mg$ when hovering, with m being the mass of the robot, whereas it only generates $T = mg$ momentarily after each takeoff while hopping. Defining the vehicle-based duty ratio where the summation $\sum f_i$ represents the net thrust

$$D = (1/T)\int_0^T \sum (f_i / mg)\mathrm{d}t \tag{3}$$

We obtain $D = 1$ for a benchmark quadcopter flight. In the endurance test, Hopcopter spent more than 85% of the time in the ballistic and stance phases with minimal thrust for attitude control ($\sum f_i < 0.11mg$ outside the powered climb; see fig. S10A). For these reasons, the overall duty ratio across 1246 s was only 0.28, implying more than a threefold improvement in efficiency and endurance. In comparison, the hybrid quadcopter robots, PogoDrone (*44*) and LEO (*72*), saved only 20 and 44% power, respectively, via jumping and walking locomotion compared with flying, with a duty ratio exceeding 0.5.

DISCUSSION

The Hopcopter is a hybrid robot that combines hopping and flying capabilities to achieve high-performance locomotion. Its passive telescopic leg distinguishes it from previous bioinspired jumping robots that rely on latched or unlatched elastic actuation because it dispenses the need for a trigger, actuation in stance, or variable mechanical advantage, making it mechanically simpler and more reliable. The thrust-based actuation of the passive telescopic leg allows shorter stance phase and higher hopping frequencies, resulting in its high agility. The robot achieves a maximum velocity of 2.38 m/s at a jump height of 1.63 m, exceeding the highest agility reported in jumping robots (Fig. 4B and table S1), including Salto-1P (*32*, *33*, *73*). The use of thrust permits fine-tuning of jump height. The Hopcopter demonstrated jumps from 0.6 to 1.6 m in experiments, showcasing its wide jump height range.

The Hopcopter's ability to both fly and hop facilitates rapid changes in flight speed and direction. The robot's jumping ability can be leveraged to mitigate damage from collisions with walls or ground by using jumping maneuvers (movie S8). The integration of attitude control and aerodynamic stabilizer establishes the robot's hopping control autonomy. Poincaré analysis and experiments show that the attitude controller and aerodynamic stabilizer work together to stabilize the Hopcopter's hopping dynamics. Without position feedback, the robot achieved more than 56 consecutive jumps with near-zero velocity and upright landing.

Compared with rotorcraft, flight-capable bipedal and legged robots (*72*, *74*–*76*), the hopping mechanism requires less energy for locomotion. The marked increase in power efficiency and endurance translates into tangible advantages for the Hopcopter's operational capabilities, specifically in terms of mission duration and range. Similar to perching (*63*, *64*, *77*–*82*), the longer operation time allows the robot to remain active for extended periods, which is valuable in scenarios such as surveillance, inspection, or exploration, where sustained presence is desired. Moreover, the improvement in endurance could potentially extend the robot's travel distance, expanding its operational range as demonstrated in other robotic flyers capable of terrestrial locomotion (*66*, *71*, *83*, *84*).

Crucially, the passive leg design of Hopcopter could be readily applied to conventional rotorcraft. With a simple addition of an elastic leg and control changes, a standard quadcopter could transform into a hopping robot, unlocking synergistic hybrid locomotion. For instance, a payload that would be too heavy for continuous flight could still be transported in a series of hops. Overall, the Hopcopter represents a platform that displays the benefits of integrating flight and hopping capabilities.

MATERIALS AND METHODS

Aerial dynamics

When flying, the robot is modeled as a rigid body with weight mg and inertia tensor $\mathbf{I} = \text{diag}(I_x, I_y, I_z)$ in a gravity field. The lightweight telescopic leg is abstracted as a thin rod with a preloaded elastomer. The leg axis passes through the CoM of the robot and aligns with the thrust direction of the propellers as schematically illustrated in Fig. 2 (A and B).

In the aerial phase, the equations of motion are identical to those of a regular multirotor vehicle (*45*–*47*, *85*) and given by

$$m\ddot{\mathbf{p}} = \mathbf{z}_b \sum f_i - \mathbf{e}_3 mg \tag{4}$$

$$\boldsymbol{\tau}_\text{p} = \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \tag{5}$$

where $\mathbf{p} = [x, y, z]^\text{T}$ denotes the position vector of the CoM of the robot in the inertial frame $\{\mathbf{x}_w, \mathbf{y}_w, \mathbf{z}_w\}$. The z axis of the body-fixed frame $\{\mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b\}$, as seen in the inertial frame, represents the thrust direction of the propellers: $\mathbf{z}_b = \mathbf{R}\mathbf{e}_3$, where $\mathbf{R} \in \text{SO}(3)$ is a rotation matrix mapping the body-fixed frame to the inertial frame and $\mathbf{e}_3 = [0,0,1]^\text{T}$ is a basis vector. $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^\text{T}$ is the body-centric angular velocity. The summation $\sum f_i$ denotes the total thrust magnitude and $\boldsymbol{\tau}_\text{p}$ is the total propelling torque in the body frame. Together $\sum f_i$ and $\boldsymbol{\tau}_\text{p}$ are control inputs that can be commanded by distributing the power between the four rotors. In a regular flight regime, aerodynamic drag can be neglected (*46*, *85*).

Stance dynamics and landing transition

Once landed (time stamp t_LD in Fig. 2A), the ground reaction force brings about the leg contraction, decreasing the leg length $l(t)$ from the nominal state of $l_0 = 22$ cm. Without any influence from the propelling thrust, the SP dynamics is completely passive. The main robot body interacts with the ground through the telescopic leg in the form of the restoring force created by the elastomer f_e. Because the robot can be treated as a point mass (*33*, *48*, *49*) instead of a rigid body ($I_x, I_y, I_z \ll ml_0^2$, thanks to the long leg length as discussed in Supplementary Methods), the robot follows the equation of motion

$$m\ddot{\mathbf{p}} = -\mathbf{e}_3 mg + \mathbf{z}_b f_\text{e} \tag{6}$$

On the basis of the measurements from the drop tests (see details below and fig. S1D), the elastic force f_e is linearly proportional to the deformation of the elastomer $l - l_0 - l_\text{p}$ by a constant k, with a constant damping force of magnitude f_c

$$f_\text{e} = -k(l - l_0 - l_\text{p}) - \text{sgn}(\dot{l})f_\text{c} \quad \text{for} \quad l < l_0 \tag{7}$$

where $l - l_0$ is the leg contraction and l_p is the prestretched length of the elastomer. The friction term accounts for the dissipated energy (likely attributed to the bearing friction and the loss in the elastomer) that results in a hysteresis behavior brought by the loading/downstroke ($\dot{l} < 0$) and unloading/upstroke ($\dot{l} > 0$) motion (see fig. S1D). By design, the preload force kl_p dominates the weight of the robot ($kl_\text{p} = 3.35$ N versus $mg = 0.34$ N; see details below and table S2). Therefore, the stance dynamics is predominated by f_e, and Eq. 6 simplifies to

$$m\ddot{\mathbf{p}} = \mathbf{z}_b f_\text{e} \tag{8}$$

Assuming no slippage (thanks to the pointy foot), the ground contact point can be regarded as a free-to-rotate spherical hinge, and the CoM of the robot is restricted to the translation along the leg axis and the rotation about the foot as shown in Fig. 2 (A and B). Because the elastic force f_e is always along the axial direction, the angular momentum measured with respect to the ground contact in the SP is conserved, and the trajectory of the CoM can be separately described as the axial and angular motion and is independent of the gravity direction. Denoting the landing velocity as $\dot{\mathbf{p}}(t_\text{LD})$ and $\theta_\text{LD} \in [0, \pi)$ as the angle between $\dot{\mathbf{p}}(t_\text{LD})$ and the leg axis $\mathbf{z}_b(t_\text{LD})$, we yield the leg contraction speed at touchdown

$$\dot{l}(t_\text{LD}) = \mathbf{z}_b(t_\text{LD}) \cdot \dot{\mathbf{p}}(t_\text{LD}) = -\|\dot{\mathbf{p}}(t_\text{LD})\|\cos\theta_\text{LD} \tag{9}$$

As long as the body-centric angular velocity immediately before landing is minimized, the angular speed of the CoM $\dot{\psi}(t_{\mathrm{LD}})$ with respect to the ground is the component of $\dot{\mathbf{p}}(t_{\mathrm{LD}})$ in the direction orthogonal to $\mathbf{z}_b$ or

$$l_0\dot{\psi}(t_{\mathrm{LD}}) = \|\dot{\mathbf{p}}(t_{\mathrm{LD}}) \times \mathbf{z}_b(t_{\mathrm{LD}})\| = \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\| \sin\theta_{\mathrm{LD}} \tag{10}$$

As illustrated in Fig. 2B, the subsequent angular motion is characterized by the rotational axis defined by

$$\mathbf{e}_\psi = -\dot{\mathbf{p}}(t_{\mathrm{LD}}) \times \mathbf{z}_b(t_{\mathrm{LD}}) / \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\| \tag{11}$$

perpendicular to both $\dot{\mathbf{p}}(t_{\mathrm{LD}})$ and $\mathbf{z}_b(t_{\mathrm{LD}})$. The axial $\dot{l}(t_{\mathrm{LD}})$ and angular $\dot{\psi}(t_{\mathrm{LD}})$ speeds become the initial conditions of the robot in the SP.

Stance phase and takeoff transition

Unlike previous studies of the 3D SLIP models (*48*, *52*), the complexity of the stance dynamics of the Hopcopter is further reduced owing to the strategic design decision to substantially prestretch the elastomer. As detailed in Supplementary Methods, when the directions of $\dot{\mathbf{p}}(t_{\mathrm{LD}})$ and $\mathbf{z}_b(t_{\mathrm{LD}})$ are largely aligned, the centrifugal acceleration is negligible, and the SP is governed by the axial and angular dynamics in the form of two semi-coupled differential equations

$$m\frac{\mathrm{d}}{\mathrm{d}t}\dot{l} = -k(l - l_0 - l_{\mathrm{p}}) - \mathrm{sgn}(\dot{l})f_{\mathrm{c}} \tag{12}$$

$$m\frac{\mathrm{d}}{\mathrm{d}t}(l^2\dot{\psi}) = 0 \tag{13}$$

Given the initial conditions ($\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$ and θ_{LD}), the axial dynamics is independent of ψ and dictated by the natural frequency $\Omega = \sqrt{k/m}$. We obtain an analytical solution of $l(t)$ and the t_{TO} corresponding to the moment when the elastic force vanishes ($l(t_{\mathrm{TO}}) = l_0$) as detailed in Supplementary Methods. With the expression of $l(t)$, Eq. 13 states that

$$\dot{\psi}(t) = l_0^2\dot{\psi}(t_{\mathrm{LD}}) / l^2(t) \tag{14}$$

The time integration

$$\Delta\psi = \int_{t_{\mathrm{LD}}}^{t_{\mathrm{TO}}} \psi(t)dt \tag{15}$$

provides the attitude of the robot at the takeoff as the rotation of $\mathbf{z}_{\mathrm{b}}(t)$ about the axis $\mathbf{e}_\psi$ in stance phase (t_{SP}, from t_{LD} to t_{TO}, see Fig. 2B). Therefore, the change in the body attitude $\mathbf{z}_{\mathrm{b}}(t)$, in terms of $\Delta\psi$, in the entire SP is also fully determined from $\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$ and θ_{LD}. Reprojecting to the inertial frame, the takeoff attitude of the robot is given by

$$\mathbf{R}(t_{\mathrm{TO}}) = \mathbf{R}(\mathbf{e}_\psi, \Delta\psi)\mathbf{R}(t_{\mathrm{LD}}) \tag{16}$$

where $\mathbf{R}(\mathbf{e}_\psi, \Delta\psi)$ is a rotation matrix based on the axis-angle representation. Letting

$$\mathbf{z}_b(t_{\mathrm{TO}}) = \mathbf{R}(t_{\mathrm{TO}})\mathbf{e}_3 \tag{17}$$

be the primary body axis at t_{TO}, the takeoff velocity $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ is obtained by projecting the axial $\dot{l}(t_{\mathrm{TO}})$ and tangential speeds $l_0\dot{\psi}(t_{\mathrm{TO}})$ of the robot at t_{TO} to the inertial frame. This results in

$$\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\| = \sqrt{l_0^2\dot{\psi}^2(t_{\mathrm{TO}}) + \dot{l}^2(t_{\mathrm{TO}})} \tag{18}$$

with the direction specified by the angle between $\mathbf{z}_b(t_{\mathrm{TO}})$ and $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ given by

$$\theta_{\mathrm{TO}} = \arctan(l_0\dot{\psi}(t_{\mathrm{TO}}) / \dot{l}(t_{\mathrm{TO}})) \tag{19}$$

In the vector form, this is given by

$$\dot{\mathbf{p}}(t_{\mathrm{TO}}) = \dot{l}(t_{\mathrm{TO}})\mathbf{z}_b(t_{\mathrm{TO}}) + l_0\dot{\psi}(t_{\mathrm{TO}})\mathbf{e}_\psi \times \mathbf{z}_b(t_{\mathrm{TO}}) \tag{20}$$

$$= \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|\mathbf{R}(\mathbf{e}_\psi, \theta_{\mathrm{TO}})\mathbf{z}_b(t_{\mathrm{TO}}) \tag{21}$$

As illustrated in Fig. 2B, because of the conservation of angular momentum, vectors $\mathbf{z}_b(t_{\mathrm{LD}})$, $\dot{\mathbf{p}}(t_{\mathrm{LD}})$, $\mathbf{z}_b(t_{\mathrm{TO}})$, and $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ are all coplanar and perpendicular to $\mathbf{e}_\psi$, and the takeoff velocity is directly influenced by the landing attitude. Therefore, the jumping trajectory can be controlled through the manipulation of $\mathbf{z}_b(t_{\mathrm{LD}})$. With the identified model coefficients ($\Omega = \sqrt{k/m}$, f_{c}/m, and l_{p} in table S2) from the drop tests (see details below and movie S9), the takeoff state of the robot specified by $\Delta\psi$, θ_{TO}, and $\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\|$ is predicted according to the landing state [θ_{LD} and $\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$] in Fig. 2C.

Overview of the hopping controller

Because the hopping motion is cyclic and the aerial phase of the trajectory is primarily ballistic, we regulate the landing attitude of the current hopping cycle (indexed k) to yield the desired landing location in the subsequent hop (indexed $k + 1$; as shown in fig. S11A). The proposed method strategically decouples the attitude, altitude, and lateral dynamics of the robot by neglecting the aerodynamic drag and exploiting the ability of the robot to adeptly command its attitude in the aerial phase. We leveraged the developed dynamic model to create a set of equations that predict the landing point at time step $k + 1$ from the landing attitude at time step k and relied on these equations to numerically solve for the set point landing attitude that brings about the desired landing location at time step $k + 1$. The structure of the controller, consisting of five submodules, labeled (i) to (v), is outlined in fig. S11B. Starting from the jumping cycle k, fig. S11B (i) predicts the landing location and velocity. This information, together with a prespecified landing attitude and a hopping altitude set point, are used to predict the takeoff state of cycle k and landing location of cycle $k + 1$ in fig. S11B (ii). In fig. S11B (iii), a numerical optimizer relies on the output of fig. S11B (ii) to form an algebraic loop (red in fig. S11B) that determines the desired landing attitude of cycle k and the thrust requirement during the climb in cycle k to reach the hopping altitude setpoint and the next landing location in cycle $k + 1$. Last, fig. S11B (iv) is a thrust and attitude manager that converts the outcome from the solver in fig. S11B (iii) to command inputs for the low-level controller fig. S11B (v). The detailed implementations of the controller are as follows.

Landing state prediction

The high-level hopping controller loop is executed once per jump cycle. Starting from the apex of cycle k, the total thrust of the Hopcopter is nominally set to zero, and the unpowered robot follows a ballistic trajectory as illustrated in fig. S11A. From the current position $\mathbf{p}(t)$ and velocity $\dot{\mathbf{p}}(t)$ at time stamp t, the time to touchdown is determined from

$$z = \mathbf{e}_3^{\mathrm{T}} \mathbf{p}(t) \text{ and } \dot{z} = \mathbf{e}_3^{\mathrm{T}} \dot{\mathbf{p}}(t) \tag{22}$$

given by

$$\Delta t_{\mathrm{LD}} = t_{\mathrm{LD}} - t = \dot{z}(t)/2 + \sqrt{2(z(t) - l_0)/g + \dot{z}^2(t)/g^2} \tag{23}$$

The landing location and velocity are

$$\mathbf{p}(t_{\mathrm{LD}})\,|_k = \mathbf{p}(t) + \dot{\mathbf{p}}(t)\Delta t_{\mathrm{LD}} - \mathbf{e}_3 \frac{1}{2} g \Delta t_{\mathrm{LD}}^2 \tag{24}$$

$$\dot{\mathbf{p}}(t_{\mathrm{LD}})\,|_k = \dot{\mathbf{p}}(t) - \mathbf{e}_3 g \Delta t_{\mathrm{LD}} \tag{25}$$

During the descent, the attitude of the robot is controlled with the collective thrust and angular velocity minimized, allowing the motion to be treated as free fall from an altitude of $z(t) - l_0$.

Altitude control and next landing location

On the basis of the landing set point attitude $\mathbf{z}_b(t_{\mathrm{LD}})$ (or $\mathbf{R}(t_{\mathrm{LD}})\mathbf{e}_3$) and the computed landing velocity $\dot{\mathbf{p}}(t_{\mathrm{LD}})$, the stance phase dynamics (Eqs. 12 and 13) is integrated forward to produce the liftoff state $\mathbf{R}(t_{\mathrm{TO}})$ and $\dot{\mathbf{p}}(t_{\mathrm{TO}})$ using Eqs. 16 and 20 with

$$\theta(t_{\mathrm{LD}}) = \arccos(-\mathbf{z}_b^{\mathrm{T}}(t_{\mathrm{LD}})\dot{\mathbf{p}}(t_{\mathrm{LD}})/\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|) \tag{26}$$

Immediately after taking off (detected by a notable change in the axial acceleration), the low-level controller (v) (described below) swiftly reorients the robot to an upright orientation ($\mathbf{z}_b = \mathbf{e}_3$) to minimize the horizontal thrust. Thus, the landing position of cycle $k + 1$ is entirely determined by the lateral component of the liftoff velocity ($[\mathbf{e}_1, \mathbf{e}_2]^{\mathrm{T}} \dot{\mathbf{p}}(t_{\mathrm{TO}})\,|_k$) and the time the robot spends in the aerial phase. This flight time is varied according to the setpoint altitude as the aerial maneuver is divided into a powered ascent (PA) and an unpowered projectile (PJ) (fig. S11A). We ensure the hopping altitude set point z_d is reached by briefly producing vertical thrust of magnitude $\sum f_i = mg$ for

$$\Delta t_{\mathrm{PA}} = \begin{cases} z_{\mathrm{d}}/\dot{z}(t_{\mathrm{TO}}) - \dot{z}(t_{\mathrm{TO}})/2g & \text{for } \dot{z}(t_{\mathrm{TO}}) < 2gz_{\mathrm{d}} \\ 0 & \text{otherwise,} \end{cases} \tag{27}$$

where

$$\dot{z}(t_{\mathrm{TO}}) = \mathrm{e}_3^{\mathrm{T}} \dot{\mathbf{p}}(t_{\mathrm{TO}}) \tag{28}$$

is the vertical liftoff speed and z_{d} denotes the relative hopping altitude. The former case is when the vertical liftoff speed is insufficient for the robot to passively reach the set point, and the latter is when the jump height without any thrust assistance is already more than z_{d}. During the powered flight phase, the propelling thrust counterbalances the weight and the vehicle retains its upward speed of $\dot{z}(t_{\mathrm{TO}})$. The rest of the trajectory is a projectile with duration of

$$\Delta t_{\mathrm{PJ}} = \dot{z}(t_{\mathrm{TO}})/g + \sqrt{2z_{\mathrm{d}}/g} \tag{29}$$

in which the first term refers to the rest of the time the robot spends ascending and the latter is free fall. Finally, the landing position for cycle $k + 1$ is iteratively updated as

$$\mathbf{p}(t_{\mathrm{LD}})\,|_{k+1} = \mathbf{p}(t_{\mathrm{LD}})\,|_k + [\mathbf{e}_1 \;\; \mathbf{e}_2]^{\mathrm{T}} \dot{\mathbf{p}}(t_{\mathrm{TO}})\,|_k (\Delta t_{\mathrm{PA}} + \Delta t_{\mathrm{PJ}}) \tag{30}$$

The outcome of Eqs. 24 to 30 verifies that the touchdown location of the next cycle ($k + 1$) is predominantly determined by the landing attitude $\mathbf{z}_b(t_{\mathrm{LD}})$ of the current cycle.

Landing attitude computation and aerial trajectory planning

From the estimated current landing position $\mathbf{p}(t_{\mathrm{LD}})|_k$ provided by Eq. 24, the controller uses Eq. 30 to iteratively search for the touchdown attitude that minimizes the predicted position error of the next landing,

$$\mathbf{z}_b(t_{\mathrm{LD}})|_k = \mathrm{argmin}_{\mathbf{z}_b(t_{\mathrm{LD}})|_k} \|\mathbf{p}_{\mathrm{d}} - \mathbf{p}(t_{\mathrm{LD}})|_{k+1}\| \tag{31}$$

under the no-slip condition as constrained by the ground friction coefficient μ (see Supplementary Methods)

$$\arccos(\mathbf{e}_3 \cdot \mathbf{z}_b(t_{\mathrm{LD}}))\,|_k, \arccos(\mathbf{e}_3 \cdot \mathbf{z}_b(t_{\mathrm{TO}}))\,|_k < \tan^{-1}\mu \tag{32}$$

Depending on the set point $\mathbf{p}_{\mathrm{d}}$ and the current state of the robot, the error in Eq. 31 can be marginalized to zero or a particular value, which indicates whether the robot is able to reach the desired location in a single or multiple steps.

Thrust and attitude manager

After $\mathbf{z}_b(t_{\mathrm{LD}})$ is determined from Eq. 31, it is realized when the robot is falling by the low-level controller (v). To regulate the jumping altitude, after the liftoff, the robot is quickly reoriented to an upright direction and the collective thrust magnitude is maintained at $\sum f_i = mg$ for Δt_{PA} as described by Eq. 27. The action injects energy into the system to compensate for any losses. Thereafter, the robot remains upright until the landing location is again predicted by fig. S11B (i) for the next jump cycle.

Planning strategy for trajectory tracking

Unlike flying robots, which are able to marginalize any control errors continuously, the cyclic nature of the hopping motion only allows the corrective measure to be executed once per cycle. The action to alter the landing attitude when the robot is falling only affects the landing location in the next cycle. Therefore, to track a predefined trajectory, the desired landing location $\mathbf{p}_d$ in Eq. 31 is chosen to be the desired location of the robot at $t_{\mathrm{LD}}|_{k+1}$, under the assumption that

$$t_{\mathrm{LD}}|_{k+1} - t = \Delta t_{\mathrm{LD}} + 2\sqrt{2z_{\mathrm{d}}/g} \tag{33}$$

This consideration eliminates the inherent latency, allowing the robot to track a time-varying trajectory more precisely. Still, the complexity of achievable trajectories is intrinsically limited by the hopping frequency.

Low-level control for flight, hopping, and hybrid maneuvers

The same set of equations was applied to determine the rotor commands for flight and when the attitude of the robot during the aerial hopping phase was regulated. To achieve the designated attitude, we first implemented a proportional-integral-derivative controller (*45*) to compute the required body-centric torque $\boldsymbol{\tau}_{\mathrm{p,\,d}}$.

Depending on the operational demand, the target thrust magnitude T_{d} can be zero, mg, or a particular value. To evaluate the force allocation between four rotors, we define $\mathbf{f} = [f_1, f_2, f_3, f_4]^{\mathrm{T}}$ and a four-by-four configuration matrix $\mathbf{A}$ for power distribution

$$[\,\boldsymbol{\tau}_{\mathrm{p}}\ \ T\,]^{\mathrm{T}} = \mathbf{A}\mathbf{f} \tag{34}$$

To ensure that $f_i \geq 0$ even when $T_{\mathrm{d}} = 0$, the following attitude-priority allocation [similar to (*47*)] is used to compute **f**.

$$\mathbf{f} = \underset{\mathbf{f}}{\operatorname{argmin}}\ |\,T_{\mathrm{d}} - T\,| \tag{35}$$

$$\text{subject to } f_i \geq 0 \text{ and } \boldsymbol{\tau}_{\mathrm{p,d}} = \boldsymbol{\tau}_{\mathrm{p}}$$

Despite that the collective thrust generation may not be correctly set to T_{d}, this ensures that the torque is correctly generated ($\boldsymbol{\tau}_{\mathrm{p,\,d}} = \boldsymbol{\tau}_{\mathrm{p}}$) and minimizes the error of thrust ($T - T_{\mathrm{d}}$) when the desired thrust and torque may cause saturation of motor commands, especially for the unpowered projectile where T_{d} is set to zero.

Experimental setup and measurements

The experiments were carried out in a 3 m–by–3 m–by–2.5 m arena with 10 motion capture cameras (OptiTrack Prime 13w). The pose measurements were used for both control and ground truth measurements. The high-level controllers, including the hopping and flight controllers, were implemented by Python on the ground computer and executed at 100 Hz. The robot and the ground station communicated through a radio module (Bitcraze Crazyradio PA) via the Crazyflie Python library. We used the built-in flight controller for low-level attitude control with the customized thrust allocation method (detailed below). To manipulate the aerodynamic stabilizer, we programmed the Crazyflie controller board to drive the servo motor.

Drop tests for parameter identification

Because the stance phase dynamics captured by Eqs. 12 and 13 are subject to a few unknown lumped physical parameters, namely, $\Omega = \sqrt{k/m}$, f_{c}/m, and l_{p} ($l_0 = 22$ cm is a design parameter), we devised a drop test to identify these model parameters according to the axial dynamics of the robot (Eq. 12) before applying the equations for predicting and controlling the hopping motion.

The setup of the drop tests is schematically presented in fig. S1A. We fabricated a robotic gripper to hold the robot via a cable and release it when triggered. The end of the cable was tied through the CoM of the vehicle to ensure that the robot was in an upright orientation. A high-speed camera (MotionBLITZ EoSens mini 2) captured the trajectory of the robot at 1000 frames per second, covering the landing site. Both the gripper and the camera were controlled by the same computer. Using the Kanade-Lucas-Tomasi tracker (*86*), we identified and traced feature points on the robot between frames. Two-dimensional trajectories were reconstructed by calculating the geometric center of the tracked feature points. Figure S1B shows a snapshot of the Hopcopter in stance with the tracked features.

The tests were carried out at three drop heights: 0.68, 0.82, and 0.96 m, with four drops at each height. The high-speed footage and visually tracked trajectories are shown in movie S9, with 12 reconstructed trajectories plotted in fig. S1C. Among these tests, the robot spent ≈32 ms in the stance phase. The elastic element was stretched to 4.68 to 5.55 cm from a preloaded length of 1.97 cm. We calculated the leg contraction $l_0 - l(t)$ from the data to plot the contraction against the acceleration in fig. S1D. The outcome suggests two vertically offset linear trend lines with the same gradient. The lower and upper lines correspond to the leg compression and retraction, respectively. The evident and uniform gap between downstroke and upstroke (dark and light) accelerations reflects the energy loss or hysteresis in the form of damping. This leads to the proposed elastic force model in Eq. 7. We applied the least squares method to estimate the model parameters. The best fitted linear model (solid lines in fig. S1D) produces $k/m = 5.00 \times 10^3$ s^{-2}, $f_{\mathrm{h}}/m = 12.7$ m/s^2, and $l_{\mathrm{p}} = 1.97$ cm. The identified parameters, listed in table S2, corroborate that the weight of the robot is negligible compared with the force of the prestretched elastomer as $kl_{\mathrm{p}}/mg \approx 10$.

Computation of the landing and takeoff states

In the controlled hopping test, the recorded position and attitude data were processed to obtain $\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|$, θ_{LD}, $\Delta\psi$, and θ_{TO}. We first identified the landing and takeoff time stamps from the CoM location. The transition velocities [$\dot{\mathbf{p}}(t_{\mathrm{LD}})$, $\dot{\mathbf{p}}(t_{\mathrm{TO}})$] and attitude states [$\mathbf{z}_b(t_{\mathrm{LD}})$, $\mathbf{z}_b(t_{\mathrm{TO}})$] with respect to the inertial frame are deduced. Because, in practice, these four vectors may not be perfectly coplanar as modeled in Fig. 2B, $\mathbf{e}_\psi$ is computed as a unit normal vector of the best-fitted plane

$$\mathbf{e}_\psi = \underset{\mathbf{n}}{\operatorname{argmin}} \frac{1}{2}\Big((\mathbf{n}^{\mathrm{T}}\dot{\mathbf{p}}(t_{\mathrm{LD}})/\|\dot{\mathbf{p}}(t_{\mathrm{LD}})\|)^2 + (\mathbf{n}^{\mathrm{T}}\dot{\mathbf{p}}(t_{\mathrm{TO}})/\|\dot{\mathbf{p}}(t_{\mathrm{TO}})\|)^2 + (\mathbf{n}^{\mathrm{T}}\mathbf{z}_b(t_{\mathrm{LD}}))^2 + (\mathbf{n}^{\mathrm{T}}\mathbf{z}_b(t_{\mathrm{TO}}))^2\Big)^{\frac{1}{2}} \tag{36}$$

The average value of the objective function in Eq. 36 was 0.8°, indicating minimal misalignment. After the reprojection onto the rotational plane normal to $\mathbf{e}_\psi$, the angles θ_{LD}, $\Delta\psi$, and θ_{TO} for each jump were obtained. From the data of 130 jumps in 90 s, a wide region of the landing state [$1.5 < \|\dot{\mathbf{p}}(t_{\mathrm{LD}})\| < 3.3$ m/s and $0^\circ < \theta_{\mathrm{LD}} < 20^\circ$] was covered.

Poincaré maps for hopping stability analysis

The constructed Poincaré maps assume a constant jumping height z_{d}. Hence, the vertical landing speed at cycle k is determined by

$$\dot{z}(t_{\mathrm{LD}}) = \sqrt{2gz_{\mathrm{d}}} \tag{37}$$

Given the landing attitude $\phi|_k$ and the angle of the landing velocity $\theta_z|_k$, we calculate the landing angle $\theta_{\mathrm{LD}}|_k = |(\phi|_k - \theta_z|_k)|$ under the assumption that vectors $\mathbf{z}_w$, $\mathbf{z}_b(t_{\mathrm{LD}})|_k$, and $\dot{\mathbf{p}}(t_{\mathrm{LD}})\,|_k$ are coplanar and orthogonal to $\mathbf{e}_\psi$. The takeoff states, $\mathbf{z}_b(t_{\mathrm{TO}})_k|$ and $\dot{\mathbf{p}}(t_{\mathrm{TO}})\,|_k$, are obtained from Eqs. 16 and 21. After takeoff, the robot briefly throttles up in an upright orientation to compensate the energy lost before entering the ballistic trajectory. The horizontal velocity remains constant during flight. This implies that the horizontal speed at takeoff at cycle k is equal to the horizontal speed at landing at cycle $k + 1$. The vertical landing speeds at cycle $k + 1$ and k are the same as dictated by the jump height z_{d}. The angle of the landing velocity $\theta_z|_{k+1}$ is then the direction of the landing velocity relative to the vertical. This allows us to compute $\theta_z|_{k+1}$ or $\theta_z|_{k+1}/\theta_z|_k$ for any $\theta_z|_k$ and $\phi|_k$, which results in the Poincaré maps illustrating the stability of θ_z over jumping cycles (Fig. 6C and fig. S8).

Supplementary Materials

This PDF file includes:

Supplementary Methods
Figs. S1 to S13
Tables S1 and S2

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S9

REFERENCES AND NOTES

1. D. D. Chin, D. Lentink, How birds direct impulse to minimize the energetic cost of foraging flight. *Sci. Adv.* **3**, e1603041 (2017).
2. S. Andersson, Sexual selection and cues for female choice in leks of Jackson's widowbird *Euplectes jacksoni*. *Behav. Ecol. Sociobiol.* **25**, 403–410 (1989).
3. H. Bennet-Clark, E. Lucey, The jump of the flea: a study of the energetics and a model of the mechanism. *J. Exp. Biol.* **47**, 59–76 (1967).
4. M. Burrows, Jumping performance of froghopper insects. *J. Exp. Biol.* **209**, 4607–4621 (2006).
5. P. J. Simmons, F. C. Rind, R. D. Santer, Escapes with and without preparation: The neuroethology of visual startle in locusts. *J. Insect Physiol.* **56**, 876–883 (2010).
6. F. Zhao, W. Wang, J. Wyrwa, J. Zhang, W. du, P. Zhong, Design and demonstration of a flying-squirrel-inspired jumping robot with two modes. *Appl. Sci.* **11**, 3362 (2021).
7. A. S. H. Lau, Z. Ma, X. Huang, The aerodynamic effects of forelimb pose on the gliding flight of Dracolizards. *Phys. Fluids* **35**, 035117 (2023).
8. N. W. Bartlett, M. T. Tolley, J. T. B. Overvelde, J. C. Weaver, B. Mosadegh, K. Bertoldi, G. M. Whitesides, R. J. Wood, A 3D-printed, functionally graded soft robot powered by combustion. *Science* **349**, 161–165 (2015).
9. M. Loepfe, C. M. Schumacher, U. B. Lustenberger, W. J. Stark, An untethered, jumping roly-poly soft robot driven by combustion. *Soft Robot.* **2**, 33–41 (2015).
10. Y. Chen, H. Wang, E. F. Helbling, N. T. Jafferis, R. Zufferey, A. Ong, K. Ma, N. Gravish, P. Chirarattananon, M. Kovac, R. J. Wood, A biologically inspired, flapping-wing, hybrid aerial-aquatic microrobot. *Sci. Robot.* **2**, eaao5619 (2017).
11. R. Zufferey, A. O. Ancel, A. Farinha, R. Siddall, S. F. Armanini, M. Nasr, R. V. Brahmal, G. Kennedy, M. Kovac, Consecutive aquatic jump-gliding with water-reactive *fuel. Robot.* **4**, eaax7330 (2019).
12. U. Scarfogliero, C. Stefanini, P. Dario, Design and development of the long-jumping "grillo" mini robot, in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation* (IEEE, 2007), pp. 467–472.
13. R. Armour, K. Paskins, A. Bowyer, J. Vincent, W. Megill, Jumping robots: A biomimetic solution to locomotion across rough terrain. *Bioinspir. Biomim.* **2**, S65–S82 (2007).
14. M. Kovac, M. Fuchs, A. Guignard, J.-C. Zufferey, D. Floreano, A miniature 7g jumping robot, in *2008 IEEE International Conference on Robotics and Automation* (IEEE, 2008), pp. 373–378.
15. M. Kovač, M. Schlegel, J.-C. Zufferey, D. Floreano, A miniature jumping robot with self-recovery capabilities, in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2009), pp. 583–588.
16. M. Kovač, O. Fauria, J.-C. Zufferey, D. Floreano, The EPFL jumpglider: A hybrid jumping and gliding robot with rigid or folding wings, in *2011 IEEE International Conference on Robotics and Biomimetics* (IEEE, 2011), pp. 1503–1508.
17. M. A. Woodward, M. Sitti, Design of a miniature integrated multi-modal jumping and gliding robot, in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2011), pp. 556–561.
18. M. A. Woodward, M. Sitti, Multimo-bat: A biologically inspired integrated jumping–gliding robot. *Intl J. Robot. Res.* **33**, 1511–1529 (2014).
19. N. T. Truong, H. V. Phan, H. C. Park, Design and demonstration of a bio-inspired flapping-wing-assisted jumping robot. *Bioinspir. Biomim.* **14**, 036010 (2019).
20. E. W. Hawkes, C. Xiao, R. A. Peloquin, C. Keeley, M. R. Begley, M. T. Pope, G. Niemeyer, Engineered jumpers overcome biological limits via work multiplication. *Nature* **604**, 657–661 (2022).
21. S.-H. Chae, S.-M. Baek, J. Lee, K.-J. Cho, Agile and energy-efficient jumping–crawling robot through rapid transition of locomotion and enhanced jumping height adjustment. *IEEE/ASME Trans. Mechatron.* **27**, 5890–5901 (2022).
22. W. D. Shin, W. Stewart, M. A. Estrada, A. J. Ijspeert, D. Floreano, Elastic-actuation mechanism for repetitive hopping based on power modulation and cyclic trajectory generation. *IEEE Trans. Robot.* **39**, 558–571 (2023).
23. V. Zaitsev, O. Gvirsman, U. Ben Hanan, A. Weiss, A. Ayali, G. Kosa, A locust-inspired miniature jumping robot. *Bioinspir. Biomim.* **10**, 066012 (2015).
24. J. Zhao, J. Xu, B. Gao, N. Xi, F. J. Cintron, M. W. Mutka, L. Xiao, MSU jumper: A single-motor-actuated miniature steerable jumping robot. *IEEE Trans. Robot.* **29**, 602–614 (2013).
25. M. Kovač, M. Schlegel, J.-C. Zufferey, D. Floreano, Steerable miniature jumping robot. *Auton Robots* **28**, 295–306 (2010).
26. G.-P. Jung, C. S. Casarez, S.-P. Jung, R. S. Fearing, K.-J. Cho, An integrated jumping-crawling robot using height-adjustable jumping module, in *2016 IEEE International Conference on Robotics and Automation* (ICRA) (IEEE, 2016), pp. 4680–4685.
27. M. Noh, S.-W. Kim, S. An, J.-S. Koh, K.-J. Cho, Flea-inspired catapult mechanism for miniature jumping robots. *IEEE Trans. Robot.* **28**, 1007–1018 (2012).
28. J. Burdick, P. Fiorini, Minimalist jumping robots for celestial exploration. *Intl. J. Robot. Res.* **22**, 653–674 (2003).
29. A. L. Desbiens, M. T. Pope, D. L. Christensen, E. W. Hawkes, M. R. Cutkosky, Design principles for efficient, repeated jumpgliding. *Bioinspir. Biomim.* **9**, 025009 (2014).
30. F. Li, W. Liu, X. Fu, G. Bonsignori, U. Scarfogliero, C. Stefanini, P. Dario, Jumping like an insect: Design and dynamic optimization of a jumping mini robot based on bio-mimetic inspiration. *Mechatron.* **22**, 167–176 (2012).
31. G. Kenneally, A. De, D. E. Koditschek, Design principles for a family of direct-drive legged robots. *IEEE Robot. Autom. Lett.* **1**, 900–907 (2016).
32. D. W. Haldane, M. M. Plecnik, J. K. Yim, R. S. Fearing, Robotic vertical jumping agility via series-elastic power modulation. *Sci. Robot.* **1**, eaag2048 (2016).
33. D. W. Haldane, J. K. Yim, R. S. Fearing, Repetitive extreme-acceleration (14-g) spatial jumping with Salto-1P, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 3345–3351.
34. J. K. Yim, B. R. P. Singh, E. K. Wang, R. Featherstone, R. S. Fearing, Precision robotic leaping and landing using stance-phase balance. *IEEE Robot. Autom. Lett.* **5**, 3422–3429 (2020).
35. A. De, D. E. Koditschek, Parallel composition of templates for tail-energized planar hopping, in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2015), pp. 4562–4569.
36. J. K. Yim, E. K. Wang, R. S. Fearing, Drift-free roll and pitch estimation for high-acceleration hopping, in *2019 International Conference on Robotics and Automation (ICRA)* (IEEE, 2019), pp. 8986–8992.
37. V. Klemm, A. Morra, C. Salzmann, F. Tschopp, K. Bodie, L. Gulich, N. Küng, D. Mannhart, C. Pfister, M. Vierneisel, F. Weber, R. Deuber, R. Siegwart, Ascento: A two-wheeled jumping robot, in *2019 International Conference on Robotics and Automation (ICRA)* (IEEE, 2019), pp. 7515–7521.
38. A. M. Johnson, D. E. Koditschek, Toward a vocabulary of legged leaping, in *2013 IEEE International Conference on Robotics and Automation* (IEEE, 2013), pp. 2568–2575.
39. A. L. Brill, A. De, A. M. Johnson, D. E. Koditschek, Tail-assisted rigid and compliant legged leaping, in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2015), pp. 6304–6311.
40. G. C. Haynes, J. Pusey, R. Knopf, A. M. Johnson, D. E. Koditschek, Laboratory on legs: an architecture for adjustable morphology with legged robots, in *Proceedings of Unmanned Systems Technology XIV* (SPIE, 2012), vol. 8387, pp. 349–362.
41. A. Vidyasagar, J.-C. Zufferey, D. Floreano, M. Kovač, Performance analysis of jump-gliding locomotion for miniature robotics. *Bioinspir. Biomim.* **10**, 025006 (2015).
42. D. Wei, T. Gao, Z. Li, X. Mo, S. Zheng, C. Zhou, Hybrid inspired research on the flying-jumping locomotion of locusts using robot counterpart. *Front. Neurorobot.* **13**, 87 (2019).
43. S.-M. Baek, S. Yim, S.-H. Chae, D.-Y. Lee, K.-J. Cho, Ladybird beetle–inspired compliant origami. *Sci. Robot.* **5**, eaaz6262 (2020).
44. B. Zhu, J. Xu, A. Charway, 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.
45. D. Mellinger, N. Michael, V. Kumar, Trajectory generation and control for precise aggressive maneuvers with quadrotors. *Intl. J. Robot. Res.* **31**, 664–674 (2012).
46. M. Faessler, A. Franchi, D. Scaramuzza, Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories. *IEEE Robot. Autom. Lett.* **3**, 620 (2018).
47. S. Sun, A. Romero, P. Foehn, E. Kaufmann, D. Scaramuzza, A comparative study of nonlinear mpc and differential-flatness-based control for quadrotor agile flight. *IEEE Trans. Robot.* **38**, 3357–3373 (2022).
48. A. Wu, H. Geyer, The 3-D spring–mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments. *IEEE Trans. Robot.* **29**, 1114–1124 (2013).
49. U. Saranlı, Ö. Arslan, M. M. Ankaralı, Ö. Morgül, Approximate analytic solutions to non-symmetric stance trajectories of the passive spring-loaded inverted pendulum with damping. *Nonlinear Dyn.* **62**, 729–742 (2010).
50. M. M. Ankaralı, U. Saranlı, Control of underactuated planar pronking through an embedded spring-mass hopper template. *Auton Robots* **30**, 217–231 (2011).
51. A. Sayyad, B. Seth, P. Seshu, Single-legged hopping robotics research–a review. *Robotica* **25**, 587–613 (2007).
52. J. E. Seipel, P. Holmes, Running in three dimensions: Analysis of a point-mass sprung-leg model. *Intl. J. Robot. Res.* **24**, 657–674 (2005).
53. Z. Batts, J. Kim, K. Yamane, Untethered one-legged hopping in 3D using linear elastic actuator in parallel (LEAP), in *2016 International Symposium on Experimental Robotics*, D. Kulic, Y. Nakamura, O. Khatib, G. Venture, Eds., vol. 1 of *Springer Proceedings in Advanced Robotics* (Springer, 2017), pp. 103–112.
54. M. H. Raibert, H. B. Brown Jr., M. Chepponis, Experiments in balance with a 3D one-legged hopping machine. *Int. J. Robot. Res.* **3**, 75–92 (1984).
55. P. Terry, G. Piovan, K. Byl, Towards precise control of hoppers: Using high order partial feedback linearization to control the hopping robot FRANK, in *2016 IEEE 55th Conference on Decision and Control (CDC)* (IEEE, 2016), pp. 6669–6675.
56. P. Foehn, A. Romero, D. Scaramuzza, Time-optimal planning for quadrotor waypoint flight. *Sci. Robot.* **6**, eabh1221 (2021).

57. Z. E. Teoh, S. B. Fuller, P. Chirarattananon, N. O. Préz-Arancibia, J. D. Greenberg, R. J. Wood, A hovering flapping-wing microrobot with altitude control and passive upright stability, in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2012), pp. 3209–3216.
58. D. W. Haldane, K. C. Peterson, F. L. G. Bermudez, R. S. Fearing, Animal-inspired design and aerodynamic stabilization of a hexapedal millirobot, in *2013 IEEE international conference on robotics and automation* (IEEE, 2013), pp. 3279–3286.
59. M. J. Anderson, J. G. Sullivan, J L. Talley, K. M. Brink, S. B. Fuller, T. L. Daniel, The "Smellicopter," a bio-hybrid odor localizing nano air vehicle, in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2019), pp. 6077–6082.
60. K. Dong, R. Ding, S. Bai, X. Cai, P. Chirarattananon, Stabilizing aerodynamic dampers for cooperative transport of a suspended payload with aerial robots. *Adv. Int. Syst.* **5**, 2300112 (2023).
61. J. Norby, J. Y. Li, C. Selby, A. Patel, A. M. Johnson, Enabling dynamic behaviors with aerodynamic drag in lightweight tails. *IEEE Trans. Robot.* **37**, 1144–1153 (2021).
62. S. Bai, R. Ding, P. Chirarattananon, A micro aircraft with passive variable-sweep wings. *IEEE Robot. Autom. Lett.* **7**, 4016–4023 (2022).
63. R. Cory, R. Tedrake, Experiments in fixed-wing UAV perching, in *AIAA Guidance, Navigation and Control Conference and Exhibit* (AIAA, 2008), p. 1930.
64. W. Hoburg, R. Tedrake, System identification of post stall aerodynamics for UAV perching, in *AIAA Infotech@ Aerospace Conference* (AIAA, 2009), p. 1930.
65. F. J. Boria, R. J. Bachmann, P. G. Ifju, R. D. Quinn, R. Vaidyanathan, C. Perry, J. Wagener, A sensor platform capable of aerial and terrestrial locomotion, in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2005), pp. 3959–3964.
66. A. Kalantari, M. Spenko, Modeling and performance assessment of the HyTAQ, a hybrid terrestrial/aerial quadrotor. *IEEE Trans. Robot.* **30**, 1278–1285 (2014).
67. J. R. Page, P. E. Pounds, The quadroller: Modeling of a UAV/UGV hybrid quadrotor, in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2014), pp. 4834–4841.
68. H. Jia, S. Bai, R. Ding, J. Shu, Y. Deng, B. L. Khoo, P. Chirarattananon, A quadrotor with a passively reconfigurable airframe for hybrid terrestrial locomotion. *IEEE/ASME Trans. Mechatron.* **27**, 4741–4751 (2022).
69. R. Zhang, Y. Wu, L. Zhang, C. Xu, F. Gao, Autonomous and adaptive navigation for terrestrial-aerial bimodal vehicles. *IEEE Robot. Autom. Lett.* **7**, 3008–3015 (2022).
70. A. Fabris, E. Aucone, S. Mintchev, Crash 2 squash: An autonomous drone for the traversal of narrow passageways. *Adv. Int. Syst.* **4**, 2200113 (2022).
71. H. Jia, R. Ding, K. Dong, S. Bai, P. Chirarattananon, Quadrolltor: A reconfigurable quadrotor with controlled rolling and turning. *IEEE Robot. Autom. Lett.* **8**, 4052–4059 (2023).
72. K. Kim, P. Spieler, E.-S. Lupu, A. Ramezani, S.-J. Chung, A bipedal walking robot that can fly, slackline, and skateboard. *Sci. Robot.* **6**, eabf8136 (2021).
73. C. Hong, D. Tang, Q. Quan, Z. Cao, Z. Deng, A combined series-elastic actuator & parallel-elastic leg no-latch bio-inspired jumping robot. *Mech. Mach. Theory* **149**, 103814 (2020).
74. L. Daler, J. Lecoeur, P. B. Hählen, D. Floreano, A flying robot with adaptive morphology for multi-modal locomotion, in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2013), pp. 1361–1366.
75. L. Daler, S. Mintchev, C. Stefanini, D. Floreano, A bioinspired multi-modal flying and walking robot. *Bioinspir. Biomim.* **10**, 016005 (2015).
76. T. Anzai, Y. Kojio, T. Makabe, K. Okada, M. Inaba, Design and development of a flying humanoid robot platform with bi-copter flight unit, in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)* (IEEE, 2021), pp. 69–75.
77. Y.-H. Hsiao, S. Bai, Y. Zhou, H. Jia, R. Ding, Y. Chen, Z. Wang, P. Chirarattananon, Energy efficient perching and takeoff of a miniature rotorcraft. *Commun. Eng.* **2**, 38 (2023).
78. Y. H. Hsiao, P. Chirarattananon, Ceiling effects for hybrid aerial–surface locomotion of small rotorcraft. *IEEE/ASME Trans. Mechatron.* **24**, 2316–2327 (2019).
79. H. W. Wopereis, T. Van Der Molen, T. Post, S. Stramigioli, M. Fumagalli, Mechanism for perching on smooth surfaces using aerial impacts, in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)* (IEEE, 2016), pp. 154–159.
80. P. Chirarattananon, K. Y. Ma, R. J. Wood, Perching with a robotic insect using adaptive tracking control and iterative learning control. *Intl. J. Robot. Res.* **35**, 1185–1206 (2016).
81. M. T. Pope, C. W. Kimes, H. Jiang, E. W. Hawkes, M. A. Estrada, C. F. Kerst, W. R. T. Roderick, A. K. Han, D. L. Christensen, M. R. Cutkosky, A multimodal robot for perching and climbing on vertical outdoor surfaces. *IEEE Trans. Robot.* **33**, 38 (2017).
82. A. Lussier Desbiens, A. T. Asbeck, M. R. Cutkosky, Landing, perching and taking off from vertical surfaces. *Intl J. Robot. Res.* **30**, 355–370 (2011).
83. N. Pan, J. Jiang, R. Zhang, C. Xu, F. Gao, Skywalker: A compact and agile air-ground omnidirectional vehicle. *IEEE Robot. Autom. Lett.* **8**, 2534–2541 (2023).
84. N. B. David, D. Zarrouk, Design and analysis of FCSTAR, a hybrid flying and climbing sprawl tuned robot. *IEEE Robot. Autom. Lett.* **6**, 6188–6195 (2021).
85. S. Bai, P. Chirarattananon, SplitFlyer Air: A modular quadcopter that disassembles into two bicopters mid-air. *IEEE/ASME Trans. Mechatron.* **27**, 4729–4740 (2022).
86. J. Shi, Tomasi, Good features to track, in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 1994), pp. 593–600.

Acknowledgments: We thank K. Dong, T. Wang, and J. Xu for helpful discussions and B. Pu. and S. Yu for assistance with the experiment. **Funding:** This work described in this paper was substantially supported by grants from the Shenzhen-Hong Kong-Macau Science & Technology Project (Category C) (contract number SGDX20220530111401009) and the Research Grants Council of the Hong Kong Special Administrative Region, China (grant number CityU 11217923). **Author contributions:** S.B. and P.C. conceived the study. S.B. designed, fabricated, and tested the prototypes. S.B. and R.D. developed the dynamic models and control strategy of the robot. S.B., R.D., Q.P., H.J., and P.C. analyzed and interpreted the results. S.B., Z.Y., and P.C. wrote and revised the manuscripts. P.C. provided funding and supervised the research. **Competing interests:** P.C., S.B., R.D. have submitted a US Patent application (no. 18/465,403) and a Chinese patent application (no. 202311781462.7) related to this work. The other authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the article are present in the paper or the Supplementary Materials. Data and corresponding codes for figure compiling can be found at https://doi.org/10.5281/zenodo.10777420.

Submitted 25 May 2023
Accepted 14 March 2024
Published 10 April 2024
10.1126/scirobotics.adi8912

Science Robotics

An agile monopedal hopping quadcopter with synergistic hybrid locomotion

Songnan Bai, Qiqi Pan, Runze Ding, Huaiyuan Jia, Zhengbao Yang, and Pakpong Chirarattananon

Sci. Robot. **9** (89), eadi8912. DOI: 10.1126/scirobotics.adi8912

Editor's summary

Conventional jumping robots are actuated in a stance phase, either by a rapid energy release mechanism or with legs driven by actuators. Previous attempts at a jumping and flying robot used these mechanisms for jump-assisted takeoff but struggled to achieve continuous hopping or adjustable jump heights. A new robot called the Hopcopter by Bai *et al.* combines a commercial quadcopter with a passive telescopic leg. The Hopcopter was actuated in the aerial phase, allowing for continuous hopping, high jump frequencies, and tunable jump height. Intermittent hops led to sudden increases in acceleration and enable highly agile movements such as rapid, tight turns. The robot was also modified with an aerodynamic stabilizer that interacts with air flow to stabilize the robot with only internal sensors. The combination of locomotion modes was shown to increase endurance and power efficiency, which could potentially extend travel distance and operational range for applications in surveillance and inspection. —Melisa Yashinski

---

Science Robotics (ISSN 2470-9476) is published by the American Association for the Advancement of Science. 1200 New York Avenue NW, Washington, DC 20005. The title *Science Robotics* is a registered trademark of AAAS.