

Bio-inspired Inverted Landing Strategy in a Small Aerial Robot Using Policy Gradient

Pan Liu¹, Junyi Geng², Yixian Li¹, Yanran Cao¹, Yagiz E. Bayiz¹, Jack W. Langelaan², *Member, IEEE*,
Bo Cheng¹, *Member, IEEE*

Abstract—Landing upside down on a ceiling is challenging as it requires a flier to invert its body and land against the gravity, a process that demands a stringent spatiotemporal coordination of body translational and rotational motion. Although such an aerobatic feat is routinely performed by biological fliers such as flies, it is not yet achieved in aerial robots using onboard sensors. This work describes the development of a bio-inspired inverted landing strategy using computationally efficient Relative Retinal Expansion Velocity (RREV) as a visual cue. This landing strategy consists of a sequence of two motions, i.e. an upward acceleration and a rapid angular maneuver. A policy search algorithm is applied to optimize the landing strategy and improve its robustness by learning the transition timing between the two motions and the magnitude of the target body angular velocity. Simulation results show that the aerial robot is able to achieve robust inverted landing, and it tends to exploit its maximal maneuverability. In addition to the computational aspects of the landing strategy, the robustness of landing is also significantly dependent on the mechanical design of the landing gear, the upward velocity at the start of body rotation, and timing of rotor shutdown.

I. INTRODUCTION

Landing upside down is a challenging aerobatic feat that is routinely performed by bats [1] and many insects species such as flies [2], however, it is rarely achieved in small aerial robots using their onboard resources. Such capability is essential for small aerial robots as it not only expands the repertoire of aerobatic maneuvers, but also enables the robots to perch robustly on surfaces of various inclinations, thereby to maintain a desired observation or resting position for long-duration inspection, surveillance, search and rescue [3][4][5], which is particularly critical for small robots as they suffer more in the flight time comparing to their larger counterparts [6]. Previous work has only achieved arguably less difficult landing types on horizontal, vertical or inclined surfaces with the help of external motion tracking [3] or with specialized grasp mechanisms [5][7]. Inverted landing however, is more challenging because it requires either the ability to hover inverted or (in the case of vehicles that cannot change blade pitch or rotation direction or the rotors) a dynamic maneuver that spatiotemporally coordinates body translation and rotation to achieve safe contact Fig. 1 [2].

A recent study by the authors [2] reveals that, with relatively limited computation at small body size [8], flies

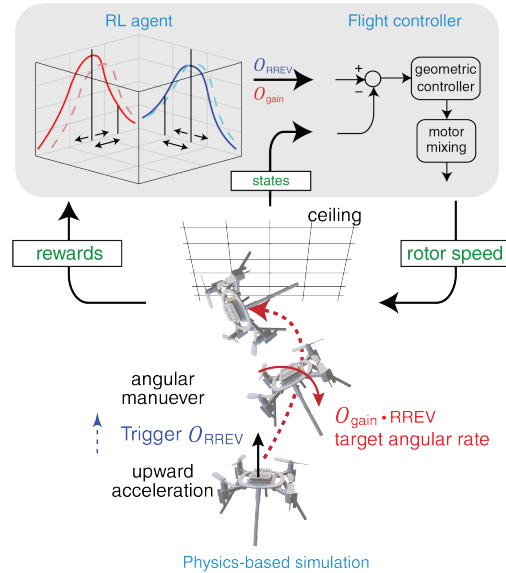


Fig. 1. **Overview of the inverted landing problem.** This study consists of three parts: the reinforcement learning agent which learns the high level parameters for the bio-inspired landing strategy; the low level controller which controls the motion of the quadrotor according to the high level parameters and the states it received; and the simulated landing environment and quadrotor.

are able to land upside down on a ceiling by executing a well coordinated sequence of behavior modules. The inverted landing starts with upward acceleration and followed by rapid body rotation and leg extension, and ends with leg-assisted body swing with feet firmly planted in the substrate, see Fig. 2(a). Remarkly, this process represents a synergistic combination of both computational and mechanical intelligence [9]. Computationally, the flies' landing strategies rely on extracting some key visual cues that can be efficiently and rapidly processed and mapped to control action by their sensorimotor system. For example, the Relative Retinal Expansion Velocity (RREV) [10][11][12] of the approaching ceiling in the fly's retina encodes the fly's remaining time to collide with the ceiling, see Fig. 2(b). Mathematically, it is the inverse of the time to collision, i.e. RREV equals to approaching speed divided by distance. It is one of the key visual cues that determine the onset timing and magnitude of the rapid rotation. Mechanically, the adhesion from their pulvilli which ensures a firm grip and the viscoelasticity of their compliant legs which damps out contact impact improve the robustness of inverted landing. This exemplifies the importance of relying on both computational and mechanical

¹ Biological and Robotic Intelligent Fluid Locomotion Lab, Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA 16802, USA. Corresponding to B.C. buc10@psu.edu

² Air Vehicle Intelligence and Autonomy Lab, Department of Aerospace Engineering, The Pennsylvania State University, University Park, PA 16802, USA

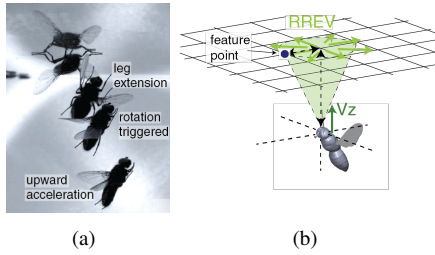


Fig. 2. **An example of the inverted landing of a fly and the visual cue RREV.** (a) Flies land upside down on the ceiling by executing a sequence of well coordinated behavioral modules. (b) The Relative Retina Expansion Velocity (RREV) is due to the looming stimuli when flies approach the ceiling with upward velocity v_z .

intelligence for robust landing, and inspires the design of landing strategy in robotic flier in the current work.

To adapt and optimize the landing strategy learned from flies to a robot, we choose to use model-free Reinforcement Learning (RL) in this work. With the ability to explore and understand the environment without or with only a crude model [13][14], RL has demonstrated its potential in achieving optimal policies for complex robotic problems, such as helicopter [15], where previously a rather accurate model is necessary for controller design [16][17]. Among the recent developed RL algorithms, the symmetrical sampling parameter-exploring policy gradient (SyS-PEPG) algorithm has been proven to be particularly effective in robot training compare to traditional policy gradient algorithms [18]. It largely reduces the noise in gradient estimation by directly and symmetrically sampling the parameter space, rather than sampling from the probabilistic policy as normally a policy search algorithm does.

In this paper we investigate how to achieve robust inverted landing strategy in a small aerial robot relying only on its heavily limited onboard resources. We present an application of the SyS-PEPG algorithm on a simulated nano quadrotor with the goal of achieving inverted landing on a ceiling. This work consists of three parts, see Fig. 1: (1) a RL agent which learns high level meta parameters; (2) the low level flight controller which generates desired rotor speed according to the high level parameters and quadrotor states it received; and (3) a simulated quadrotor with landing environment. Inspired by the inverted landing strategies exemplified by flies, the landing process is designed to use a sequence of two motion primitives controlled by specific optical flow: i.e. an upward acceleration and a rapid body rotation. The quadrotor is assumed to be able to extract the visual cue RREV (which can be calculated directly from the optical flow using the optical flow sensor available to the quadcopter). The SyS-PEPG algorithm learns the distribution of two high level meta parameters, i.e. the threshold for RREV which determines the timing of the transition of the two motion primitives, and the gain which determines the magnitude of angular velocity. We show that the quadrotor learned robust inverted landing strategy after around 100 rollouts. The learned inverted landing patterns are a subset of what a flying animal is

capable of doing, as the maneuverability of a quadrotor is lower than that of a flying animal.

This work has two major contributions. First, the bio-inspired inverted landing strategy achieved in this work provides insights on designing aggressive maneuvers in aerial robots. It also highlights the importance of combining computational and mechanical intelligence to improve robustness. Second, this paper demonstrated that RREV can play an important role for motion planning while being highly computational efficient.

The rest of the paper is organized as follows. Sec. II gives a brief introduction to the inverted landing strategy of flies. Sec. III describes the high level learning problem which aims to find the optimal meta parameters in the landing strategy. These parameters are inputs of the low level flight controllers, which is described in Sec. IV. Results and discussion are then presented in Sec. V. Finally, Sec. VI concludes the study and provides directions for the future work.

II. BIO-INSPIRED LANDING STRATEGY

The recent study by the authors [2] shows that, rather than previously considered as stereotyped, flies demonstrated highly variable inverted landing behaviors. In particular, the observed landing behaviors composed of an upward acceleration followed by a rapid body rotation with variable magnitude and axes of rotation. The large variance in the body rotation is shown statistically to be mediated by the visual cues the flies perceived prior to the start of the rotation, while the visual control is likely absent during the course of the body rotation. As a result, when flying with higher forward or lower upward velocities, flies decrease the pitch rate but increase the degree of leg-assisted swing, thereby leveraging the transfer of body linear momentum.

The inverted landing strategy extracted from the flies [2] underlines a sequence of two motions: an upward acceleration and a rapid body rotation. More importantly, the transition timing of the two motions is determined by a threshold of RREV, and furthermore, the desired magnitude of the rotational maneuvers is linearly related to the RREV, and two other visual cues that encode the relative fore/aft and lateral ceiling rotation. Note that, as inspired by the flies' landing strategy, the visual feedback is only used to trigger and determine the target degree of the angular maneuver, while it is not used during the angular maneuver *per se*, which is visually open-loop.

This landing strategy is simplified in two ways to apply to the aerial robot. First, as the quadrotor is symmetric in longitudinal and lateral, the rotation motion in the landing is restricted to only pitch. Second, the forward motion is assumed to be small in this work so that the fore/aft visual cue is negligible. With the simplified landing strategy (or policy) applied, policy parameters learned specifically for the robot are the threshold and proportional gain, i.e. $O = [O_{RREV}, O_{gain}]^T$, see Fig. 3. The desired pitch rate after rotation is triggered is designed as $q_d = O_{gain} \cdot RREV$.

III. LEARNING PROBLEM

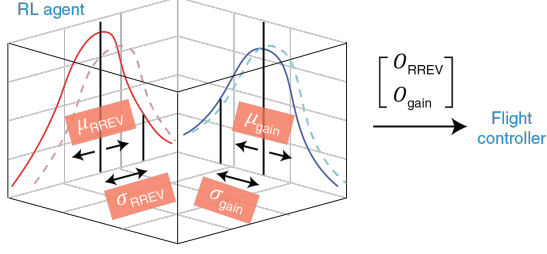


Fig. 3. **The reinforcement learning agent.** The agent learns the distribution of two meta parameters, which are the transition timing O_{RREV} between the two motions executed in the landing strategy, and the gain O_{gain} which determines the magnitude of the rotation velocity, respectively.

As mentioned in the introduction, the SyS-PEPG [18] reduces variance in estimating policy gradient by directly sampling from the parameter space θ . Unlike traditional policy search algorithms where actions are sampled from the policy, here we define the policy to be over the distribution of $O = [O_{RREV}, O_{gain}]^T$, i.e. $\pi_\theta = p(O|\theta)$. The sampling occurs at the start of each rollout and the drawn sample O is used throughout this rollout. In this way, the controller is deterministic for each rollout. Here, we assume the distribution of the sample O is a 2D normal distribution where the two dimensions are independent, i.e. $O \sim \mathcal{N}(\mu, \Sigma)$, see Fig. 3, where $\mu = [\mu_{RREV}, \mu_{gain}]^T$, and $\Sigma = \begin{bmatrix} \sigma_{RREV} & 0 \\ 0 & \sigma_{gain} \end{bmatrix}$. Therefore, the parameters to be optimized by the RL algorithm is $\theta = [\mu_{RREV}, \sigma_{RREV}, \mu_{gain}, \sigma_{gain}]^T$, where $\mu \cdot$ and $\sigma \cdot$ are the means and standard deviations for the distribution $O = [O_{RREV}, O_{gain}]^T$, respectively. In addition, the SyS-PEPG symmetrically sample from the two sides of the means, i.e. if $O^+ = [\mu_{RREV}, \mu_{gain}]^T + \epsilon$ is sampled, the $O^- = [\mu_{RREV}, \mu_{gain}]^T - \epsilon$ will also be included in the rollouts, where $\epsilon \in \mathbb{R}^2$ is a perturbation. This symmetric sampling further improves the gradient approximation by removing the usage of the possibly poorly estimated baseline in updating $\mu \cdot$.

At each rollout, the quadrotor starts at the initial state $\mathbf{x}_0 = [0, 0, h_0]^T$, $\mathbf{R}_0 = [\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3]$, $\mathbf{v}_0 = [0, 0, 0]^T$, $\mathbf{\Omega}_0 = [0, 0, 0]^T$, where $\mathbf{x}_0$, $\mathbf{R}_0$, $\mathbf{v}_0$ and $\mathbf{\Omega}$ are initial position, orientation, velocity and angular rates, respectively. And h_0 is the initial height of the quadrotor, and $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ are the coordinate axes defined in Fig. 4(a). It then accelerates upwards until the RREV exceeds O_{RREV} (the upward velocity is randomly selected in each rollout). At this moment, the quadrotor switch from the upward motion to the pitch up maneuver, whose desired pitch velocity is $q_d = O_{gain} \cdot \text{RREV}$. Note that the RREV at the actual start of the body rotation could be slightly different from the O_{RREV} in practice due to the delay in the system. After the switch, the quadrotor tries to reach the q_d while having its position and velocity uncontrolled. The rollout ends after a timeout T_{to} after the switch.

The successful inverted landing is featured by two conditions: the height of the quadrotor is equal to the height of

the ceiling h_c substrates the initial height of the quadrotor h_0 , i.e. $h = h_c - h_0$; and the $\mathbf{b}_3$ -axis points vertically down, i.e. $\mathbf{b}_3 = -\mathbf{e}_3$, see Fig. 4(a). Therefore, the reward at each time step of the landing process is defined as

$$r_t = \frac{h}{h_c - h_0} + \mathbf{b}_3 \cdot (-\mathbf{e}_3) \quad (1)$$

Here the reward from the rotation, i.e. $\mathbf{b}_3 \cdot (-\mathbf{e}_3)$, is not normalized so that the under rotated rollouts are penalized and well rotated rollouts are incentivized. Suppose for a rollout the trace of the reward is $r = [r_1, r_2, \dots, r_t, \dots, r_{T-1}, r_T]$. Then the cumulative reward for a rollout is defined as

$$r_c = \sum_{t=1}^T \gamma^{T-t} r_t \quad (2)$$

where γ is the discount factor. Defining cumulative reward in this way highlights the importance of sustained r_t over a period of time, further guarantees a true successful landing. The overall goal of the learning is to optimize the parameter θ such that the expected reward for each episode, i.e. $J(\theta)$, is maximized. The θ is updated as

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta) \quad (3)$$

where α is the learning rate and $\nabla_\theta J(\theta)$ is the policy gradient. The derivation and the approximation for the $\nabla_\theta J(\theta)$ for the SyS-PEPG algorithm are presented in the original paper [18][19] in detail.

IV. FLIGHT CONTROL AND SIMULATION SETUP

To investigate the inverted landing in a small aerial robot, we select the light-weight (27 gram) and small-size (9 cm²) quadrotor Crazyflie 2.1¹ as the platform. Although the majority of this work is done in simulation under Gazebo environment² using ROS³ framework, the result of this work is applicable to real experiment as we tried to build the simulation as close to the real world as possible. For example, the parameters of the quadrotor and the rotor aerodynamics we use are from a previous system identification work [20]. In addition, a first order ODE is used to approximate the rotor dynamics, and we also added delays (~ 10 ms) and noise (uniform distribution with bounds equal to 10% of magnitude of the variables) in both sensing and control.

A. Rigid body dynamics and the dominant aerodynamics

Under the assumption that it undergoes no deformation during flight, the nano quadrotor is modeled as a 6-DOF dynamic system subjected to a combination of gravitational, aerodynamic and propulsion forces and moments. Its configuration is determined by the location of the center of mass (COM) and the attitude relative to the inertial frame, i.e. the configuration space is the special Euclidean space SE(3).

Two reference frames are defined, see Fig. 4(a), i.e. the inertial frame, $\mathcal{F}_e = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$, which is fixed to the earth with $\mathbf{e}_3$ -axis pointing vertically upward and $\mathbf{e}_1 - \mathbf{e}_2$ on

¹Bitcraze AB <https://bitcraze.io/crazyflie-2-1/>

²Open Robotics <http://www.gazebosim.org/>

³Open Robotics <https://www.ros.org/>

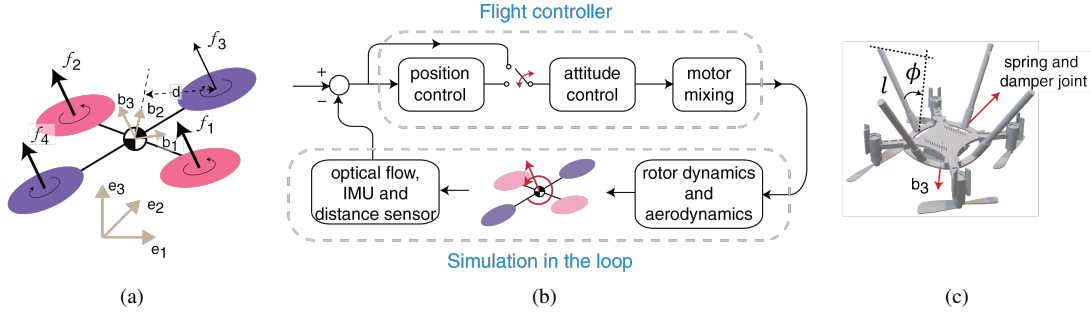


Fig. 4. **The flight controller and simulation setup.** (a) The thrust generated by the quadrotor is along the positive $\mathbf{b}_3$ -axis. (b) The flight controller is used to control the position and attitude of the quadrotor. (c) The design of the landing gear.

the horizontal plane; the body-fixed reference frame $\mathcal{F}_b = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$ with $\mathbf{b}_1$ and $\mathbf{b}_2$ on the quadrotor plane along the angle bisectors of the arms and $\mathbf{b}_3$ normal to the quadrotor plane. We use $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{v} \in \mathbb{R}^3$ to denote the position and velocity vector of the COM of the quadrotor relative to the inertial frame $\mathcal{F}_e$. To avoid singularity or ambiguity issues, we use rotation matrix $\mathbf{R} \in SO(3)$ to represent the attitude, where $SO(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} | \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det \mathbf{R} = 1\}$ is the special orthogonal group. This is beneficial for controlling aggressive maneuvers, such as inverted landing. We also let $\boldsymbol{\Omega} \in \mathbb{R}^3$ to be the angular velocity. The equation of motion is

$$\dot{\mathbf{x}} = \mathbf{v} \quad (4)$$

$$\dot{\mathbf{R}} = \mathbf{R}\hat{\boldsymbol{\Omega}} \quad (5)$$

$$m\dot{\mathbf{v}} = -mg\mathbf{e}_3 + f\mathbf{R}\mathbf{e}_3 \quad (6)$$

$$\mathbf{J}\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} + \boldsymbol{\tau} \quad (7)$$

where $\mathbf{J}$, $f \in \mathbb{R}$ and $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3]^T$ are the moment of inertia matrix, the total aerodynamic force and torque, respectively. The wedge map $\hat{\cdot} : \mathbb{R}^3 \rightarrow so(3)$ is defined by the condition $\hat{\mathbf{x}}\mathbf{y} = \mathbf{x} \times \mathbf{y}$.

A near-hover rotor model with zero inflow is used. This assumes that the effects of blade aeromechanics (e.g. flapping [16]) are very small and that advance ratio (the ratio of flight speed and blade tip speed) is small. Thus, the dominant aerodynamics is induced by the four spinning rotors. The steady-state thrust f_i and reaction torque τ_{ri} generated by a hovering rotor can be modeled as

$$f_i = c_T \omega^2 \quad (8)$$

$$\tau_{ri} = c_\tau f_i \quad (9)$$

where c_T , c_τ and ω are the motor thrust constant, motor torque constant and angular velocity of the rotor, respectively. Given the force/moment acting on the quadrotor, the motor speed can be obtained from the classic quadrotor dynamics.

B. Position and attitude controller

To perform an inverted landing, the quadrotor is likely to fly aggressively that involve large-angle maneuvers. To address this problem, we use the geometric tracking controller [21] as the low level position and attitude controller,

Fig. 4(b). This geometric tracking controller presents almost global exponential attractiveness in $SE(3)$.

The tracking error for position $\mathbf{x}$, velocity $\mathbf{v}$, orientation $\mathbf{R}$, and angular velocity $\boldsymbol{\Omega}$ are defined as follows.

$$\mathbf{e}_x = \mathbf{x} - \mathbf{x}_d \quad (10)$$

$$\mathbf{e}_v = \mathbf{v} - \mathbf{v}_d \quad (11)$$

$$\mathbf{e}_R = \frac{1}{2} (\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee \quad (12)$$

$$\mathbf{e}_\Omega = \boldsymbol{\Omega} - \mathbf{R}^T \mathbf{R}_d \boldsymbol{\Omega}_d \quad (13)$$

where the subscript d indicates the desired states. The $\vee$ map is the inverse of the wedge map.

The quadrotor is an underactuated system with 4 rotors and 6 DOF. Under this constraint, the geometric tracking controller tracks a given translational command $\mathbf{b}_1$ by translating it to a desired orientation $\mathbf{R}_d$. The control inputs f and $\boldsymbol{\tau}$ can then be written as

$$f = (-k_x \mathbf{e}_x - k_v \mathbf{e}_v + mg\mathbf{e}_3 + m\ddot{\mathbf{x}}_d) \cdot \mathbf{R}\mathbf{e}_3 \quad (14)$$

$$\boldsymbol{\tau} = -k_R \mathbf{e}_R - k_\Omega \mathbf{e}_\Omega \quad (15)$$

$$+ \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} - \mathbf{J}(\hat{\boldsymbol{\Omega}}\mathbf{R}^T \mathbf{R}_d \boldsymbol{\Omega}_d - \mathbf{R}^T \mathbf{R}_d \hat{\boldsymbol{\Omega}}_d) \quad (16)$$

where k_x , k_v , k_R and k_Ω are positive controller gains. The quadrotor is only able to generate lift in the positive direction of $\mathbf{b}_3$. This will lead to the lift pulling the quadrotor downwards when the third component of $\mathbf{b}_3$ is negative, see Fig. 4(c). To prevent this, the rotors are commanded to shut down when the third component of $\mathbf{b}_3$ becomes negative.

C. Design of landing gear

We also designed a landing gear with four legs to assist the inverted landing, Fig. 4(c). The length of the legs l and their angle ϕ relative to $\mathbf{b}_3$ are 40 mm and 45 degrees, respectively. The tip of the legs are adhesive to the ceiling and the joints at the root of the joints are flexible, i.e. with torsional spring and damper properties. The force of adhesion is assumed to be able to ensure a firm grip in the simulation once the feet or the tip of a leg makes contact to the ceiling. Landing gear stiffness and damping were manually tuned in simulation to improve performance.

V. RESULTS AND DISCUSSION

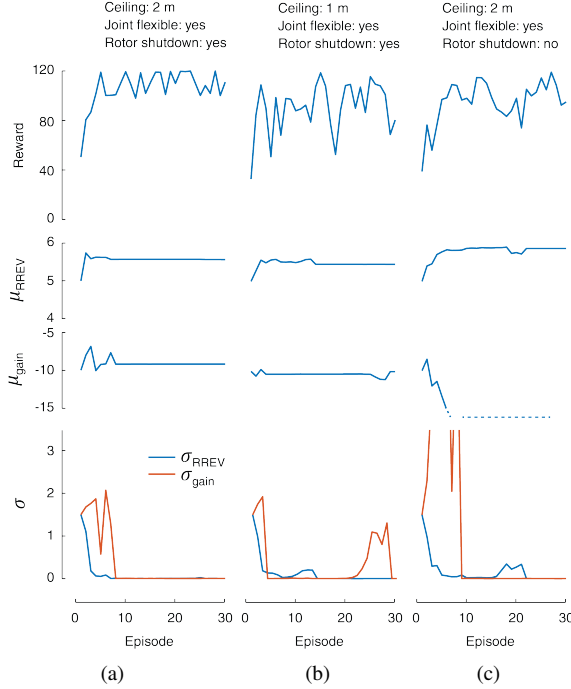


Fig. 5. **The learning results.** (a) The quadrotor learns robust inverted landing strategy after about 100 rollouts. (b) Insufficient upward accelerating or (c) disabling rotor shutdown decreases the robustness of landing.

To understand what factors affect the robustness of the inverted landing, we investigated three scenarios in simulation when carrying out the learning as shown in Fig. 5. In the first scenario (Fig. 5(a)), with the height of the ceiling equals to 2 m, the quadrotor was able to sufficiently accelerate upward and shut down rotors when the third component of $\mathbf{b}_3$ became negative, while for other two scenarios we either changed the height of the ceiling to 1 m, Fig. 5(b) or disabled the rotor shutdown, Fig. 5(c). In the first scenario, robust successful inverted landing was achieved (see next paragraph for details). The second scenario (Fig. 5(b)) resulted in insufficient upward acceleration before the rotation triggered (this was due to the limit in the lift generation capacity of the Crazyflie, therefore requiring sufficiently long traveling distance, e.g., more than 1 m, to accelerate to a sufficiently high initial upward velocity). And the third scenario (Fig. 5(c)) resulted in the lift generated by the rotors pulling the quadrotor down after the pitch angle is over 90 degrees. The results showed that these two cases resulted in lower expected cumulative reward than Fig. 5(a), and hence lower successful landing rate. In addition, it took longer time for them to converge to the optimal landing strategy. This indicated that a higher upward velocity was better for the robustness of the inverted landing, which was consistent with the ubiquitous upward acceleration behavior observed in all the successful landings in flies [2].

In the first scenario, the SyS-PEPG converged after approximately 10 episodes (10 rollouts for each episode), (Fig. 5(a)). As shown by the reward plot, 9 out of 20 episodes

after convergence have expected cumulative reward equal to approximately 120, which means all 10 rollouts in these episodes ended with 3 or 4 legs of the quadrotor attaching to the ceiling (the tip of the four legs might not be at a same plane due to the flexibility in the leg joints), which we classified as successful inverted landing. For other episodes after the convergence, the expected cumulative rewards were higher than 100, indicating that on average at least two legs of the quadrotor were attached to the ceiling. In addition, the standard deviations $[\sigma_{RREV}, \sigma_{gain}]^T$ became close to zero after the convergence, indicating that the landing strategy converged to an approximately deterministic one - with very low level of exploration in the end.

In the successful inverted landings, the quadrotor started with accelerating upward, Fig. 6(a). And after the RREV exceeded the threshold $O_{RREV} \sim \mathcal{N}(5.58, < 0.001)$ (indicated by time t_1), it switched to pitch up motion with the desired pitch velocity to be $q_d = O_{gain} \cdot RREV$, where $O_{gain} \sim \mathcal{N}(-9.17, < 0.001)$. During the pitch-up maneuver, the quadrotor continued to ascend with its remaining upward momentum while the velocity continued to decrease due to gravity. In addition, the thrust generated by propellers also pulled the quadrotor backwards (in the direction of $\mathbf{b}_3$ -axis). At the moment when the third component of $\mathbf{b}_3$ became negative (indicated by time t_2), the rotors were commanded to shut down (the process of slow down was approximated by a first order ODE). After the rotors stopped rotating, the quadrotor was only subjected to the force of gravity. The remaining momentum brought the quadrotor to an inverted orientation prior to touchdown, although most of cases not perfectly inverted. In the end, the quadrotor landed on the ceiling with either fore leg or rear leg and then the remaining momentum from upward and pitch motion brought other legs attach to the ceiling.

It is worth noting that, in different rollouts, the quadrotor could be at different upward velocity and distance towards the ceiling at the start of the rotation motion, despite with the same threshold O_{RREV} . This was because of the differences in the initial upward velocity selected in each rollout. For example, Fig. 6 showed the states for one of the successful landing rollout. This quadrotor was about 0.6 m away from the ceiling with upward velocity 3.2 m/s when it started to rotate (time t_1). And the maximum pitch velocity it reached before touchdown was about 1400 degree/s.

The quadrotor is less maneuverable than flies considering their maximal linear and rotational acceleration relative to their size. This was also reflected by the landing strategies they learned. The flies typically switch to the rotation when RREV exceeds a value between 19 and 32 rad/s, much higher than the $O_{RREV} \sim \mathcal{N}(5.58, < 0.001)$ of the quadrotor. As the inverse of RREV indicates the time to collision, this result showed that the quadrotor required about 180 ms to prepare for the inverted landing, while a fly only needed about 31 ms to 53 ms. This could also mean that the quadrotor required a much bigger spatial clearance for successful landing at the start of body rotation. Furthermore, the learned $O_{gain} \sim \mathcal{N}(-9.17, < 0.001)$ for the quadrotor

ACKNOWLEDGMENT

This research was supported by the National Science Foundation (IIS-1815519 and CMMI-1554429).

REFERENCES

- [1] A. J. Bergou, S. M. Swartz, H. Vejdani, D. K. Riskin, L. Reimnitz, G. Taubin, and K. S. Breuer, "Falling with style: bats perform complex aerial rotations by adjusting wing inertia," *PLoS Biol.*, vol. 13, no. 11, p. e1002297, 2015.
- [2] P. Liu, S. P. Sane, J.-M. Mongeau, J. Zhao, and B. Cheng, "Flies land upside down on a ceiling using rapid visually mediated rotational maneuvers," *Science advances*, vol. 5, no. 10, p. eaax1877, 2019.
- [3] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 664–674, 2012.
- [4] D. Mehanovic, J. Bass, T. Courteau, D. Rancourt, and A. L. Desbiens, "Autonomous thrust-assisted perching of a fixed-wing uav on vertical surfaces," in *Conference on Biomimetic and Biohybrid Systems*. Springer, 2017, pp. 302–314.
- [5] M. Graule, P. Chirarattananon, S. Fuller, N. Jafferis, K. Ma, M. Spenko, R. Kornbluh, and R. Wood, "Perching and takeoff of a robotic insect on overhangs using switchable electrostatic adhesion," *Science*, vol. 352, no. 6288, pp. 978–982, 2016.
- [6] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.
- [7] H. Jiang, M. T. Pope, E. W. Hawkes, D. L. Christensen, M. A. Estrada, A. Parlier, R. Tran, and M. R. Cutkosky, "Modeling the dynamics of perching with opposed-grip mechanisms," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3102–3108.
- [8] P. Liu and B. Cheng, "Limitations of rotational manoeuvrability in insects and hummingbirds: evaluating the effects of neuro-biomechanical delays and muscle mechanical power," *Journal of the Royal Society Interface*, vol. 14, no. 132, p. 20170068, 2017.
- [9] M. Kovac, "Learning from nature how to land aerial robots," *Science*, vol. 352, no. 6288, pp. 895–896, 2016.
- [10] E. Baird, N. Boeddeker, M. R. Ibbotson, and M. V. Srinivasan, "A universal strategy for visually guided landing," *Proceedings of the National Academy of Sciences*, vol. 110, no. 46, pp. 18 686–18 691, 2013.
- [11] H. Wagner, "Flow-field variables trigger landing in flies," *Nature*, vol. 297, no. 5862, pp. 147–148, 1982.
- [12] M. V. Srinivasan, S.-W. Zhang, J. S. Chahl, E. Barth, and S. Venkatesh, "How honeybees make grazing landings on flat surfaces," *Biological cybernetics*, vol. 83, no. 3, pp. 171–183, 2000.
- [13] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [14] S. Levine and V. Koltun, "Guided policy search," in *International Conference on Machine Learning*, 2013, pp. 1–9.
- [15] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *Advances in neural information processing systems*, 2007, pp. 1–8.
- [16] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics and Automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [17] Y. Song, J. F. Horn, Z. Li, and J. W. Langealaan, "Modeling, simulation, and non-linear control of a rotorcraft multi-lift system," in *American helicopter society 69th annual forum. American Helicopter Society, Phoenix, May*, 2013, pp. 21–23.
- [18] F. Sehne, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, and J. Schmidhuber, "Parameter-exploring policy gradients," *Neural Networks*, vol. 23, no. 4, pp. 551–559, 2010.
- [19] Y. E. Bayiz, L. Chen, S.-J. Hsu, P. Liu, A. N. Aguilés, and B. Cheng, "Real-time learning of efficient lift generation on a dynamically scaled flapping wing using policy search," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–5.
- [20] J. Förster, "System identification of the crazyflie 2.0 nano quadcopter," B.S. thesis, ETH Zurich, 2015.
- [21] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor uav on se (3)," in *49th IEEE conference on decision and control (CDC)*. IEEE, 2010, pp. 5420–5425.

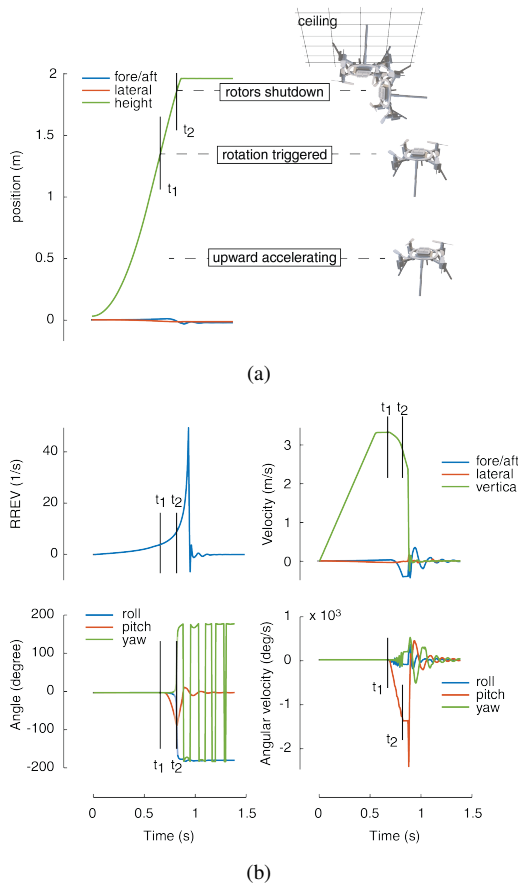


Fig. 6. **A sample of a successful inverted landing.** (a) The position and the animated landing sequence. (b) The time trajectory of RREV, linear velocity, attitude, and angular velocity of the quadrotor.

led to $q_d \approx 2900$ degree/s.

VI. CONCLUSION AND FUTURE WORK

In this study, we designed a vision-guided robust inverted landing strategy in a small aerial robot, where a policy search algorithm (SyS-PEPG) is used to adapt the parameters in the bioinspired strategy for a nano quadrotor. This landing strategy is consisted of a sequence of two motions, i.e. an upward acceleration and a rapid pitch-up rotation. The computationally efficient visual cue RREV is used as the key sensory cue in this strategy to determine the transition timing of the two motions and the magnitude of the rotation rate, i.e. $O = [O_{RREV}, O_{gain}]^T$. Once the rotation is triggered, the desired rotation rate is determined based on the RREV at the triggering moment and the vision is not involved in the later stage of the landing any longer. This is particularly advantageous as the image blur due to the fast rotational maneuver is completely avoided. All together, the strategy designed in this study achieved vision-guided robust inverted landing in the simulated environment. It is computationally low-cost for sensing and control and can be implemented in a limited-resources small aerial robot. In the future, this work will be extended to real aerial robots aiming to achieve a full range of landing behaviors.