Ubicomp CPD 2023

# SmoothLander:
# A Quadrotor Landing Control System with Smooth Trajectory Guarantee Based on Reinforcement Learning

Chenyu Zhao[1], Haoyang Wang[1], Jiaqi Li, Fanhang Man, Shilong Mu,
Wenbo Ding, Xiao-Ping Zhang, Xinlei Chen

Tsinghua University

TBSI 清华-伯克利深圳学院 Tsinghua-Berkeley Shenzhen Institute

# Landing of Quadrotors

Quadrotors need to land and take off during all kinds of tasks.

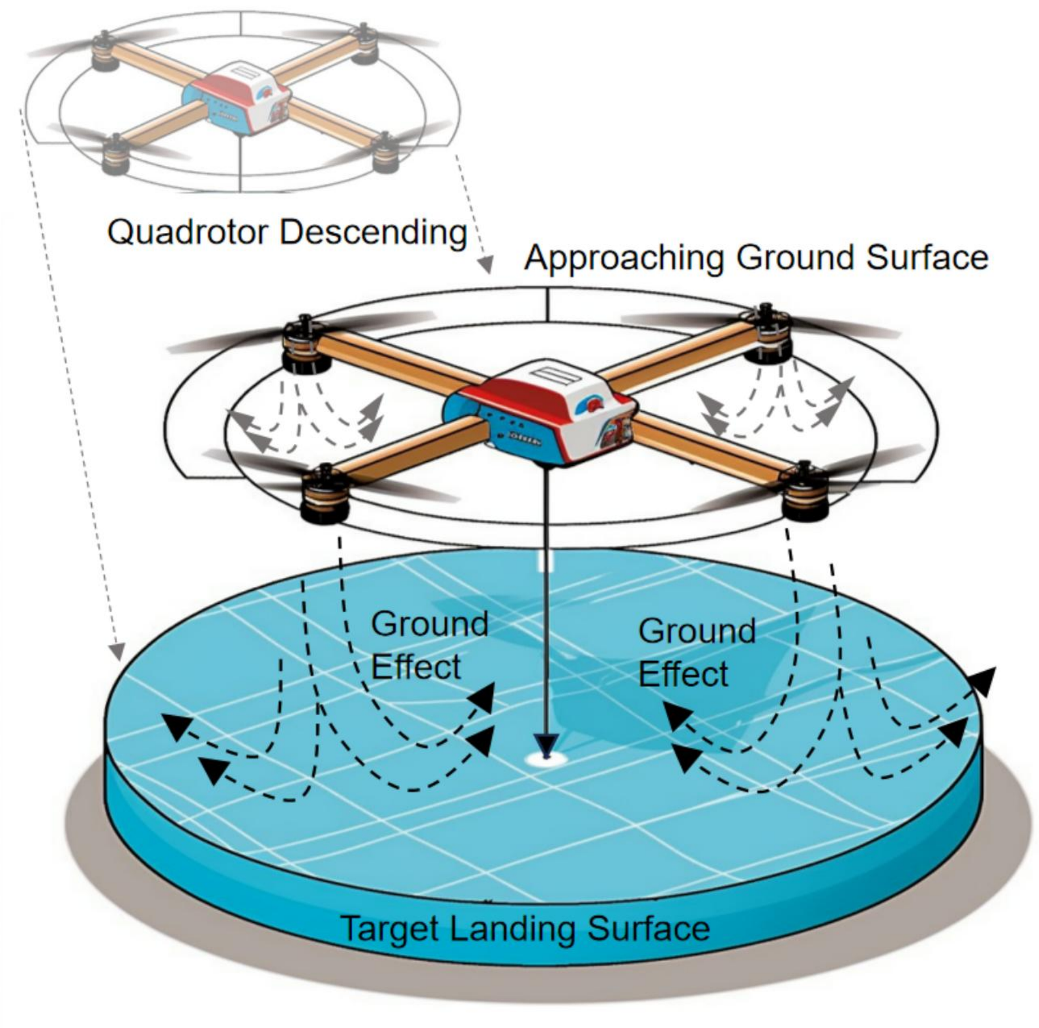

Delivery

Near crowd

Equipment of high precision

Search and Rescue

# Ground Effect(GE) during landing of Quadrotors



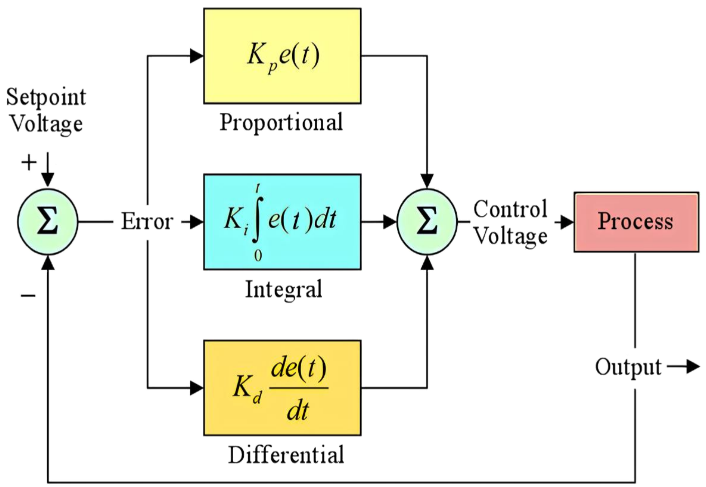

Then, this lift force may cause unstability...

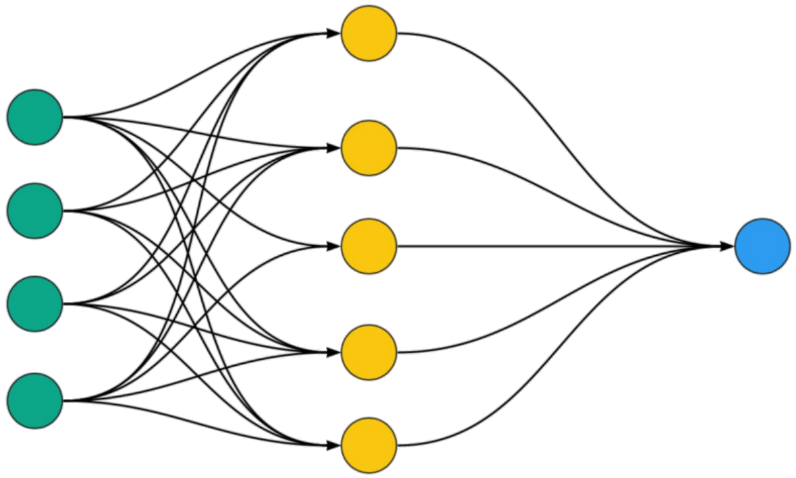
- Quadrotor collision
- Equipment damage
- ......

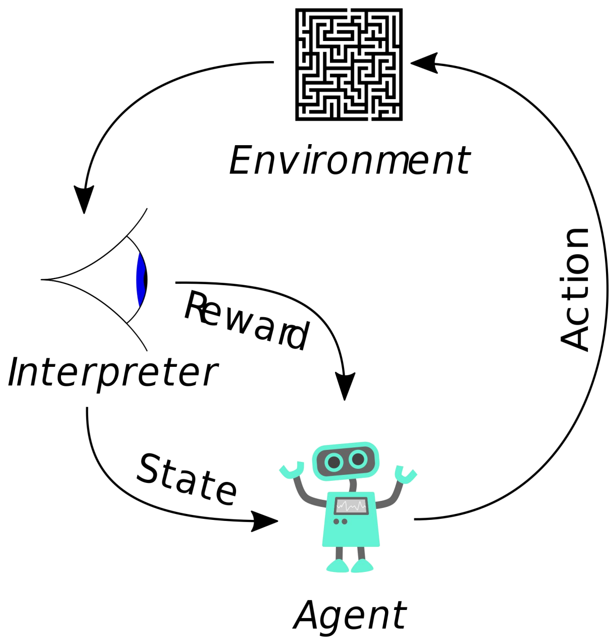
Need a controller to land smoothly

# Methods of alleviating Ground Effect

How to control the quadrotors to land smoothly and stably under the interference of the ground effect and control noise?






Self-adaptive PID

---Latency?

ANN

---Stability guarantee?

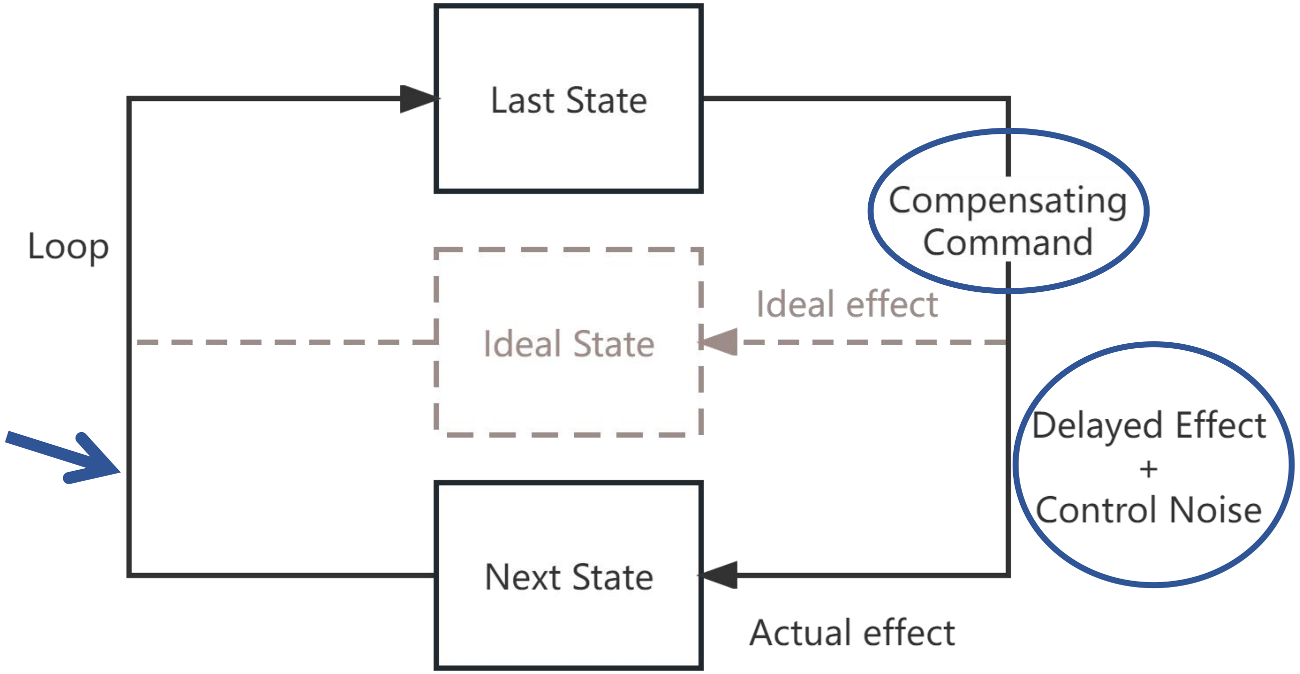
RL

---In-field influences?

# Two Challenges

C1: Delayed compensating time v.s. oscillation,
and control noise v.s.control accuracy.

Not Accurate




Oscillation Ocurrs!

Solution:

- Use pre-trained RL to account for the future state changes.

# Two Challenges

C2: The lack of training data to learn the distribution and influence of GE.

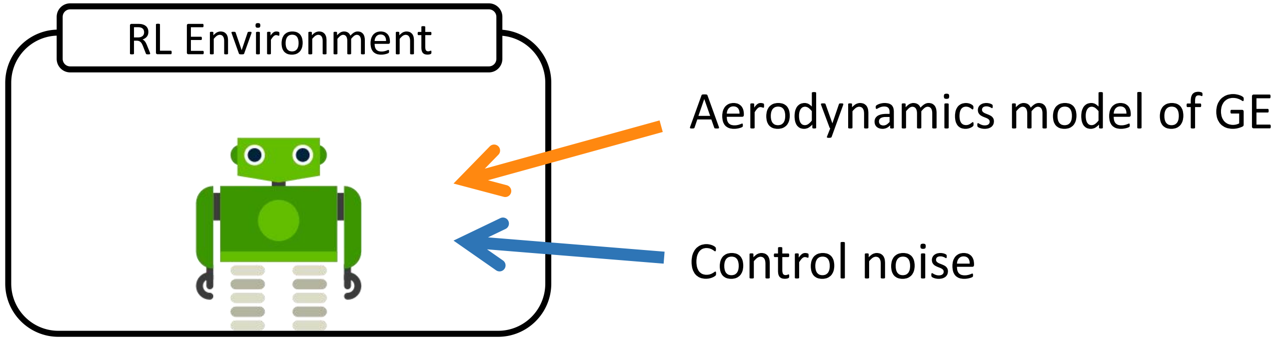
- Large state space

- High-dimensional transition matrix

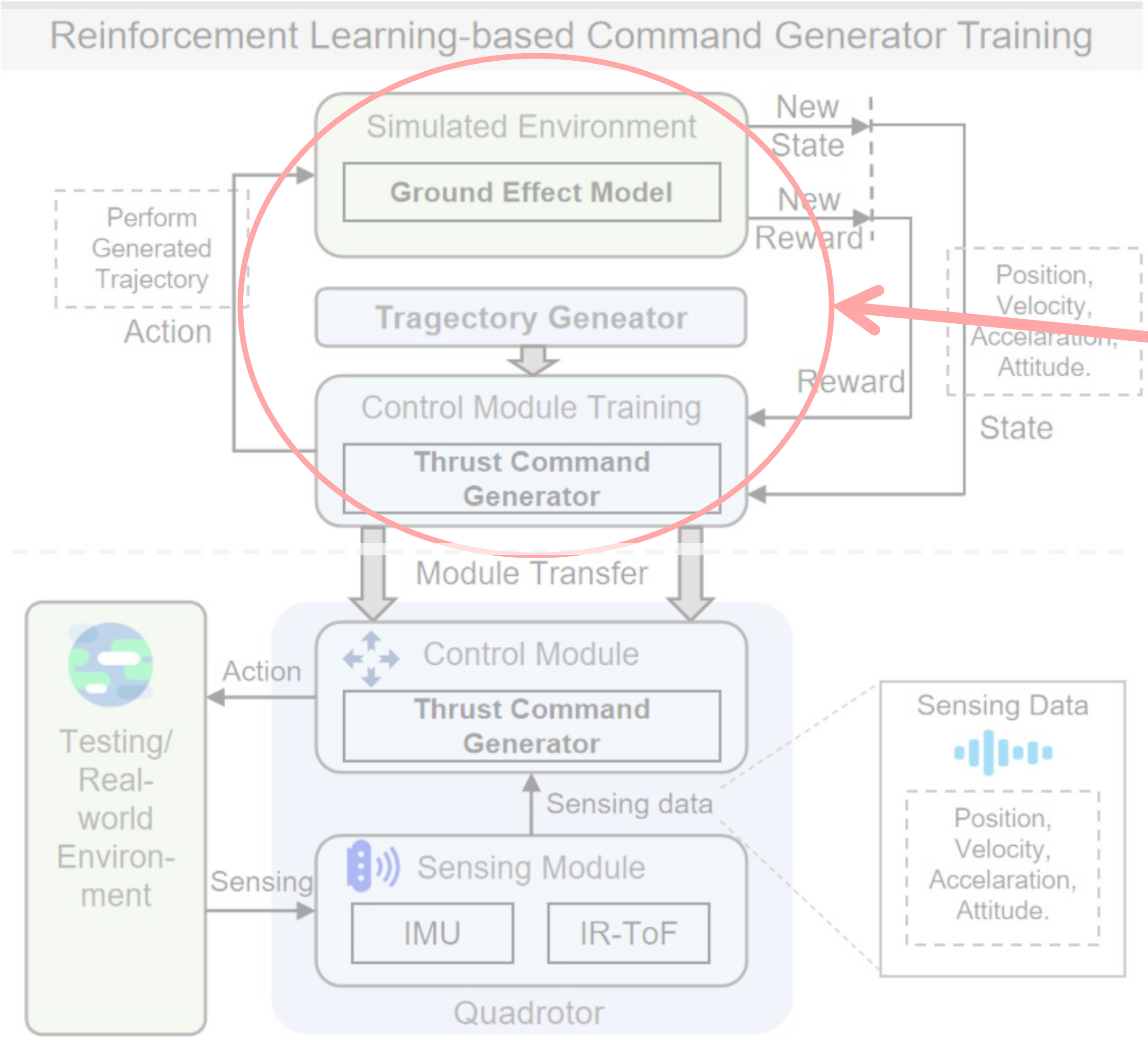
- Insufficient command-state pair data

Solution:

- Adding physical-feature based model of GE into training environment.

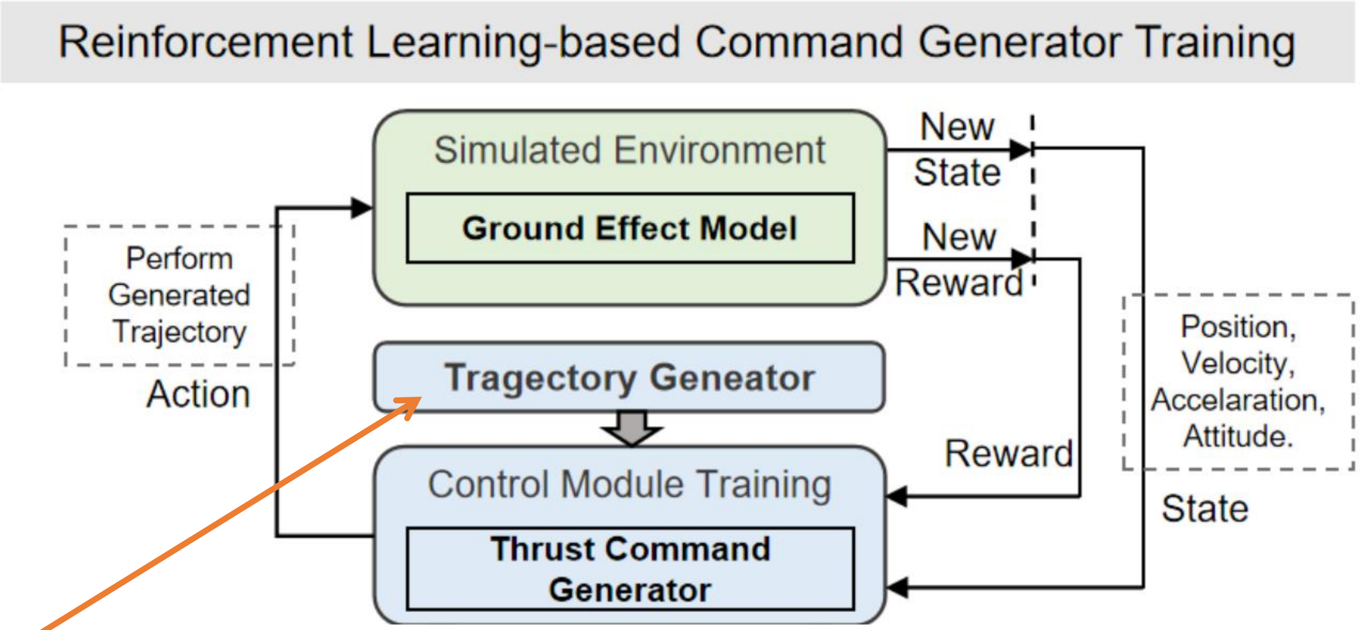


# System Design




Three key components of the system

# System Design



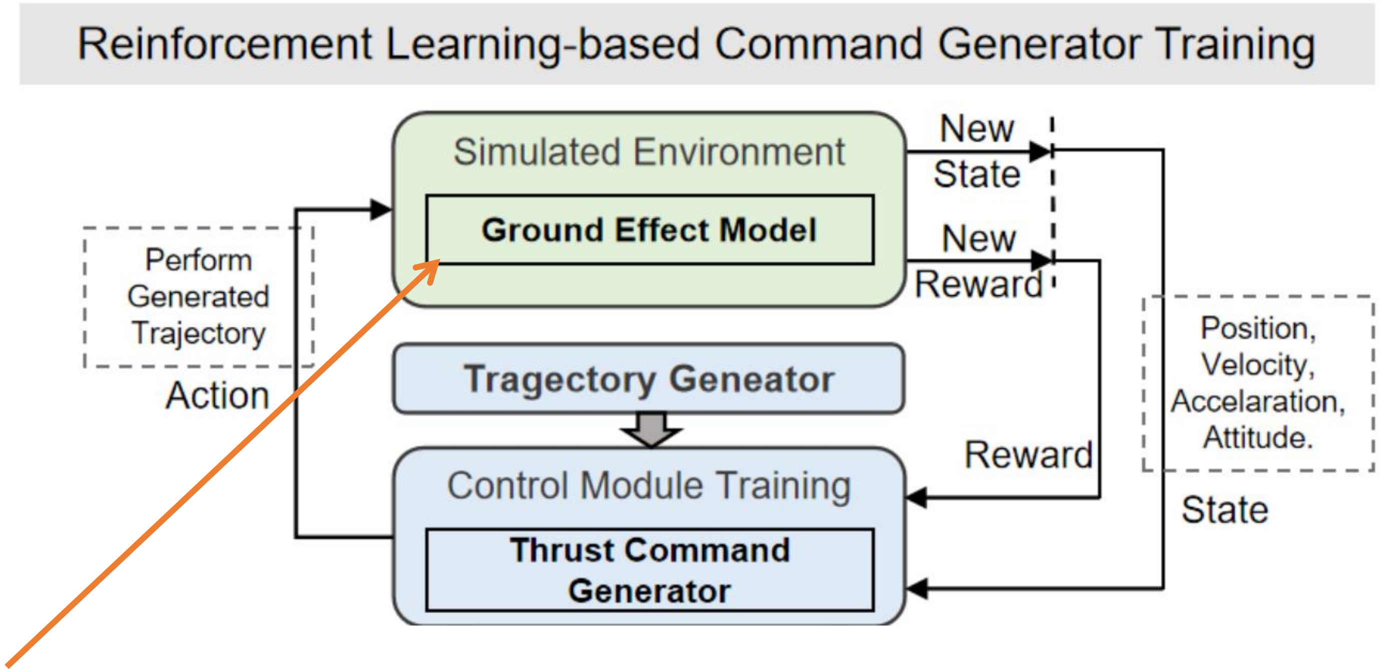

## Component 1: Trajectory Generator

From critical damping state

Designed Trajectory: $\mathbf{p}_d(t) = e^{(-Ct)}(1 + Ct)(\mathbf{p}_{init} - \mathbf{p}_{end}) + \mathbf{p}_{end},\ t \in \mathbb{R}^+$

- Continuity in first two orders' derivatives
- Slow to approach final ground

# System Design




## Component 2: Ground Effect Modeling

Ground Effect Model:

$$T(n,p_z) = k_T n^2 / [\rho D^4 [1 - (D/4p_z)^2 - D^2 p_z / \sqrt{(d^2+4p_z^2)^3} - (D^2/2)(p_z/\sqrt{(2d^2+4p_z^2)^3}\backslash) - 2D^2(p_z/\sqrt{(b^2+4p_z^2)^3})I_b]]$$ [1]

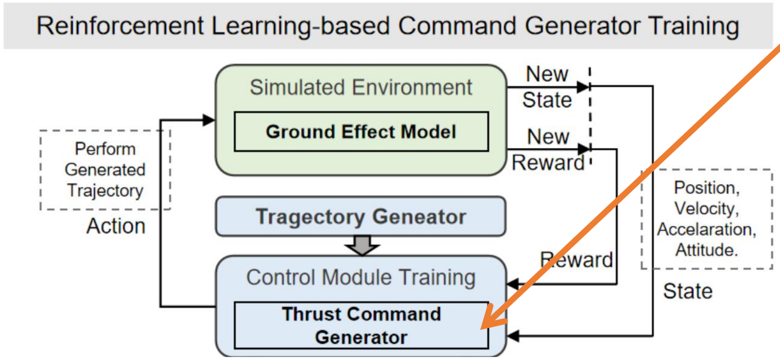
, which is validated experimentally on a real-world testbench



[1] Pedro Sanchez-Cuevas, Guillermo Heredia, and Anibal Ollero. 2017. Characterization of the aerodynamic ground effect and its influence in multirotor control. International Journal of Aerospace Engineering 2017 (2017).

# System Design

## Component 3: Thrust Command Generator




p: Position
v: Velocity
a: Acceleration
$\omega$: Angular velocity

Reward: $1/\mathrm{sum}(\mathrm{abs}(p - p_d))$
Action: Actuation command
Strategy: Commands - Certain states

---

**Algorithm 1** Learning a policy for the control of quadrotors based on reinforcement learning algorithm

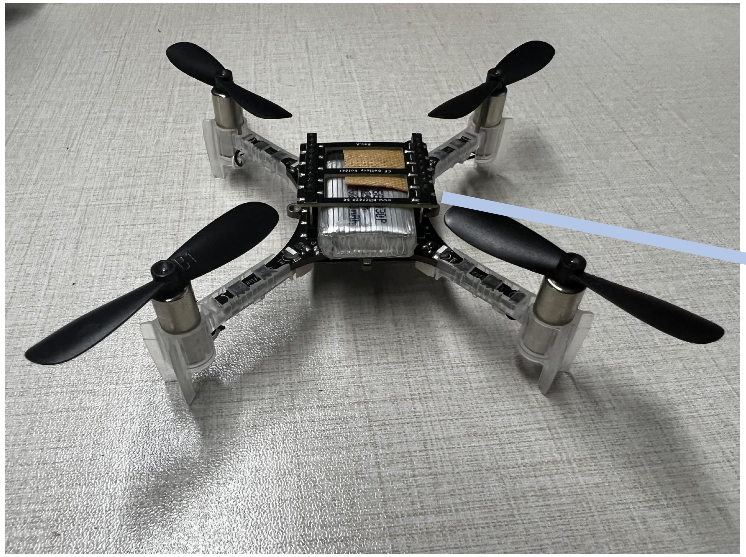
---

Randomly initialize a model $\pi$
**for** epoch=1:$M$ **do**
    randomly initialize $\mathbf{p}_t, \mathbf{v}_t, \mathbf{a}_t, \boldsymbol{\omega}_t$;
    **for** $t = 0 : T - 1$ **do**
        Quadrotor executes an action $\mathbf{u}_t = \pi(\mathbf{p}_t)$;
        Calculate $\mathbf{f_u}, \mathbf{f_w}, \boldsymbol{\tau}_\mathbf{u}, \boldsymbol{\tau}_\mathbf{w}$;
        Calculate $\mathbf{a}_{t+1}, \mathbf{v}_{t+1}$;% *according to Eq(1)*
        Calculate $\mathbf{p}_{t+1}$;% *according to Eq(1)*
        Reward $\mathbf{r}_{t+1} = R(\mathbf{u}_t, \mathbf{p}_{t+1})$
        Update the model $\boldsymbol{\pi}$ and the state $\mathbf{p}_{t+1}$
    **end for**
**end for**

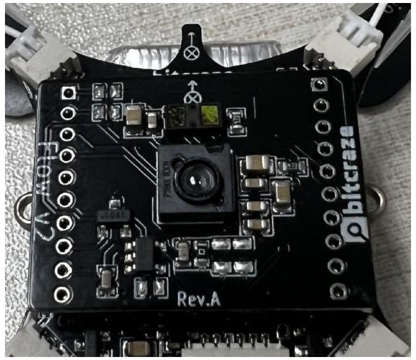
---

# Evaluation Setup

Environment: in both simulation and in-field experiment, only in z-axis, based on Crazyflie 2.1, with configuration and data collected from the real world.



Crazyflie, only weights 33 grams



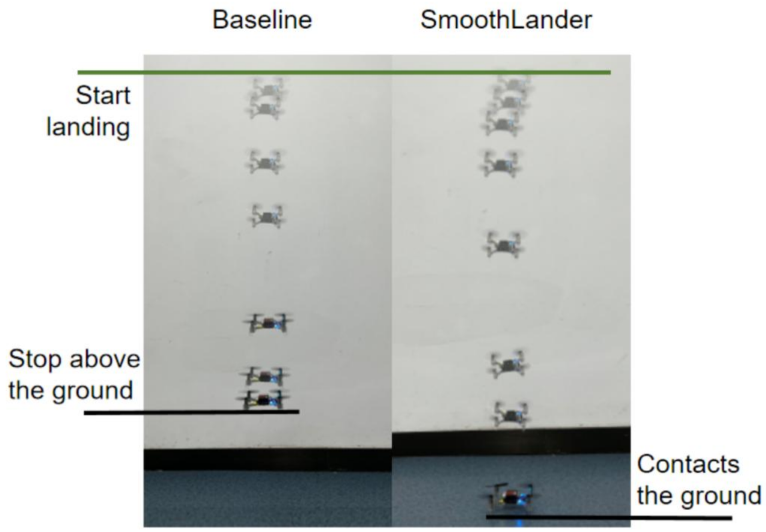

Mounted with a Flow Deck v2.

Baseline: A non-linear tracking controller does not consider outside wind.

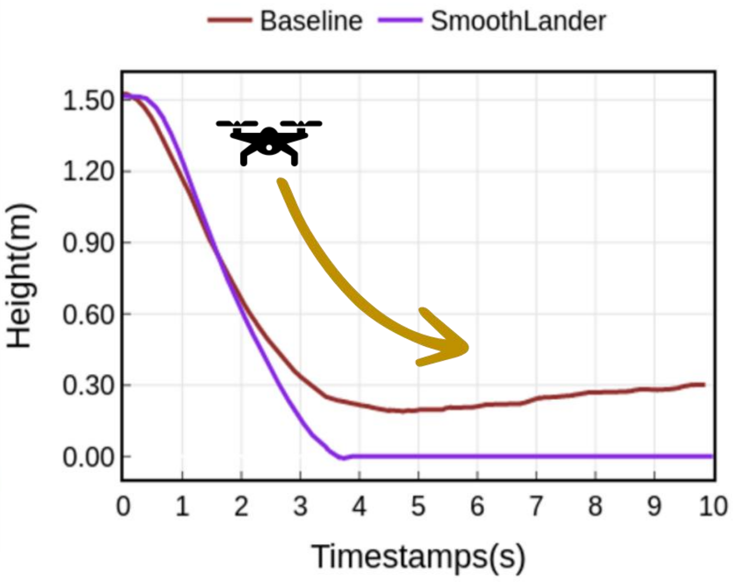
Metrics: Error between actual trajectory and designed trajectory.

# Performance

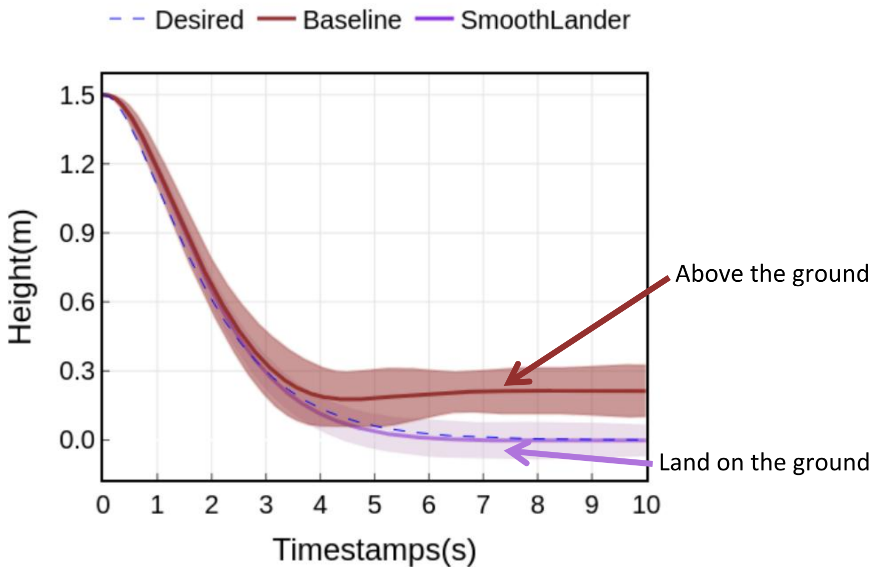
## Resisting the upward lift from GE:




Clips of landing crazyflie



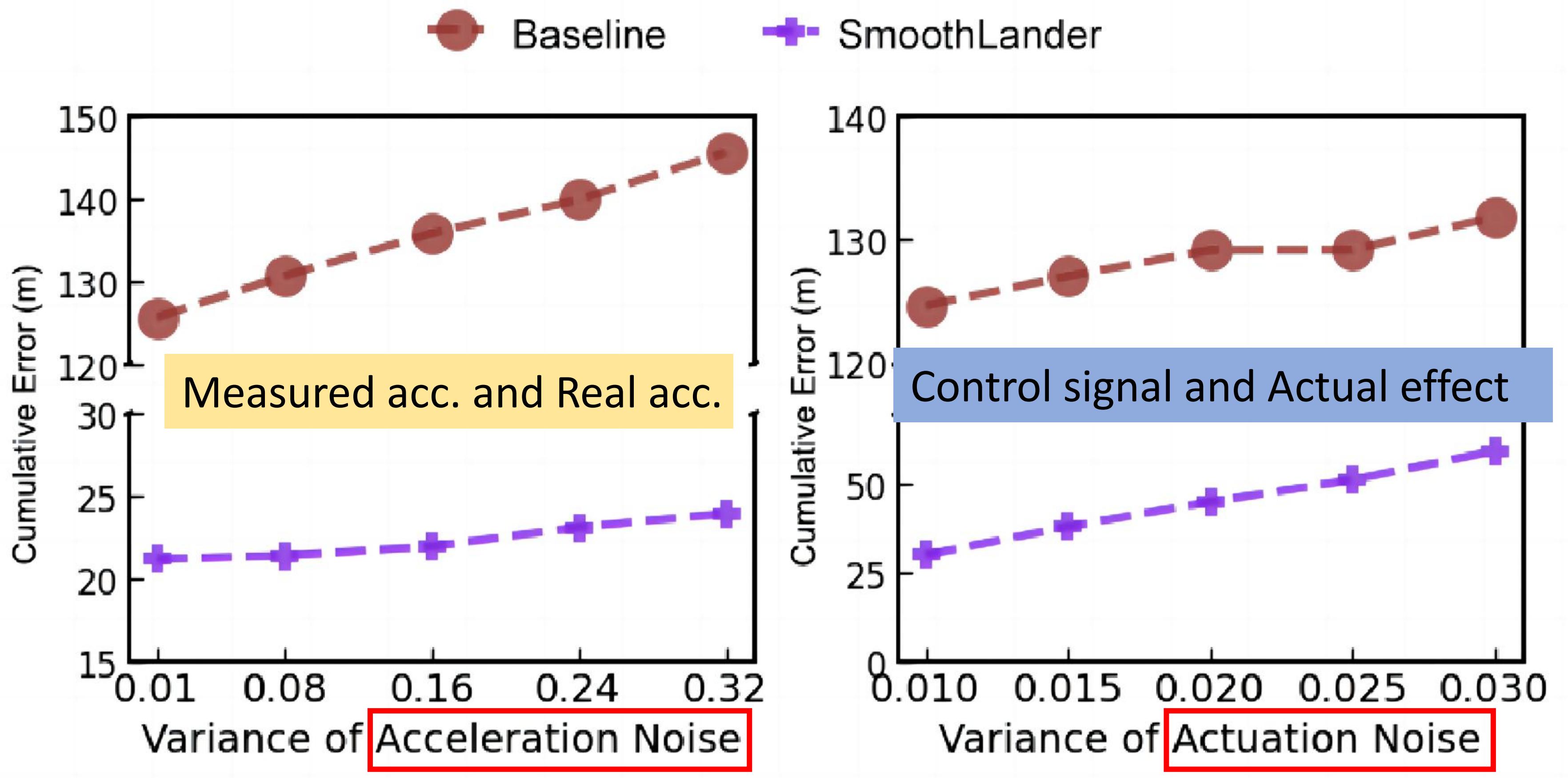

Trajectory of landing crazyflie




Trajectory of 100-time simulations

# Performance

Control noise reduction:




- As uncertainty of noise increasing, errors increase.
- Errors of ours are always lower than the Baseline.

# Summary of SmoothLander

- Design a RL-based landing control system by considering the future interaction with GE.
- Propose a physical feature-based method to generate training data in the RL.
- Evaluate the system through both physical feature-based simulation and real-world implementation.

# Thank You!

Presenter: Chenyu Zhao, TBSI
Email: zhaocy22@mails.tsinghua.edu.cn