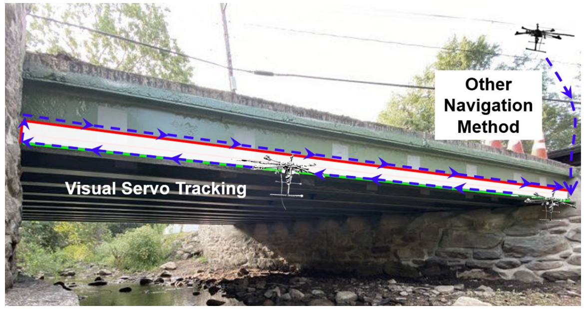
Physics-infused Learning for Aerial Manipulator in Winds and Near-Wall Environments

Yiming Zhang and Junyi Geng
The Pennsylvania State University

Aerial Manipulation

AERIAL ROBOTS & INTERACTION

Aerial manipulation provides a new solution to manipulation tasks at high altitudes and in otherwise inaccessible or hazardous environments.



He et al., IROS 2023

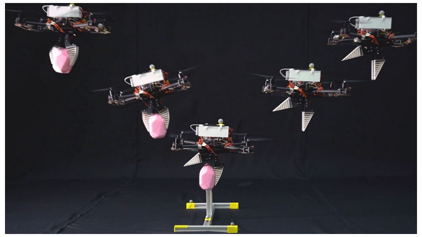
Peng et al.,
Nature Communications, 2025

Lloyd et al.,
IEEE/ASME Transactions on Mechatronics, 2025

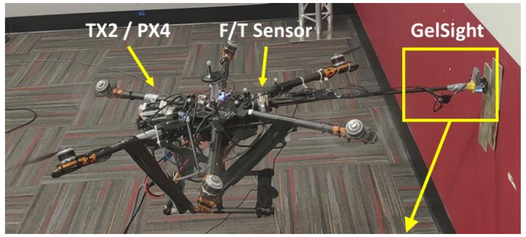
Li et al., Drones, 2024

Existing Applications

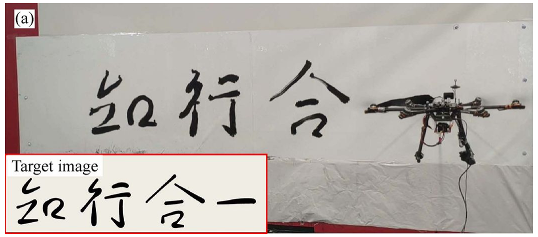
Most existing approaches are developed and validated in controlled indoor environments.



Pick-and-place
Appius et al., IROS, 2022

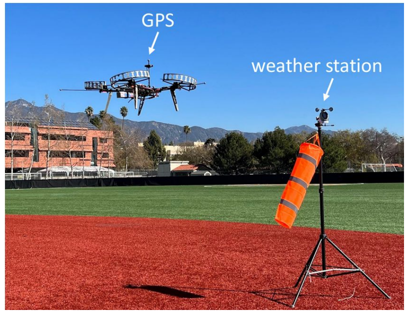


Contact Inspection
Guo et al., ICRA, 2024



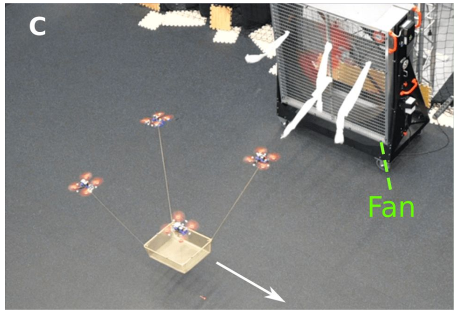
Aerial Writing
Guo et al., RA-L, 2024

Exceptions rely on controller robustness or generic disturbance rejection, rather than explicitly modeling ambient wind or considering proximity-induced effects.



O'Connell et al. Science Robotics, 2022

Cao et al., Nature, 2025

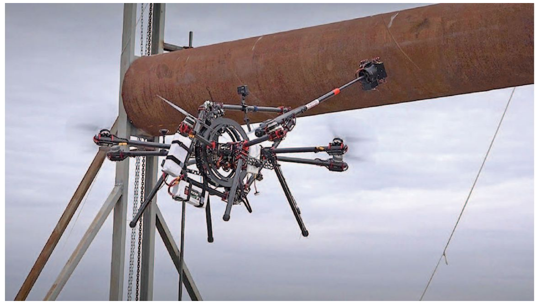


Sun et al. Science Robotics, 2025

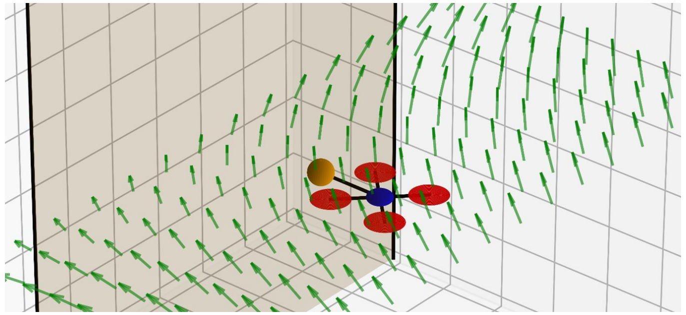
Challenges and Our Contribution

Aerial manipulation is affected by complex aerodynamic disturbances:

- Strong ambient wind
- Proximity-induced effects near structures



Ollero et al.,
IEEE Robotics & Automation Magazine 2018



Limitations of existing approaches:

- CFD-based modeling: computationally expensive
- Simplified models: inaccurate in strong wind and unable to capture near-wall aerodynamic effects

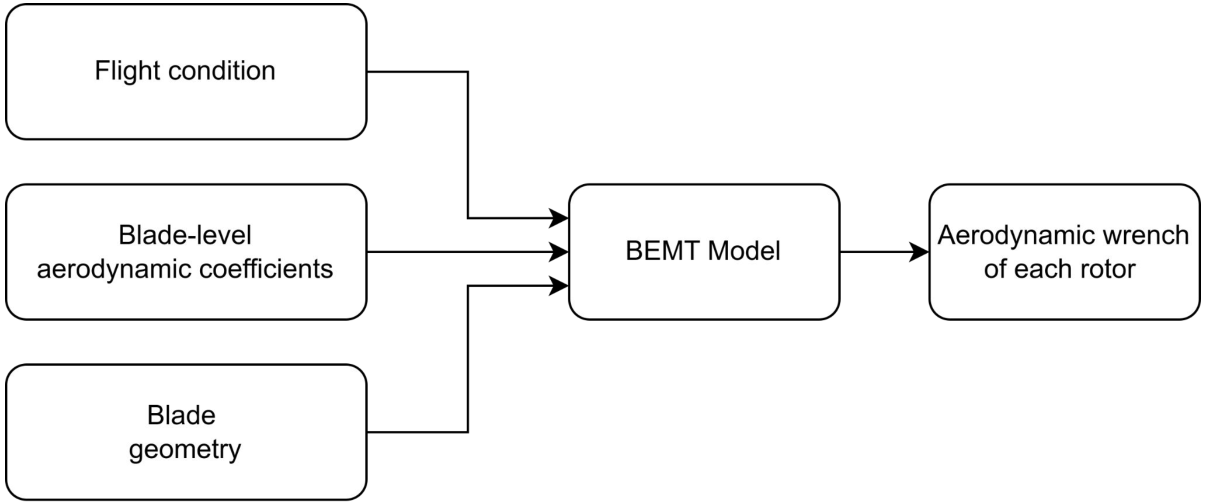
We propose to use an online computation friendly method to handle disturbance from near-wall wind effects.

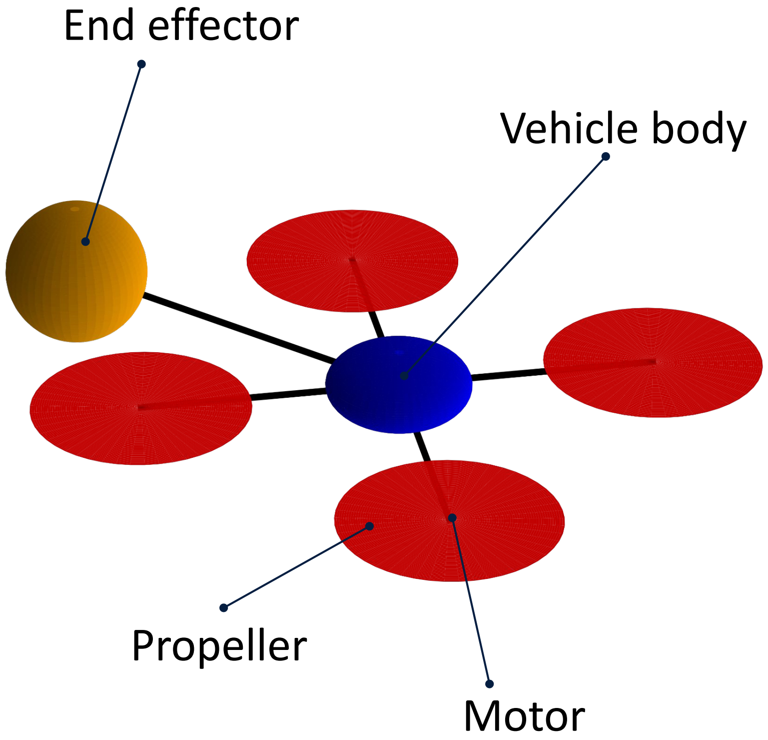
Modeling

Drone and end-effector dynamics: $\mathbf{M}\dot{\mathbf{V}} + \mathbf{C}\mathbf{V} = \boldsymbol{\tau}_{\text{ext}} + \mathbf{G}$

Motor speed dynamics: $\dot{n}_i = \frac{1}{T_m}(n_{i,\text{cmd}} - n_i)$

Aerodynamics (Blade Element Momentum Theory) to accurately model strong wind and near-structure interactions:





Methods – Overview

Integrating disturbance compensator into the controller:

$$\mathbf{M}\dot{\mathbf{V}} + \mathbf{C}\mathbf{V} = \boldsymbol{u} + \boldsymbol{\tau}_d + \mathbf{G}$$

$$\boldsymbol{u} = \boldsymbol{u}_{feedforward} + \boldsymbol{u}_{feedback} - \hat{\boldsymbol{\tau}}_d$$

estimated disturbance wrench

force components only $\hat{\boldsymbol{\tau}}_d = [\hat{\mathbf{F}}_d,\ \mathbf{0}]^\top$

The estimated disturbance force is decomposed into three parts:

$$\hat{\mathbf{F}}_d = \mathbf{F}_{\text{phys}} + \mathbf{F}_{\text{res}} + \hat{\boldsymbol{\varepsilon}}$$

$\mathbf{F}_{\text{phys}}$ Physics-based prediction: is good generalization but has limited expressiveness

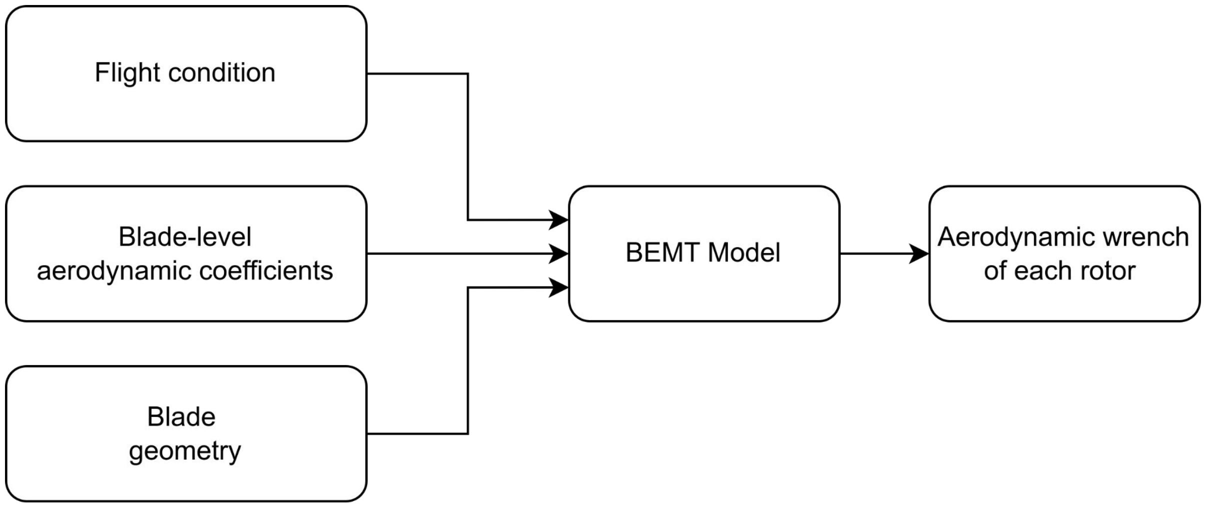
$\mathbf{F}_{\text{res}}$ Learned residual prediction: is flexible in fitting but has limited generalization

$\hat{\boldsymbol{\varepsilon}}$ Online adaptation: can estimate unmodeled disturbances but has latency

Methods – Physics-Based Learning

AERIAL ROBOTS & INTERACTION

We will use BEMT as physics based prediction model. However, blade-level aerodynamic coefficients are usually unknown.



$$\mathbf{F}_i = \mathbf{F}_{\text{BEMT}}(n_i,\ \mathbf{v}_{\text{wind},i},\ \mathbf{v}_{\text{prop},i}\ ;\ c_{\ell,1},\ c_{\ell,2},\ c_d,\ \alpha_0,\ \theta(\cdot),\ c(\cdot))$$

$$\mathbf{T}_i = \mathbf{T}_{\text{BEMT}}(\underbrace{n_i,\ \mathbf{v}_{\text{wind},i},\ \mathbf{v}_{\text{prop},i}}_{\text{Flight condition}}\ ;\ \underbrace{c_{\ell,1},\ c_{\ell,2},\ c_d,\ \alpha_0,}_{\text{Blade-level aerodynamic coefficients}}\ \underbrace{\theta(\cdot),\ c(\cdot)}_{\text{Blade geometry}})$$

Flight condition → From test data

Blade-level aerodynamic coefficients → **To be determined**

Blade geometry → Directly measurable

We will collect some test data and solve a parameter learning problem to find the best-fit aerodynamic coefficients.

Methods – Physics-Based Learning

Learn the aerodynamic coefficients to complete the BEMT model, by minimizing the prediction error of net aerodynamic force:

$$x_a^* = \arg\min_{x_a} \|\bar{\mathbf{F}}_a(x_a) - \mathbf{F}_{\text{sensed}}\|_2^2$$

Estimated net aerodynamic force:

$$\bar{\mathbf{F}}_a(x_a) = \sum_{i=1}^{4} \mathbf{F}_{\text{BEMT}}(n_i,\ \mathbf{v}_{\text{wind},i},\ \mathbf{v}_{\text{prop},i}\ ;\ x_a,\ \theta(\cdot),\ c(\cdot))$$

Aerodynamic coefficients to be learned:

$$x_a = (c_{\ell,1},\ c_{\ell,2},\ c_d,\ \alpha_0)$$

We can use free flight data under various wind conditions (near-wall condition is only optional) to learn this model.

$$\bar{\mathbf{F}}_a = (\underbrace{\bar{F}_x,\ \bar{F}_y}_{\text{Take lateral component as disturbance}},\ \bar{F}_z)^\top$$

Take lateral component as disturbance

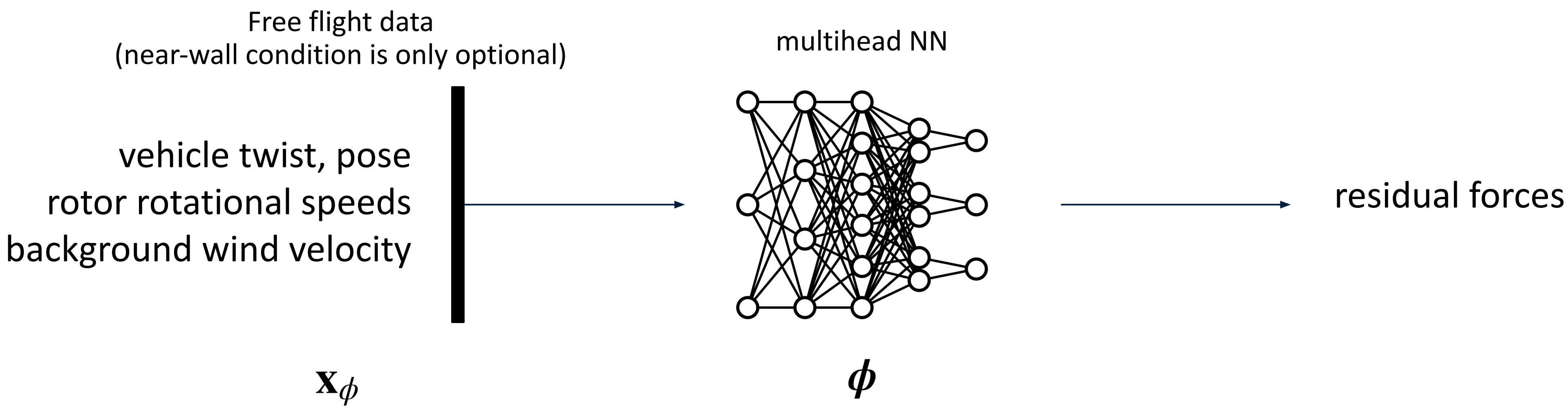
Thrust-dominated component

$$\bar{\mathbf{F}}_{a,\text{lat}} = (\bar{F}_x,\ \bar{F}_y,\ 0)^\top$$

$$\mathbf{F}_{\text{phys}} = \bar{\mathbf{F}}_{a,\text{lat}}$$

Methods – Residual Learning

Multihead network:



Use the network to minimize the mismatch between the BEMT-predicted forces and the measured forces:

$$\mathcal{L} = \sum_{i=1}^{N} \left\| \mathbf{F}_{\text{sensed}}^{(i)} - \bar{\mathbf{F}}_{a}^{(i)} - \boldsymbol{\phi}\left(\mathbf{x}_{\phi}^{(i)}\right) \right\|^2$$

$$\mathbf{F}_{\text{res}} = \boldsymbol{\phi}(\mathbf{x}_\phi)$$

Methods – Online Adaptation

Online adaptation corrects unmodeled disturbances using force residuals, including aerodynamic and contact-induced effects:

Residual update:

$$\dot{\hat{\boldsymbol{\varepsilon}}} = -\lambda\hat{\boldsymbol{\varepsilon}} - \mathbf{P}\mathbf{R}^{-1}(\hat{\boldsymbol{\varepsilon}} - \boldsymbol{y}) + \mathbf{P}\mathbf{H}\mathbf{s}$$

Covariance update:

$$\dot{\mathbf{P}} = -2\lambda\mathbf{P} + \mathbf{Q} - \mathbf{P}\mathbf{R}^{-1}\mathbf{P}$$

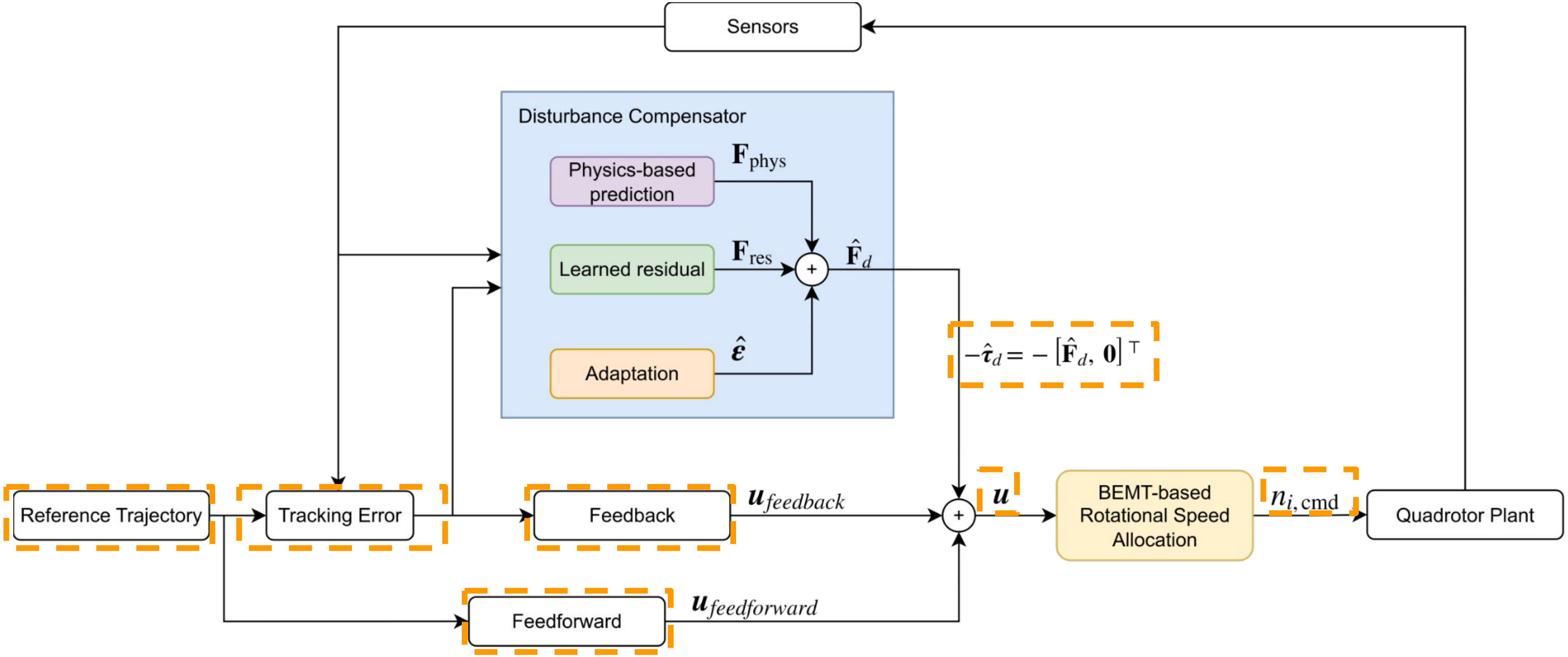
Innovation:

$$\boldsymbol{y} = \mathbf{F}_{\text{sensed}} - \left(\bar{\mathbf{F}}_a + \boldsymbol{\phi}(\mathbf{x}_\phi)\right)$$

λ : the regularization factor
$\mathbf{R}$ and $\mathbf{Q}$: the gain matrices
$\mathbf{P}$: the covariance matrix
$\mathbf{s}$: the control error
$\mathbf{H}$: the coefficient

Methods – Summary

The controller is augmented with the proposed disturbance compensation.



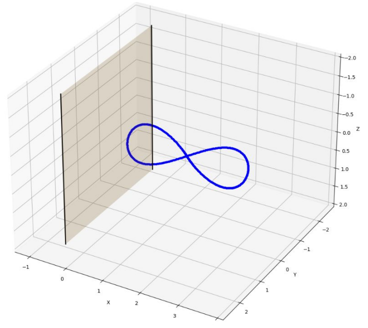
Simulation Environment Setup

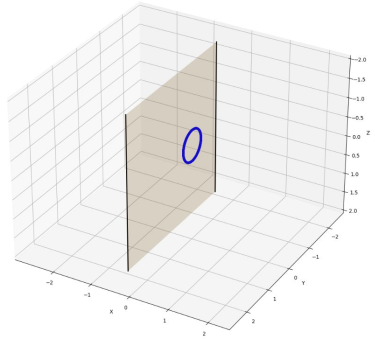
Simulate an aerial manipulation near a tall building subject to strong wind:

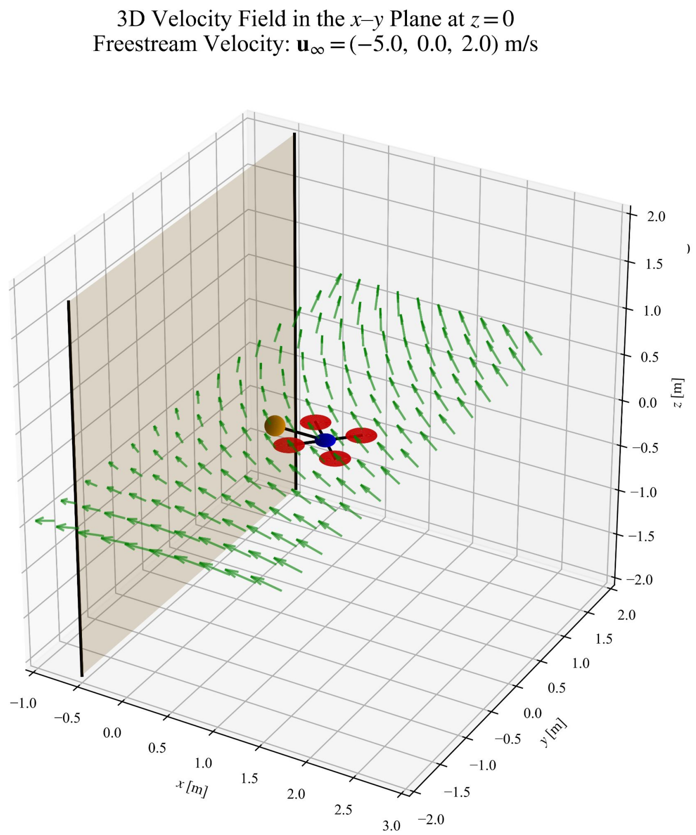
- Vertical wall
- Horizontal and vertical wind
- End-effector wall-contact
- Sensor misalignment and noise
- Finer BEMT discretization in simulation than in learned model

The quadrotor has two tasks:

- Figure-eight near-wall flight
  - spatially distributed wind field
- Circular wall-contact flight
  - spatially distributed wind field
  - unmodeled contact disturbance





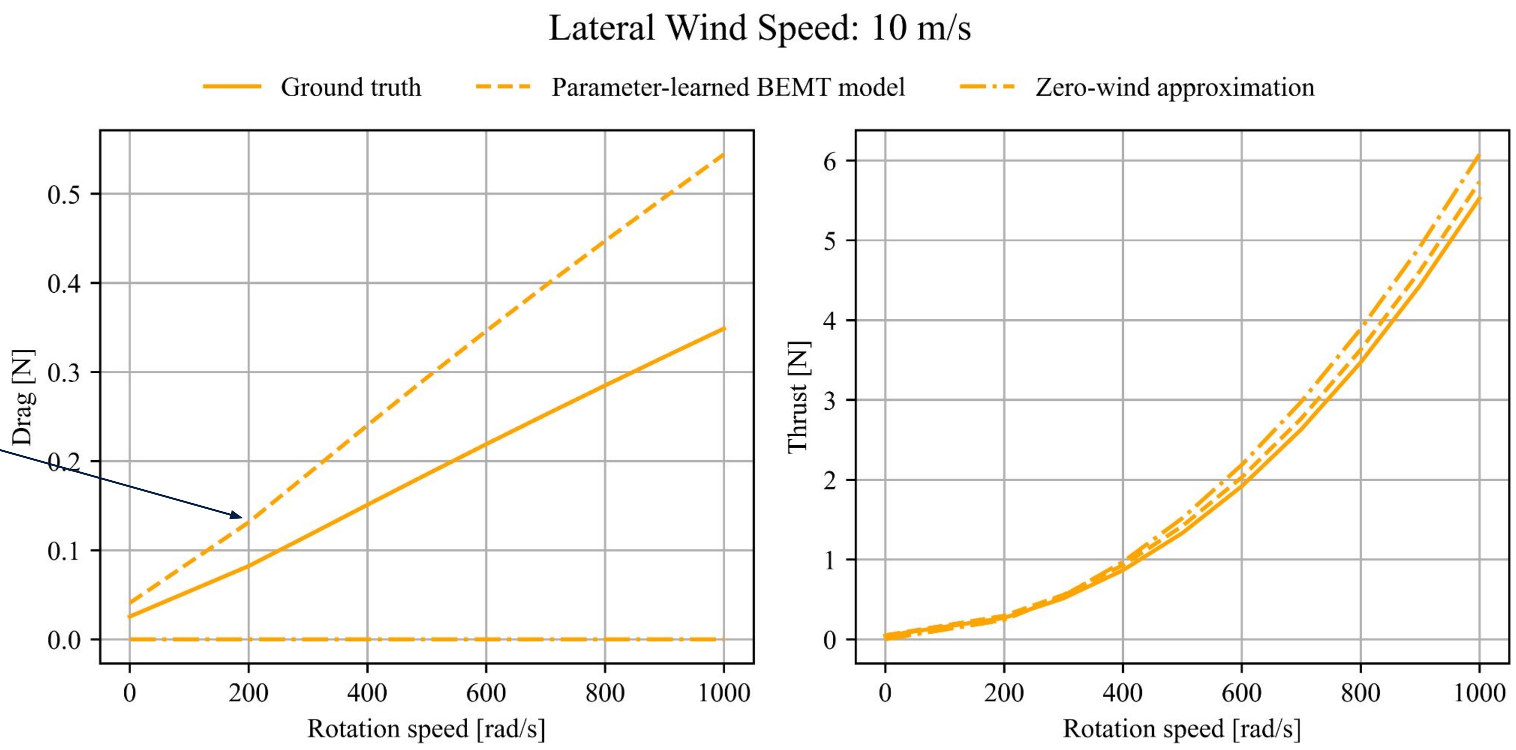


Result – Offline Physics Learning

Comparison between parameter-learned BEMT model and a baseline (zero-wind approximation):

- **Zero-wind approximation**: a static thrust map assuming no ambient wind, as commonly adopted in existing flight controllers.

BEMT-based prediction captures drag

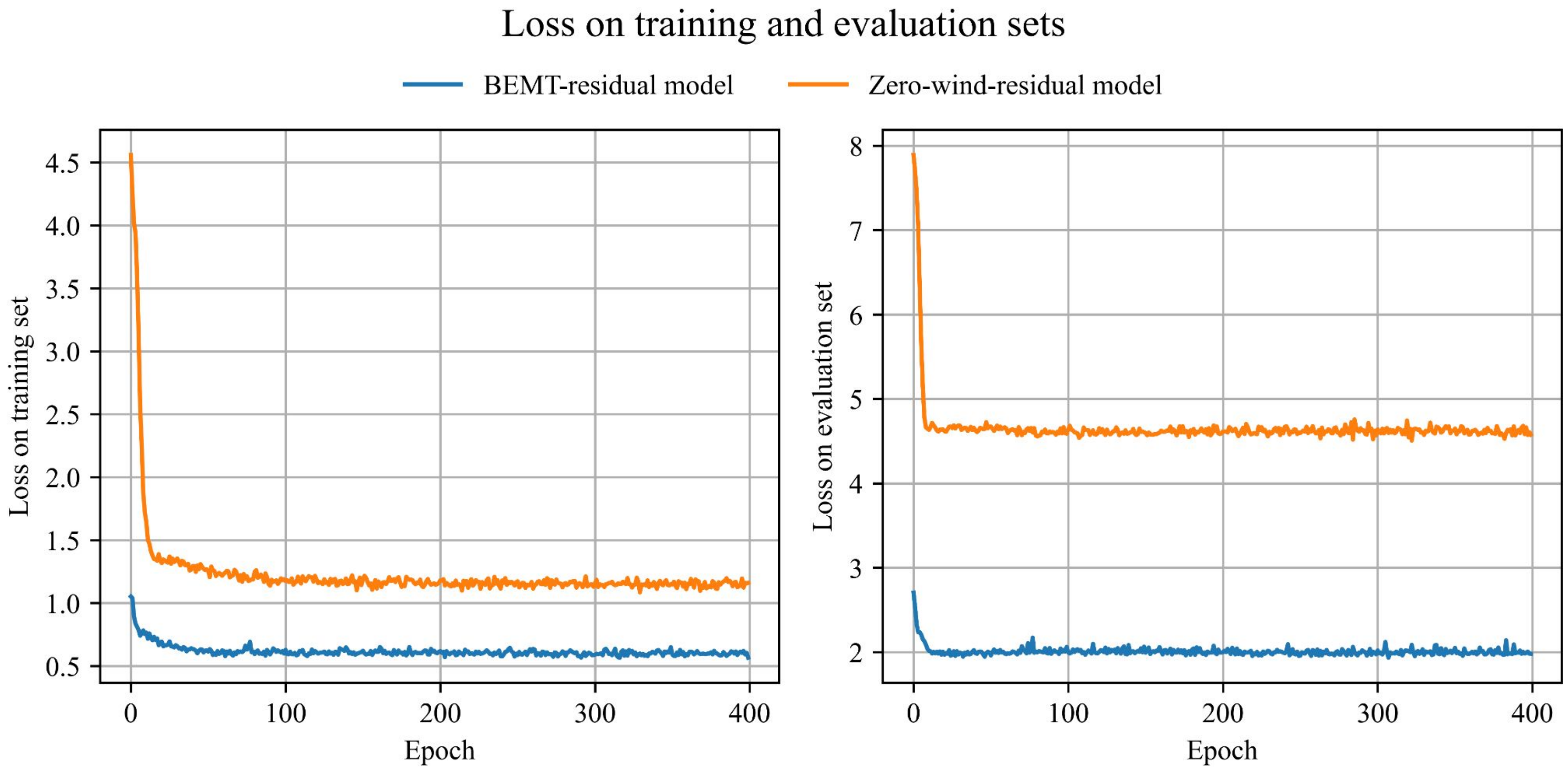


Parameter-learned BEMT has better drag and thrust prediction.

Result – Offline Residual Training

Comparison between learning residual from the proposed physics model and a baseline:

- **BEMT-residual**: learns residuals relative to a physics-based BEMT model
- **Zero-wind residual**: learns residuals relative to a static thrust map

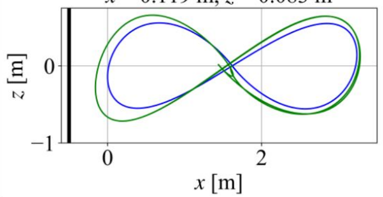
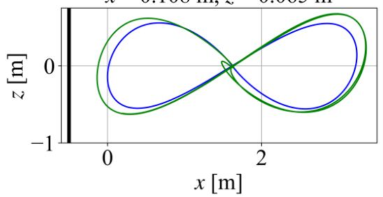
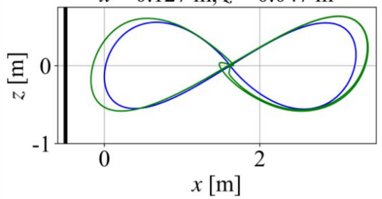
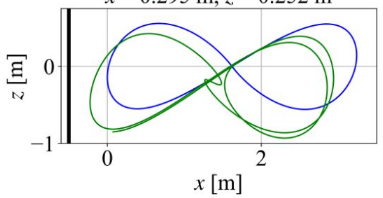
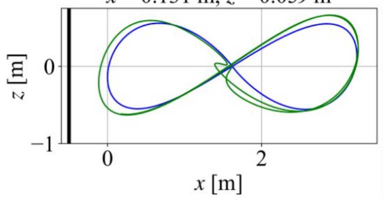
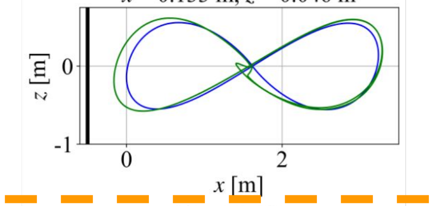
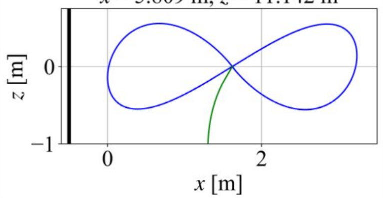
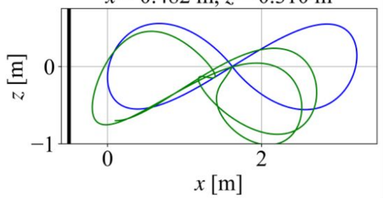
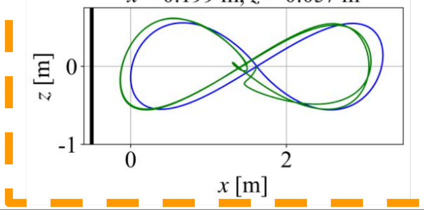


BEMT-residual model:

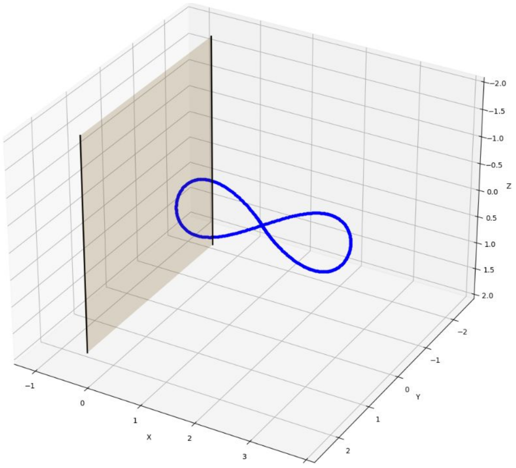
- Better convergence
- Better generalizability

Result – Figure-eight Task

Table 1: Figure-eight tracking performance

Wind [m/s]	No Compensator	DAIML	Proposed BEMT-Based Method
−4	Position RMS Error: $x = 0.119$ m, $z = 0.083$ m 	Position RMS Error: $x = 0.108$ m, $z = 0.065$ m 	Position RMS Error: $x = 0.127$ m, $z = 0.047$ m 
−8	Position RMS Error: $x = 0.295$ m, $z = 0.252$ m 	Position RMS Error: $x = 0.131$ m, $z = 0.059$ m 	Position RMS Error: $x = 0.133$ m, $z = 0.046$ m 
−12	Position RMS Error: $x = 5.809$ m, $z = 11.142$ m 	Position RMS Error: $x = 0.482$ m, $z = 0.310$ m 	Position RMS Error: $x = 0.199$ m, $z = 0.057$ m 

Wind values shown in the first column are applied equally along the x- and z-directions.



Proposed method:

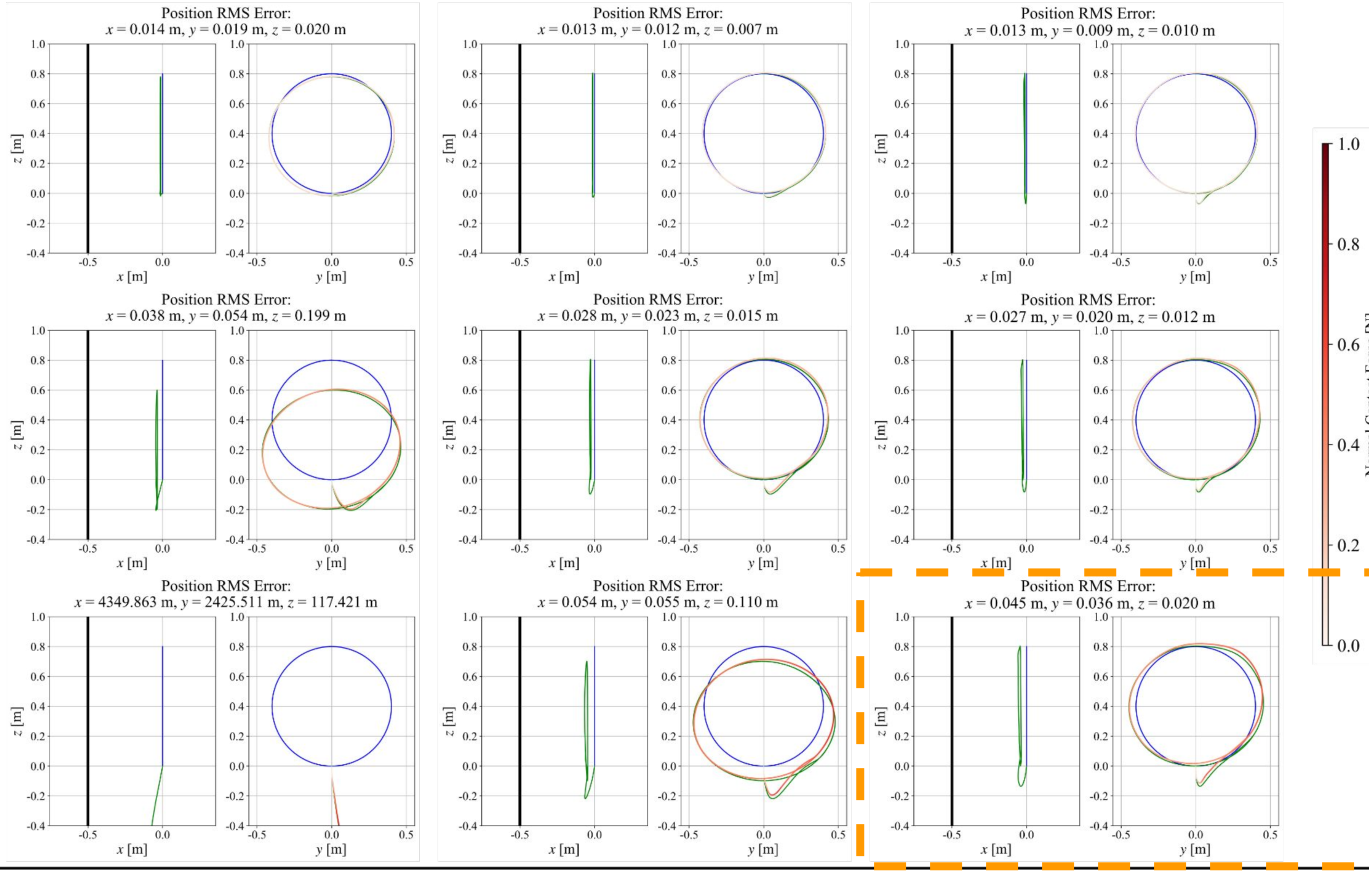
- More robust to disturbance
- Smaller tracking error

DAIML is a state-of-the-art disturbance compensation method without physics model infused.
(O'Connell, M., Shi, G., Shi, X., Azizzadenesheli, K., Anandkumar, A., Yue, Y., and Chung, S.-J., "Neural-fly enables rapid learning for agile flight in strong winds," Science Robotics, Vol. 7, No. 66, 2022, p. eabm6597.)

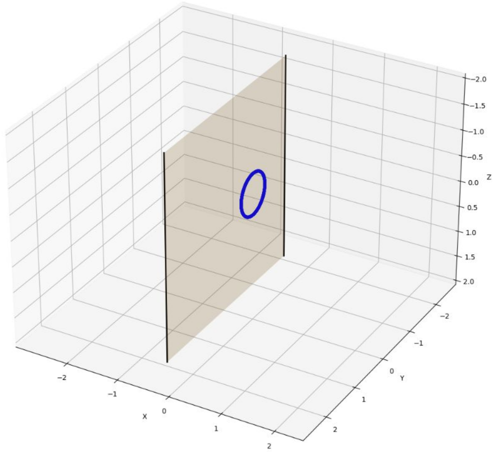
Result – Wall-contact Task

Table 2: Wall-contact tracking performance

Wind [m/s]	No Compensator	DAIML	Proposed BEMT-Based Method
−4	Position RMS Error: $x = 0.014$ m, $y = 0.019$ m, $z = 0.020$ m	Position RMS Error: $x = 0.013$ m, $y = 0.012$ m, $z = 0.007$ m	Position RMS Error: $x = 0.013$ m, $y = 0.009$ m, $z = 0.010$ m
−8	Position RMS Error: $x = 0.038$ m, $y = 0.054$ m, $z = 0.199$ m	Position RMS Error: $x = 0.028$ m, $y = 0.023$ m, $z = 0.015$ m	Position RMS Error: $x = 0.027$ m, $y = 0.020$ m, $z = 0.012$ m
−12	Position RMS Error: $x = 4349.863$ m, $y = 2425.511$ m, $z = 117.421$ m	Position RMS Error: $x = 0.054$ m, $y = 0.055$ m, $z = 0.110$ m	Position RMS Error: $x = 0.045$ m, $y = 0.036$ m, $z = 0.020$ m



Wind values shown in the first column are applied equally along the x- and z-directions.



Proposed method:

- More robust to disturbance
- Better wall-contact consistency

Conclusion

- We proposed a physics-infused disturbance-rejection framework.
- The framework integrates three components:
  - a physics based aerodynamic model with parameters learned
  - a neural-network based residual force model
  - an online adaptive disturbance observer
- By simulation, we showed that the proposed controller achieves improved robustness, smaller tracking errors, and more consistent wall contact
- The proposed framework represents a step toward reliable aerial manipulation in realistic environments.

Thank you!