

# A Narwhal-Inspired Sensing-to-Control Framework for Small Fixed-Wing Aircraft

Fengze Xie<sup>\*1</sup>, Xiaozhou Fan<sup>\*1</sup>, Jacob Schuster<sup>1</sup>, Yisong Yue<sup>1</sup>, Morteza Gharib<sup>1</sup>

**Abstract**—Fixed-wing unmanned aerial vehicles (UAVs) offer endurance and efficiency but lack low-speed agility because its highly-coupled dynamical model. We present an end-to-end sensing-to-control pipeline that combines bio-inspired hardware instrumentation, physics-informed dynamics learning, and convex control allocation. Measuring airflow on a small airframe is difficult as near-body aerodynamics, propeller slipstream, control surfaces actuation, and the present of gusts would distort pressure signals. Inspired by the narwhal whale’s signature protruding tusk, we stick our in-house developed multi-hole probes far into the upstream, and complement it with sparse yet carefully placed wing pressure sensors for local flow measurement. A data-driven calibration scheme was adopted to map the pressure signal of the probes to airspeed and flow angles. We next learn a control-affine dynamical model using Pitot-tube estimated airspeed and flow angles as inputs, along with sparse sensor measurements. We implement a soft left/right symmetry regularizer that improves the model’s identifiability under partial observability and limits confounding between wing pressures and flaperon inputs. Desired wrenches (forces and moments output) are realized by a regularized least-squares optimizer that yields smooth, trimmed actuation. Wind tunnel studies, across a wide range of parameter space, demonstrate that adding wing pressures reduces force-estimation error by 25%–30%, the proposed model degrades less under distribution shift (about 12% versus 44% for an unstructured baseline), and force tracking improves with smoother inputs, including a 27% reduction in normal-force RMSE relative to a plain affine model and 34% relative to an unstructured baseline.

## I. INTRODUCTION

Unmanned aerial vehicles (UAVs) have progressed rapidly across multiple platform classes. There are three major categories: flapping-wing, quadrotor, and fixed-wing aerial vehicles. Hybrid models are common. For example, a VTOL platform can use rotorcraft lift for takeoff and landing and a fixed-wing layout for efficient cruise [1]. Flapping wing vehicles are superior at birds- or insect-size due to their capability to harvest unsteadiness in the flow [2], [3], and has seen considerable research in understanding its mechanics with robotics instrumentation [4], [5], but may still be years from becoming viable competitors to fixed-wing or quadrotors, due to their extremely interdisciplinary nature that couples unsteady aerodynamics, design, flight control and material selection [6], [7]. Quadrotors provide precise hovering and aggressive maneuvering in clutter with comparatively simple mechanics, enabling inspection, cinematography, indoor logistics, and first response [8], [9]. High

This work is supported by the Technology Innovation Institute. \* Equal contribution. <sup>1</sup> California Institute of Technology, CA 91106, USA. Corresponding author: Xiaozhou Fan (xiaozhoufan@hkust-gz.edu.cn), current position: Assistant Professor at HKUST (Guangzhou).

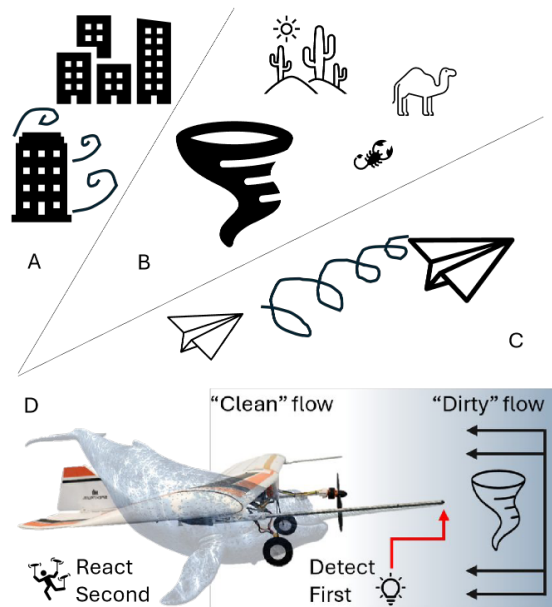


Fig. 1. Fixed-wing vehicles operate across highly diverse terrain and environments, such as (A) through populated urban environments, (B) across vast deserts, or (C) trailing behind other aerial vehicles in formation flight. The underlying challenging gust are vortex dynamics which are highly time-varying. (D) To stay ahead of the “curve”, we built in-house developed, avionic sensors that measures incoming gust (in loose language, “dirty” flow) before it arrives at the wing, and affect drone’s dynamics. This approach is inspired by the Narwhal whale, where its tooth protrudes forward, and detects sea water salinity and potentially pressure signals [14].

control authority and a well-characterized mapping from motor commands to body forces and moments explain their agility under disturbance [10]. The mapping depends little on forward airspeed, and propeller thrust is well approximated by a quadratic function of angular velocity. Recent work demonstrates robust autonomous flight in challenging conditions [11], [12]. In contrast, fixed-wing vehicles offer long endurance, high cruise efficiency, and wide-area coverage, which is well-suited to explore uncharted territory, environmental monitoring, and search mission (Fig. 1A-C), but at the cost of limited low-speed agility and control effectiveness due to highly-coupled dynamics between flight parameters, such as airspeed, angle of attack and sideslip angles, just to name a few [13]. In addition, drones in general need to consider a good trade-off between sensing and payload carrying capability, and while optical and inertial sensors are essential, they may not be able to sense clear-air turbulence, which is detrimental and poses life-threatening danger to

the safe operation of any drones. To make matters worse, drones often maneuver at a fast rate, and thus, processing high-dimensional data like images may introduce significant delay in the control scheme.

To combine fixed-wing endurance with multirotor agility, we aim to learn a trustworthy mapping from control-surface deflections to aerodynamic forces and moments, supported by low-dimensional onboard sensors that reveal flow information not otherwise available. Achieving this goal requires two complementary capabilities: first, reliable real-time sensing of the local flow state; and second, models that capture nonlinear, sensor-dependent aerodynamics well enough to enable agile flight. To this end, we build an end-to-end sensing-to-control pipeline on a standard fixed-wing platform using *sparse wing pressure sensors* and a *low-cost, in-house multi-hole probe* (we use multi-hole probe, Pitot tube, and Pitot probe interchangeably). A central challenge is that near-body flow distortions induced by the fuselage, propeller slipstream, and control-surface deflections can contaminate pressure measurements. Inspired by standoff sensing in nature, such as the narwhal whale’s tusk [14], we mount an extended Pitot probe on a long boom (Fig. 1D) to increase the sensor distance from the airframe, reduce self-induced aerodynamic interference, and promote earlier detection of incoming flow changes. We then introduce a three-part physics-informed, data-driven approach. First, we calibrate the multi-hole pressure measurements to estimate airspeed and flow angles. Second, we learn an observation-to-wrench model that links onboard sensing to aerodynamic forces and moments while enforcing a soft left–right symmetry prior. Third, we allocate control surfaces to track desired forces and moments using smooth, well-trimmed commands. Wind-tunnel experiments spanning a wide range of airspeeds, angles of attack, sideslip, and gust profiles show more accurate force estimation and improved tracking than unstructured baselines, with clear generalization beyond the training domain.

## II. RELATED WORKS

**Fixed-Wing Drone Aerodynamics:** Fixed-wing drone aerodynamics presents several interrelated challenges. Unlike its bigger airliner counterpart that carries passengers, small drones have limited payload, rendering redundant sensing unrealistic, and its lightweight is often much more susceptible to atmospheric gust disturbance. At moderate to high angles of attack, airflow over the wing can separate, leading to abrupt loss of lift and unstable pitching behavior. Embedding pressure sensors within the wing or fuselage can provide early warning of flow separation, enabling corrective control [15]. However, the wing itself induces an upstream updraft [16], so any pressure-based avionics must be carefully calibrated to account for this local flow distortion. Furthermore, changes in pitch (angle of attack) and yaw (sideslip) are coupled: when a fixed-wing drone banks, it not only needs differential flap and rudder adjustments to coordinate the turn, but also must increase airspeed to offset reduction of lift as part of it now becomes centripetal force [17].

These coupled effects make precise sensing and adaptive control essential for reliable maneuvering. Classic flight-control-oriented aerodynamic analysis [18] relies heavily on aerodynamic derivatives—partial derivatives of aerodynamic forces and moments with respect to states such as angle of attack, sideslip angle, angular rates. This approach is valid primarily in small-angle regimes: the linearization assumes nearly steady, attached flow where lift and moment coefficients vary smoothly and predictably.

**Fixed-Wing Drone Automation:** Research on fixed-wing drone automation is converging on three active directions. First, wind-aware guidance and robust control incorporate realistic gust models and disturbance observers to improve path following and disturbance rejection [19]. Second, real-world reinforcement learning has progressed from simulation to outdoor flights by using domain randomization and latency/noise modeling, achieving data-efficient attitude control [20]. Third, comparative evaluations under turbulence indicate that model-based approaches often match or exceed model-free methods and PID in tracking and robustness while yielding smoother actuation, although gains can shrink in severe perturbations [21]. Recent surveys catalog tasks and toolchains and highlight the absence of standardized metrics and benchmarks, which limits fair comparison and reproducibility [22]. These trends motivate integrated, wind-aware pipelines that combine sensing, structure-aware modeling, and control allocation for agile fixed-wing autonomy.

**Physics-Informed Machine Learning:** Research at the interface of classical modeling and learning has shown that embedding physical structure in learned models improves accuracy, interpretability, and sample efficiency [23], [12], [24], [25]. Representative approaches encode governing laws or invariants directly in the architecture or loss, including PDE-constrained learning and PINNs [26], [27], mechanics-aware networks that preserve energy or momentum [28], [29], and structure-aware representations that exploit symmetry and locality [30], [31], [32]. By constraining the hypothesis space to physically plausible behaviors, these methods improve robustness under distribution shift and reduce the reliance on dense instrumentation or exhaustive data collection. For fixed-wing flight, physics-informed learning is especially relevant because aerodynamic forces and moments depend on the airspeed in a strongly nonlinear way, and the mapping from sparse, partially observed sensors to these quantities is hard to identify. Instead of relying on PDE-constrained learning or PINNs, we encode domain structure directly by using how control surface deflections generate aerodynamics, and by utilizing fixed-wing drones left-right physical symmetries configuration, and baking it into the control-affine structure of the learned model.

## III. HARDWARE AND SYSTEM

We design and construct a multi-hole probe system to measure local airflow direction and magnitude (Fig. 2A). The probe consists of five Albion aluminum microtubes (0.5 mm outer diameter (OD)  $\times$  0.3 mm inner diameter (ID)) arranged within a steel casing (5 mm OD  $\times$  4 mm ID). Each

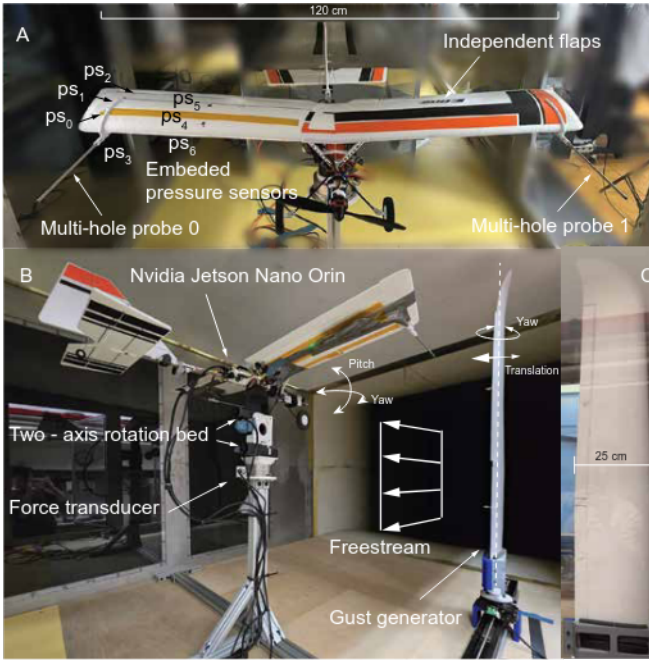


Fig. 2. The model fixed-wing drone, model positioning system, and the gust generator in a wind tunnel. (A) The fixed-wing are instrumented with in-house developed avionics as well as an onboard Nvidia Jetson Nano Orin computer. (B) The test environment in the wind tunnel includes a gust generator as well as an in-house developed two-axis rotation bed (or model position system).

aluminum tube is cut to approximately 32 cm in length and threaded through a black v4 resin-printed nose cone and end caps. Loctite super glue is applied to fix the tubes at both the nose cone and rear end caps. To facilitate connections with the onboard pressure sensors, each aluminum tube is linked to a short segment of silicon tubing (3mm OD x 1.5 mm ID) through resin-printed connectors, which are tightly sealed with glue to prevent air leakage. These five tubes then attached directly to the bottom ports of 5 pressure sensors on a custom circuit board, which interface with a multiplexer routed to a Teensy 4.1 microcontroller, a compact ARM-based board with high-rate PWM/ADC and low-latency I/O for real-time interfacing.

The model airplane (Fig. 2A) is a commercially available E-flite Slow Ultra Stick with a 1.2m wingspan. It uses a propeller, rudder, elevator, and left and right flaperons, which are combined flap-aileron surfaces driven independently. The spanwise size of the wind tunnel is 1.8m, considerably larger than the drone, thus we are clear of any close-to-wall effect. We mount two in-house fabricated and calibrated five-hole probes close to the left and right wing tips, respectively, and they protrude about 30 cm from the leading edge of the wing. Seven additional pressure sensor taps (named  $ps_i$ , where  $i = 0$  to 6) are placed on the right wing, flush with the surface.  $ps_0 - ps_3$  are placed near the tip of the wing, and  $ps_4 - ps_6$  are placed in the middle-span. In addition,  $ps_0$  and  $ps_4$  reside on the upper (suction side) of the leading edge,  $ps_1$  and  $ps_5$  mid-chord, and  $ps_2$  trailing edge.  $ps_3$  and  $ps_6$  are also close to the leading edge, but are under the wing (pressure side).

The model fixed-wing is then (Fig. 2B) placed on a two-axis in-house developed model positioning system, which varies the pitch (angle of attack) and yaw (sideslip angle) of the aircraft. An ATI six-axis force transducer records in realtime the forces and torques. The gust generator is a vertically-mounted wing, with length 1.2m and chord 25cm, and is positioned on the right side of the drone, 70cm away from the center of the station in both the streamwise and spanwise direction. The gust generator is capable of yawing around its axis as well as translating back and forth on a linear stage. With a chord on the same order as the drone, the gust generator induces shear flow at small yaw angles and transitions to periodic vortex shedding as the yaw angle increases. The entire system is installed in a wind tunnel that produces clean, cyclic airflow with controllable airspeed.

For computing resources, an NVIDIA Jetson Orin Nano, a compact edge-AI module with an ARM CPU and a CUDA-capable GPU designed for embedded robotics, is mounted onboard and serves as the primary onboard computer. The Jetson sends control surface commands and receives pressure sensors measurements via the Teensy 4.1 microcontroller. The setup uses a workstation connected to the two-axis rotation rig, the force/torque transducer, and the gust generator. Both the workstation and the NVIDIA Jetson Nano run Ubuntu 20.04. The Jetson acts as the client, and the workstation acts as the system server. All actuation and inter-process communication use the robotic operating system (ROS) Noetic. Clocks across devices are synchronized to ensure consistent timestamps and coordinated execution.

#### IV. METHODOLOGY

We present a sensing-to-control pipeline for small fixed-wing aircraft that couples data-driven probe calibration, physics-informed dynamics learning, and convex control allocation, as summarized by the flow chart in Fig. 3. First, we calibrate the five-hole probes: from normalized pressures, we estimate a dynamic pressure correction and the flow angles, which enable robust recovery of wind features across different environments. Next, we learn a physics-informed, control-affine dynamics model that maps observations to forces and torques. Finally, we compute control-surface deflections by solving a small regularized least-squares problem that tracks a desired wrench.

##### A. Pitot Tube Calibration

The multi-hole probe is mounted on the airplane for *in vivo* calibration, as the presence of the wing itself introduces updraft and would be better to consider these effects directly.

We adopt a data-driven calibration for multi-hole velocity probes[33], [34]. A probe tip with  $n$  taps provides dimensional pressures  $\{p_i\}_{i=1}^n$ . Let

$$p_{\max} = \max_i p_i, \quad p_{\min} = \min_i p_i, \quad \Delta p = p_{\max} - p_{\min},$$

and define the nondimensional coefficients

$$C_{p_i} = (p_{\max} - p_i) / \Delta p, \quad i = 1, \dots, n. \quad (1)$$

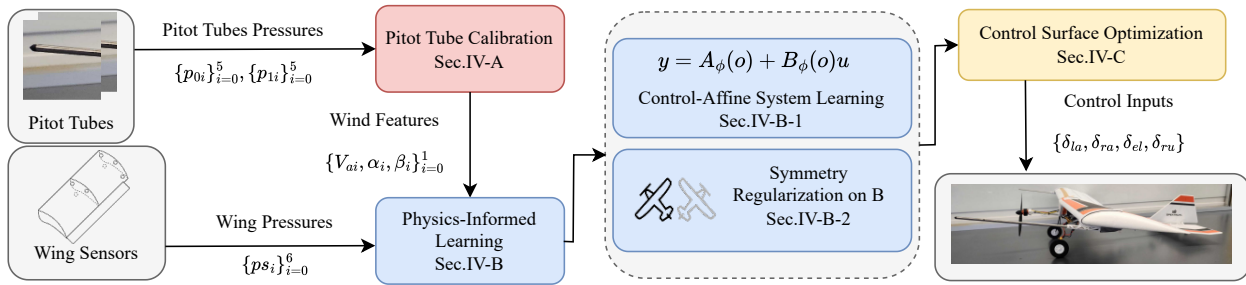


Fig. 3. **Sensor-to-control pipeline.** **Section IV-A:** Dual multi-hole Pitot and sparse wing-pressure measurements are calibrated to estimate the flow state  $(V_a, \alpha, \beta)$ . **Section IV-B:** A symmetry-aware, control-affine model maps onboard observations to aerodynamic forces and moments. **Section IV-C:** A convex controller then computes surface deflections to realize a desired wrench.

These coefficients are largely independent of airspeed and vary primarily with flow direction [34]. The approximation error in using  $\Delta p$  as a surrogate for dynamic pressure is captured by

$$C_d = \frac{1}{2} \rho V_a^2 / \Delta p, \quad (2)$$

where  $\rho$  denotes air density and  $V_a$  denotes the airspeed.

We realize the calibration with a neural network  $f$  that consumes all normalized pressures and outputs the dynamic-pressure correction and flow angles explicitly:

$$C_d, \alpha, \beta = f(C_{p_1}, \dots, C_{p_n}), \quad (3)$$

where  $\alpha$  denotes the angle of attack relative to the probe axis and  $\beta$  denotes sideslip angle relative to the probe axis. In deployment, we first compute  $C_{p_i}$  using (1), evaluate (3) to obtain  $C_d, \alpha, \beta$ , and then reconstruct the airspeed by inverting (2):

$$V_a = \sqrt{2 \Delta p C_d / \rho}. \quad (4)$$

In our experiments, we use five-hole probes, where  $n = 5$ .

### B. Physics-Informed Learning

Physics-informed learning augments data with a lightweight physics-based structure to improve sample efficiency and generalization. Classical physics-informed approaches (such as PINNs) typically penalize the PDE and boundary to enforce the governing laws [26]. In our low-speed fixed-wing setting with gusts and partial observability, we instead encode structural [31] and aerodynamic priors as constraints and regularizers on the network, which mitigates overfitting and yields a model whose inputs and outputs are naturally compatible with the control of fixed-wing drones.

Fixed-wing dynamics are nonlinear and vary under different wind conditions. Training directly on limited data tends to overfit to the training dataset and environment. Wing pressure sensors improve measurement coverage by capturing local flow effects that front-mounted probes miss. However, their signals are strongly coupled with flap deflections, which can introduce causal confounding and unstable gradients if used naively. We address these issues with a control-affine architecture and a symmetry-based regularizer on the control matrix, which improves parameter identification and stabilizes learning while allowing bounded asymmetric effects.

1) *Control-Affine System Learning:* Let  $o$  denote the observation vector that aggregates flow-state features from two calibrated five-hole Pitot probes and wing pressure sensors. Specifically,

$$o = [\hat{V}_{a,0}, \hat{\alpha}_0, \hat{\beta}_0, \hat{V}_{a,1}, \hat{\alpha}_1, \hat{\beta}_1, ps_0, \dots, ps_6]^\top,$$

where  $(\hat{V}_{a,j}, \hat{\alpha}_j, \hat{\beta}_j)$  are the estimated airspeed, angle of attack, and sideslip from probe  $j \in \{0, 1\}$ , and  $\{ps_i\}_{i=0}^6$  are the wing pressure sensors measurements. Let the control vector be

$$u = [\delta_{la}, \delta_{ra}, \delta_{el}, \delta_{ru}]^\top \in \mathbb{R}^4,$$

comprising left flaperons, right flaperons, elevator, and rudder deflections. We predict the six-dimensional wrench  $y = [F_x, F_y, F_z, T_x, T_y, T_z]^\top \in \mathbb{R}^6$  with a control-affine model

$$y = A_\phi(o) + B_\phi(o)u, \quad (5)$$

where  $A_\phi(\cdot)$  and  $B_\phi(\cdot)$  are neural networks that share the first  $k-1$  layers of a depth- $k$  backbone with neural network parameter  $\phi$ .

The control-affine parameterization is informed by fixed-wing aerodynamics. For relatively small angles of attack that are away from stall conditions, the incremental forces and moments induced by small control-surface deflections are approximately linear with deflection angles, and their effectiveness varies with the airspeed [35], [36]. The term  $A_\phi(o)$  captures the baseline aerodynamics due to the current flow and configuration, while  $B_\phi(o)$  captures the state-dependent control effectiveness, which changes with dynamic pressure reflected by wind features measured by Pitot tubes and local loading reflected in the wing pressure sensors. By conditioning both  $A$  and  $B$  on  $o$ , the model respects known near-linear control effects, allows effectiveness to vary across operating points, and improves generalization relative to an unstructured mapping  $y = F_\phi(o, u)$ .

2) *Symmetry Regularization on  $B_\phi$ :* Building on the control-affine parameterization, reliably identifying the flaperon-related columns of  $B_\phi(o)$  is challenging when training on datasets that combine wing-pressure measured on wings with flaperon commands. Without an explicit structural prior,  $B_\phi(o)$  often fails to recover the expected

small-deflection aerodynamic patterns, which biases control-effectiveness estimates and degrades closed-loop performance. The inherent left–right symmetry of the flaperons provides a principled calibration signal for these columns [31], [37]. At the same time, real-world flow asymmetries like sideslip angles and gusts can cause the estimator to assign state-dependent effects unevenly between the two inputs. This motivates a soft, symmetry-aware regularization rather than a hard equality constraint.

To address this, we impose a physics-informed symmetry prior on the flaperon columns while keeping it soft to accommodate real asymmetries. Columns 0 and 1 of  $B_\phi(o) \in \mathbb{R}^{6 \times 4}$  correspond to the left and right flaperons. Aerodynamic symmetry implies opposite effects on some channels and same-sense effects on others. We encode this with the sign vector  $\mathbf{s} = (1, -1, 1, -1, 1, -1)^\top$  and penalize deviations from the expected left–right relationship. We use a Huber-penalized regularizer that encourages symmetry but permits bounded deviations arising from sideslip and the upstream gust generator, so the model is not over-constrained in inherently asymmetric conditions. The loss is defined as:

$$\mathcal{L}_{\text{sym}} = \lambda \sum_{i=1}^6 \psi_{\delta_i} \left( (B_{\phi \cdot, 0} + \mathbf{s} \odot B_{\phi \cdot, 1})_i \right). \quad (6)$$

with per-channel Huber thresholds  $\delta_i$  chosen by the user. The Huber penalty is

$$\psi_\delta(e) = \begin{cases} \frac{1}{2}e^2, & |e| \leq \delta, \\ \delta \left( |e| - \frac{1}{2}\delta \right), & |e| > \delta. \end{cases}$$

Here  $\lambda$  scales the regularization, and  $\{\delta_i\}$  set the allowable deviation range for symmetry. This soft symmetry prior improves the practical identifiability of the two flaperon columns and limits leakage of wing-sensor correlations into control-effectiveness terms, while remaining faithful to conditions where symmetry does not hold.

### C. Control Surface Optimization

We compute the control-surface deflections that realize a desired wrench at each discrete time step  $t \in \mathbb{N}$ . Let  $o_t$  denote the observation vector at time  $t$ ,  $u_t \in \mathbb{R}^4$  the control input (left flaperon, right flaperon, elevator, rudder), and  $y_t \in \mathbb{R}^6$  the desired wrench  $[F_x, F_y, F_z, T_x, T_y, T_z]^\top$ . Given the control-affine predictor  $y_t \approx A_\phi(o_t) + B_\phi(o_t)u_t$ , we select  $u_t$  by solving a convex, regularized least-squares problem that balances tracking, smoothness, and trim adherence. For compactness, we write  $A_t := A_\phi(o_t)$  and  $B_t := B_\phi(o_t)$ . Let  $I$  be the  $4 \times 4$  identity;  $u_{t-1}$  the previous command; and  $u_0$  a fixed actuator-neutral trim. With nonnegative weights  $\lambda_0, \lambda_1 \geq 0$ , we solve

$$\min_{u_t} \underbrace{\|y_t - A_t - B_t u_t\|_2^2}_{\text{force tracking}} + \underbrace{\lambda_1 \|u_t - u_{t-1}\|_2^2}_{\text{smooth regularization}} + \underbrace{\lambda_0 \|u_t - u_0\|_2^2}_{\text{damping term}}. \quad (7)$$

*Force tracking* enforces consistency between the predicted wrench  $A_t + B_t u_t$  and the target  $y_t$ . *Smooth regularization* penalizes departures from  $u_{t-1}$  to promote temporal smoothness and avoid abrupt actuator changes. *Damping* biases

the command toward the neutral trim  $u_0$ , reducing magnitude, helping prevent saturation, and improving numerical conditioning. Problem (7) is quadratic and strictly convex when  $\lambda_0 + \lambda_1 > 0$ , thus admitting a closed form. Setting the gradient to zero yields

$$(B_t^\top B_t + (\lambda_0 + \lambda_1)I)u_t = B_t^\top (y_t - A_t) + \lambda_1 u_{t-1} + \lambda_0 u_0. \quad (8)$$

Define

$$Q_t = 2(B_t^\top B_t + (\lambda_0 + \lambda_1)I),$$

$$c_t = 2(B_t^\top (y_t - A_t) + \lambda_1 u_{t-1} + \lambda_0 u_0),$$

and compute the optimal command as the unique solution to  $Q_t u_t^* = c_t$ . Positive  $(\lambda_0, \lambda_1)$  render  $Q_t$  positive definite and trade off tracking fidelity against command smoothness and damping, typically yielding more stable, well-conditioned inputs in closed-loop operation.

## V. EXPERIMENTS

We conduct all experiments in a wind tunnel with a cross section of 1.3 m tall and 1.8 m wide. All learning components are implemented in Python using the PyTorch library [38]. We trained and validated the Pitot tube calibration model on a grid of airspeeds  $V_a \in \{8, 10, 12\}$  m/s, angles of attack  $\alpha \in \{-10^\circ, -5^\circ, 0^\circ, 5^\circ, 10^\circ\}$ , and sideslip angles  $\beta \in \{-10^\circ, -5^\circ, 0^\circ, 5^\circ, 10^\circ\}$ . For validation, we additionally actuated the upstream gust generator to introduce unsteady disturbances. The results are shown in Fig. 4. Both probes recover the flow features  $(V_a, \alpha, \beta)$  with good accuracy, though errors remain; in particular, probe 0 exhibits noticeably larger errors when the gust generator is active and positioned close to the test article. This highlights the fact our gust generator produces considerable amount of turbulence to our model airplane.

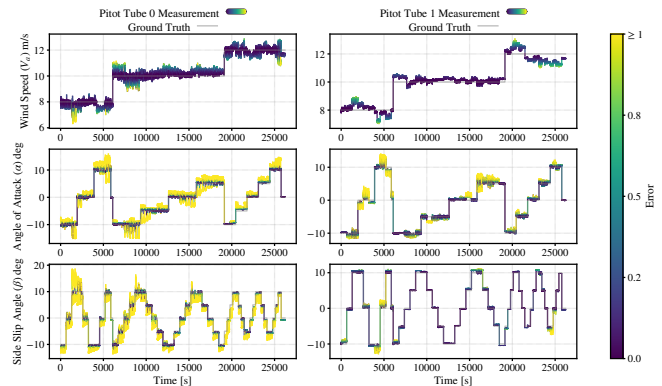


Fig. 4. **Estimated vs. ground-truth wind features.** Columns: Pitot tube 0 (left) and Pitot tube 1 (right). Rows: airspeed  $V_a$ , angle of attack  $\alpha$ , and sideslip angle  $\beta$ . The gust is positioned near probe 0.

Next, we train force models on the collected dataset and evaluate them using baselines and ablations. Our proposed model, *Affine Sym*, is a control-affine network  $y = A_\phi(o) + B_\phi(o)u$  with a soft left–right symmetry regularizer on the flaperon columns of  $B_\phi(o)$ . The variant *Affine* removes the symmetry regularization while retaining the control-affine

structure. To assess the contribution of wing-sensor inputs, *Affine w/o WS* excludes wing-pressure measurements from the observation vector. As non-affine baselines, *Unstructured* denotes a standard MLP mapping  $(o, u) \mapsto y$ , and *Unstructured w/o WS* further removes wing pressure inputs. Throughout, “WS” refers to wing sensor pressures.

We first assess the contribution of wing-pressure sensors to forces and moments learning. At  $V_a = 10$  m/s, Table I reports root-mean-square errors (RMSE) for the six-dimensional wrench—forces (N) and moments (N·m). Due to the page limit, we report a single RMSE over the concatenated 6D wrench (forces in N, torques in N·m) for all tables in this paper. This mixes units and should be read as a comparative aggregate. Incorporating wing-pressure measurements into the network’s input reduces error by 24.6%–29.5% relative to the corresponding models without wing-sensor inputs. Based on experimental observations, the wing-mounted pressure sensors are located close to the upstream gust generator. Their readings, particularly from the leading-edge sensors  $ps_0$  and  $ps_4$ , vary systematically with the gust angle. Including these wing-pressure signals as inputs therefore provides local context about gust-induced unsteady flow, improving calibration and estimation under disturbed conditions.

TABLE I  
ESTIMATION ERRORS (N/N·m) UNDER 10 m/s WIND

Affine	Affine w/o WS	Unstructured	Unstructured w/o WS
0.450	0.597	<b>0.442</b>	0.627

However, adding wing-pressure readings without additional structure introduces confounding. Because these signals respond directly to the right-flaperon deflection, supplying them unregularized creates strong endogeneity between the observation channels and the control input. The network can then explain flaperon-induced changes through the wing pressure inputs rather than the right-flaperon control input, so the estimated control-effectiveness matrix  $B_\phi(o)$  assigns insufficient sensitivity to the right-flaperon column and fails to capture the true control–force relationship. This misattribution is evident in the right plot of Fig. 5, where the right-flaperon column appears inconsistent. As an illustration on the lift channel  $F_z$ , the estimated control-effectiveness entries in  $B_\phi$  are  $B_{F_z, \delta_{ra}} = -1.81$  (right flaperon) and  $B_{F_z, \delta_{la}} = 6.96$  (left flaperon), i.e., the left-flaperon magnitude is  $\approx 3.8\times$  that of the right, whereas in reality, the actuation of left and right flaperons should have the same aerodynamic impact on the lift force  $F_z$ . This pronounced imbalance indicates that, without regularization, the model misattributes flaperon-induced lift changes, yielding a  $B_\phi$  matrix that does not faithfully represent the true input–output mapping.

To mitigate this issue, we impose a soft left–right symmetry regularization on  $B_\phi(o)$ . The regularizer penalizes deviations of the left- and right-flaperon columns from the expected sign pattern implied by aerodynamic symmetry, while allowing bounded departures to accommodate genuine

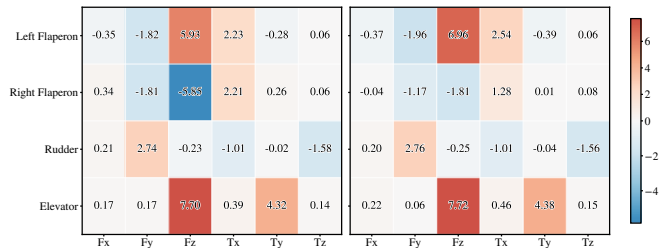


Fig. 5.  $B_\phi^T$  heatmaps. Left: model trained with symmetry regularization; right: without symmetry regularization. The regularized map is more aerodynamically plausible.

asymmetries induced by sideslip and the gust generator. As shown in the left panel of Fig. 5, the resulting estimate of  $B_\phi$  restores a consistent right-flaperon sensitivity.

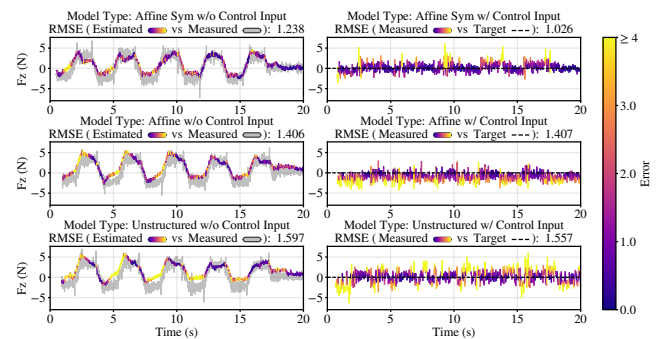


Fig. 6. Normal force  $F_z$  estimation under wind speed 14m/s. Columns: left, Measured vs. Estimated without control input; right, Measured vs. Target with control input. Rows: *Affine Sym*, *Affine*, and *Nonlinear* models. The results indicate that the *Affine Sym* model yields the lowest estimation and tracking errors.

TABLE II  
ESTIMATION ERRORS (N/N·m) UNDER DIFFERENT WIND SPEEDS

Wind Speed	Affine Sym	Affine	Unstructured
7 m/s	<b>0.521</b>	0.531	0.521
9 m/s	0.495	0.493	<b>0.484</b>
10 m/s	<b>0.522</b>	0.450	<b>0.442</b>
14 m/s	<b>0.592</b>	0.598	0.637

We evaluated the trained force–prediction models under an in-distribution condition ( $V_a = 10$  m/s) and under out-of-distribution wind speeds  $V_a \in \{7, 9, 14\}$  m/s to assess generalizability. Unlike the training settings, the validation experiments comprises two stages. **Stage I:** the angle of attack  $\alpha$ , sideslip  $\beta$ , and gust generator actuation are varied simultaneously and continuously to induce time-varying flow conditions. **Stage II:**  $\alpha$  and  $\beta$  are held at prescribed setpoints while the gust generator remains active, allowing us to check whether the aircraft can correctly estimate forces and torques and remain balanced under turbulence. These scenarios test whether the system can (i) accurately estimate the instantaneous wrench and (ii) generate control inputs that realize a specified target wrench under such

conditions. Table II reports the RMSE between predicted forces and measurements from the force transducer. The *Unstructured* baseline exhibits clear overfitting, with errors growing markedly at higher airspeed. Its RMSE increases by 44.1% from 10 to 14 m/s. In contrast, *Affine Sym* degrades more gracefully, with only an 11.8% increase over the same shift, indicating substantially better out-of-distribution robustness. We visualize the time series of the estimated

TABLE III  
TRACKING ERRORS ( $N/N \cdot m$ ) WITH CONTROL

Wind Speed	Affine Sym	Affine	Unstructured
7 m/s	<b>0.635 ± 0.13</b>	0.678 ± 0.13	0.748 ± 0.14
9 m/s	0.660 ± 0.06	<b>0.648 ± 0.09</b>	0.791 ± 0.01
10 m/s	<b>0.609 ± 0.07</b>	<b>0.598 ± 0.07</b>	<b>0.603 ± 0.07</b>
14 m/s	<b>1.287 ± 0.07</b>	1.309 ± 0.07	1.417 ± 0.02

normal force  $F_z$  for Stage-I in the left column of Fig. 6, with the per-subplot RMSE reported above each panel. The *Unstructured* baseline exhibits substantially larger errors than the *Affine* and *Affine Sym* models across conditions, consistent with overfitting. These results support the advantage of the control-affine formulation: by constraining the mapping to  $y = A_\phi(o) + B_\phi(o)u$ , it generalizes more reliably in the limited-data scenarios.

TABLE IV  
RMSSD FOR CONTROL INPUT UNDER 14 m/s WIND

Control Input	Affine Sym	Affine	Unstructured
Right Flaperon	<b>16.30 ± 1.65</b>	39.27 ± 3.60	57.78 ± 30.35
Average	<b>13.46 ± 1.36</b>	17.96 ± 2.40	49.54 ± 31.10

Finally, we evaluate whether the learned model, combined with our control-surface optimizer, can track and maintain a desired aerodynamic force across varying wind conditions. In this experiment, the angle of attack, sideslip, and gust generator actuation vary simultaneously and continuously, while the optimizer computes control-surface deflections  $\{\delta_{la}, \delta_{ra}, \delta_{el}, \delta_{ru}\}$  to realize a prescribed target wrench. Across all experiments, every model uses the same regularization weights  $\lambda_0$  and  $\lambda_1$ . Table III reports tracking RMSEs across three cases, namely extrapolated out-of-distribution airspeeds, an interpolated within-range airspeed at  $V_a = 9$  m/s that is held out, and an in-distribution airspeed at  $V_a = 10$  m/s. In the extrapolated case, *Affine Sym* attains the lowest error. At  $V_a = 9$  m/s, *Affine Sym* and *Affine* both outperform the unstructured model and are comparable, with *Affine* slightly lower. At  $V_a = 10$  m/s, differences among all methods are small.

We run three rounds of experiments at different airspeeds to guard against run-to-run variability and evaluate consistency. The analysis focuses on  $V_a = 14$  m/s, which lies in the extrapolative regime where generalization is most demanding. A representative Stage-I control sequence is shown in the right column of Fig. 6. In these plots, the *Affine* model exhibits a small but persistent bias relative to

the target, which is consistent with estimation error, while the *Nonlinear* baseline shows noticeably less stable behavior. On the  $F_z$  channel over this interval, *Affine Sym* reduces the tracking RMSE by 27% relative to *Affine* and by 34% relative to *Nonlinear*.

These results indicate that enforcing a control-affine structure improves generalization across conditions, while the symmetry prior further strengthens extrapolation beyond the training range. Although symmetry regularization improves identifiability, it is not universally optimal for fixed-wing systems. Aerodynamic and actuation asymmetries can arise from many reasons, so enforcing a strict mirror prior with a finite tolerance may mis-specify the model in some regions. This helps explain why *Affine Sym* does not always have the top performance in certain interpolated cases, where the approximate prior can conflict with the true physics. Under extrapolation, however, the symmetry prior provides a stronger inductive bias and yields the best generalization among the compared models. Beyond accuracy, the symmetry regularization also promotes steadier control, as reflected by lower root mean square of successive differences (RMSSD) values in Table IV at airspeed  $V_a = 14$  m/s. The control penalties  $\lambda_0$  and  $\lambda_1$  introduce tolerance to inaccuracies in the  $B_\phi$  and enable force tracking where the estimated forces are reliable, but they can induce oscillations in the control inputs, which RMSSD captures quantitatively.

## VI. CONCLUSION

This paper presents a sensing-to-control pipeline for fixed-wing UAVs that combines bio-inspired hardware, physics-informed learning, and convex control allocation. A narwhal-inspired standoff multi-hole probe increases sensor standoff to reduce near-body flow distortion, while sparse wing pressure sensors provide local context under unsteady conditions. A data-driven calibration estimates airspeed and flow angles from normalized pressures, and a control-affine force model with a soft left-right symmetry regularizer improves identifiability of state-dependent control effectiveness under partial observability. Desired aerodynamic wrenches are tracked via a small regularized least-squares allocator.

Wind-tunnel experiments across multiple airspeeds and gust conditions show improved estimation, robustness, and actuation. Adding wing pressures reduces force-estimation error by 25% to 30%. Under distribution shift, performance degrades by  $\sim 12\%$ , compared with 44% for an unstructured baseline. In closed loop, normal-force tracking error drops by 27% versus a plain affine model and by 34% versus a nonlinear baseline, with smoother inputs, supporting the benefit of standoff sensing with physics-informed learning.

Future work will incorporate temporal dynamics and broaden operating regimes. Observation histories can help infer turbulence and vortices, estimate gust location/strength/angle, and forecast dynamics under inconsistent flow. We will also integrate uncertainty-aware free-flight control by embedding observation and environment uncertainty into an MPC framework for stability and safety, study stall

in free flight, and refine the symmetry prior and sensor placement to improve robustness and generalization.

## ACKNOWLEDGMENT

We thank J. Humml for guidance on fabricating our pitot probes, R. Nemovi for help evaluating the Ultra Slow Stick as a payload-capable airframe, and M. Anderson for insightful input throughout the project.

## REFERENCES

- [1] X. Shi, P. Spieler, E. Tang, E.-S. Lupu, P. Tokumaru, and S.-J. Chung, "Adaptive nonlinear control of fixed-wing vtol with airflow vector sensing," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 5321–5327.
- [2] W. Shyy, H. Aono, C.-k. Kang, and H. Liu, *An introduction to flapping wing aerodynamics*. Cambridge University Press, 2013, vol. 37.
- [3] P. Windes, X. Fan, M. Bender, D. K. Tafti, and R. Müller, "A computational investigation of lift generation and power expenditure of pratt's roundleaf bat (*hipposideros pratti*) in forward flight," *PLoS one*, vol. 13, no. 11, p. e0207613, 2018.
- [4] X. Fan, A. Gehrke, and K. Breuer, "Wing twist and folding work in synergy to propel flapping wing animals and robots," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 13 915–13 921.
- [5] X. Fan, A. Bortoni, S. Hao, S. Swartz, and K. Breuer, "Upstroke wing clapping in bats and bat-inspired robots offers efficient lift generation," *Journal of the Royal Society Interface*, vol. 22, no. 223, p. 20240590, 2025.
- [6] F. B. L. Network, "Bionicflyingfox," [https://www.festo.com/us/en/e/about-festo/research-and-development/bionic-learning-network/bionic-flying-objects/bionicflyingfox-id\\_32755/](https://www.festo.com/us/en/e/about-festo/research-and-development/bionic-learning-network/bionic-flying-objects/bionicflyingfox-id_32755/), n.d., accessed: 2025-09-14.
- [7] A. Chen, B. Song, K. Liu, Z. Wang, D. Xue, and H. Qi, "Flapping-wing robot achieves bird-style self-takeoff by adopting reconfigurable mechanisms," *Science Advances*, vol. 11, no. 36, p. eadx0465, 2025.
- [8] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [9] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, ser. Springer Handbooks. Springer International Publishing, 2016. [Online]. Available: <https://books.google.com/books?id=RTvADAAAQBAJ>
- [10] L. Meier, D. Honegger, and M. Pollefeys, "Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6235–6240.
- [11] F. Xie, G. Shi, M. O'Connell, Y. Yue, and S.-J. Chung, "Hierarchical meta-learning-based adaptive controller," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 18 309–10 315.
- [12] M. O'Connell, G. Shi, X. Shi, K. Azizadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural-fly enables rapid learning for agile flight in strong winds," *Science Robotics*, vol. 7, no. 66, p. eabm6597, 2022. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.abm6597>
- [13] R. Beard and T. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012. [Online]. Available: <https://books.google.com/books?id=YqQtjhPUaNEC>
- [14] M. T. Nweeia, F. C. Eichmiller, P. V. Hauschka, G. A. Donahue, J. R. Orr, S. H. Ferguson, C. A. Watt, J. G. Mead, C. W. Potter, R. Dietz *et al.*, "Sensory ability in the narwhal tooth organ system," *The Anatomical Record*, vol. 297, no. 4, pp. 599–617, 2014.
- [15] K. T. Wood, S. Araujo-Estrada, T. Richardson, and S. Windsor, "Distributed pressure sensing-based flight control for small fixed-wing unmanned aerial systems," *Journal of Aircraft*, vol. 56, no. 5, pp. 1951–1960, 2019. [Online]. Available: <https://doi.org/10.2514/1.C035416>
- [16] J. John D. Anderson, *Fundamentals of Aerodynamics*, 6th ed. New York, NY: McGraw-Hill Education, 2016.
- [17] A. F. Handbook, "Airplane flying handbook," *UPDATE*, 2025.
- [18] R. C. Nelson, *Flight Stability and Automatic Control*, 2nd ed. Boston, MA: WCB/McGraw-Hill, 1998.
- [19] P. Chen, G. Zhang, J. Li, Z. Chang, and Q. Yan, "Path-following control of small fixed-wing uavs under wind disturbance," *Drones*, vol. 7, no. 4, 2023. [Online]. Available: <https://www.mdpi.com/2504-446X/7/4/253>
- [20] E. Böhn, E. M. Coates, D. Reinhardt, and T. A. Johansen, "Data-efficient deep reinforcement learning for attitude control of fixed-wing uavs: Field experiments," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 3, pp. 3168–3180, 2024.
- [21] D. Olivares, P. Fournier, P. Vasishta, and J. Marzat, "Model-free versus model-based reinforcement learning for fixed-wing uav attitude control under varying wind conditions," 2024. [Online]. Available: <https://arxiv.org/abs/2409.17896>
- [22] D. J. Richter, R. A. Calix, and K. Kim, "A review of reinforcement learning for fixed-wing aircraft control tasks," *IEEE Access*, vol. 12, pp. 103 026–103 048, 2024.
- [23] E. S. Lupu, F. Xie, J. A. Preiss, J. Alindogan, M. Anderson, and S.-J. Chung, "MAGIC<sup>VFM</sup>: meta-learning adaptation for ground interaction control with visual foundation models," *IEEE Transactions on Robotics*, vol. 41, pp. 180–199, 2025.
- [24] T. X. Nghiem, J. Drgona, C. N. Jones, Z. Nagy, R. Schwan, B. Dey, A. Chakrabarty, S. D. Cairano, J. A. Paulson, A. Carron, M. N. Zeilinger, W. Shaw-Cortez, and D. L. Vrabie, "Physics-informed machine learning for modeling and control of dynamical systems," in *American Control Conference*, 2023, pp. 3735–3750.
- [25] F. Djeumou, C. Neary, E. Goubault, S. Putot, and U. Topcu, "Neural networks with physics-informed architectures and constraints for dynamical systems modeling," in *Learning for Dynamics and Control Conference*, 2022, pp. 263–277.
- [26] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics*, vol. 378, pp. 686–707, 2019.
- [27] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian neural networks," in *Advances in Neural Information Processing Systems*, 2019, pp. 15 353–15 363.
- [28] M. D. Cranmer, S. Greydanus, S. Hoyer, P. W. Battaglia, D. N. Spergel, and S. Ho, "Lagrangian neural networks," *arXiv preprint arXiv: 2003.04630*, 2020.
- [29] T. Duong, A. Altawaitan, J. Stanley, and N. Atanasov, "Port-hamiltonian neural ODE networks on lie groups for robot dynamics learning and control," *IEEE Transactions on Robotics*, vol. 40, pp. 3695–3715, 2024.
- [30] A. Sanchez-Gonzalez, N. Heess, J. T. Springenberg, J. Merel, M. A. Riedmiller, R. Hadsell, and P. W. Battaglia, "Graph networks as learnable physics engines for inference and control," in *International Conference on Machine Learning*, 2018, pp. 4467–4476.
- [31] F. Xie, S. Wei, Y. Song, Y. Yue, and L. Gan, "Morphological-symmetry-equivariant heterogeneous graph neural network for robotic dynamics learning," *Proceedings of the 7th Annual Learning for Dynamics & Control Conference*, 2024.
- [32] D. Butterfield, S. S. Garimella, N.-J. Cheng, and L. Gan, "MI-HGNN: Morphology-informed heterogeneous graph neural network for legged robot contact perception," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 10 110–10 116.
- [33] D. Birch, S. Shaw-Ward, and A. Titchmarsh, "Calibration and use of n-hole velocity probes," *AIAA Journal: devoted to aerospace research and development*, vol. 53, no. 2, pp. 336 – 346, 2014.
- [34] A. Ghosh, D. M. Birch, and O. Marxen, "Neural-network-based sensor data fusion for multi-hole fluid velocity probes," *IEEE Sensors Journal*, vol. 20, no. 10, pp. 5398–5405, 2020.
- [35] B. Stevens, F. Lewis, and E. Johnson, *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*. Wiley, 2015. [Online]. Available: <https://books.google.com/books?id=N4abCgAAQBAJ>
- [36] B. Etkin and L. Reid, *Dynamics of Flight: Stability and Control*. Wiley, 1995. [Online]. Available: <https://books.google.com/books?id=9pWbEAAAQBAJ>
- [37] D. O. Apraetz, G. Turrisi, V. Kostic, M. Martin, A. Agudo, F. Moreno-Noguer, M. Pontil, C. Semini, and C. Mastalli, "Morphological symmetries in robotics," *The International Journal of Robotics Research*, vol. 44, no. 10-11, pp. 1743–1766, 2025.
- [38] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.