Viability-Guided Sim-to-Real Transfer for a Small Fixed-Wing Glider in Uncertain Indoor Updrafts

by

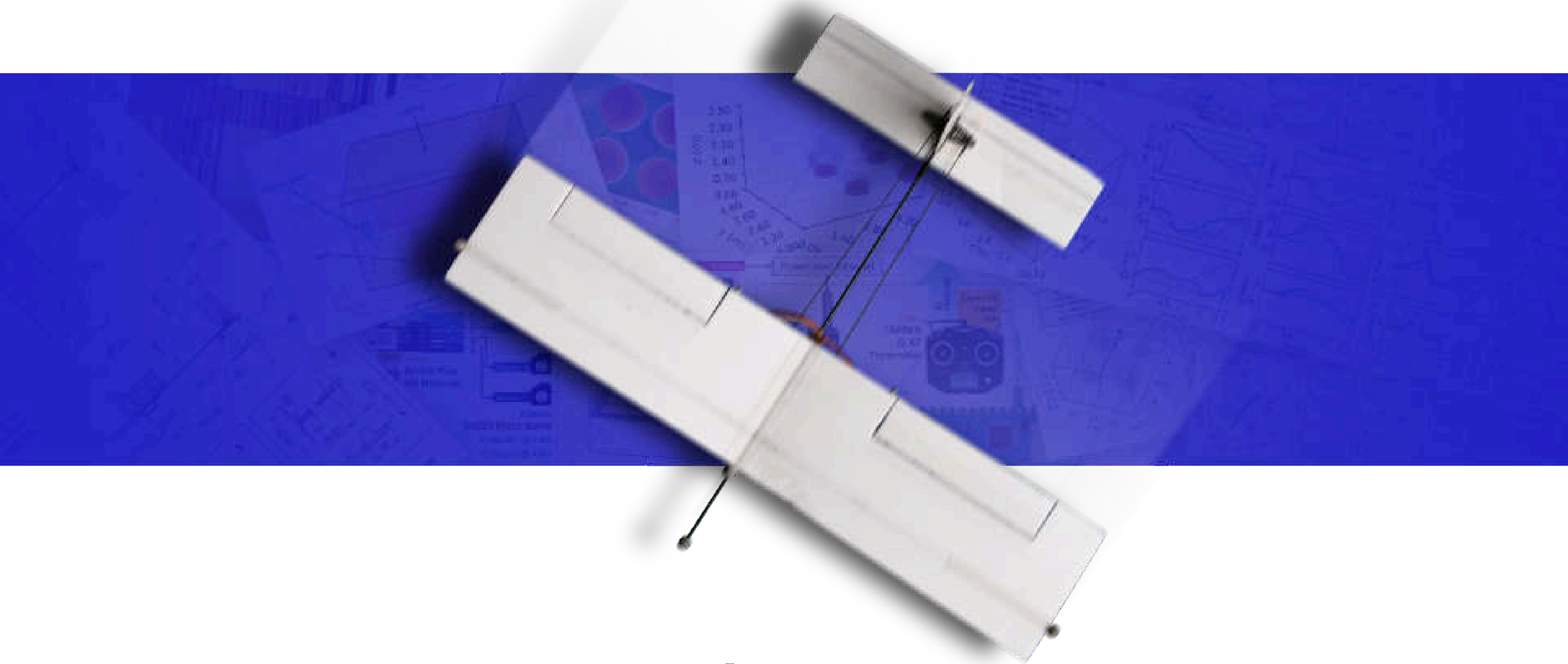
Hanchen Li

A thesis submitted to the Department of Aeronautics
in partial fulfilment of the requirements for the degree of

Master of Engineering (MEng) in Aeronautical Engineering

at

Imperial College London



June 2026

Author .... Hanchen Li
Candidate Number: 02229881
Department of Aeronautics, Imperial College London

Supervisor .... Dr Urban Fasel
Associate Professor in Data-Driven Aerospace Engineering
Department of Aeronautics, Imperial College London

Second Marker .... Dr Rhea P. Liem
Reader in Sustainable Aviation
Department of Aeronautics, Imperial College London

Abstract

Motivated by low-altitude flight near terrain, structures, or other flow-shaping boundaries, where local lift can extend endurance but can also remove recovery margin when encountered from an unfavourable state, this thesis investigates whether a small fixed-wing glider controller developed in simulation can transfer to repeated real flights through uncertain local updrafts. To study this problem repeatably, the thesis develops an indoor fixed-wing sim-to-real workflow in Imperial College London's Brahmal Vasudevan Multi Terrain Aerial Robotics Arena, combining Vicon motion capture state feedback, a measured command path and latency model, a manufactured glider, a safety-bounded flight volume, and fan-generated updrafts represented by measured but imperfect surrogates.

The controller is formulated as a viability-guided manoeuvre primitive selection. Instead of tracking a long planned trajectory through an uncertain flow field, the glider repeatedly selects short stabilised manoeuvre primitives every $0.10\,\mathrm{s}$ in flight. Each primitive is generated, replayed, and validated offline so that online selection can use explicit evidence about entry compatibility, continuation probability, hard failure risk, safety margin, lift exposure, and energy change. The resulting dense primitive library is then compressed into executable tiers, exposing the trade-off between manoeuvre coverage and computation speed. A bounded spatial lift-memory mechanism is also tested to examine whether repeated launches can effectively explore the lift field and improve updraft exploitation.

Simulation results show that the approach is computationally practical and physically constrained by launch energy. The final mission simulations contain 36000 runs and 384795 executed $0.10\,\mathrm{s}$ primitive segments. For feasible launch speeds, approximately above $5.0\,\mathrm{m\,s^{-1}}$, the balanced compressed library preserves mission performance of the raw dense library while reducing the mean evaluated set from 110.1 to 7.0 candidates and completing 85.2% of selector decisions within the primitive horizon. Spatial memory changes fewer than 10% of selections and gives little aggregate benefit in the arena.

Real flight testing confirms the main sim-to-real transfer claim under a more conservative practical launch speed range. Above the $6.0\,\mathrm{m\,s^{-1}}$, the still-air and single-fan cases transfer reliably, while four-fan flights expose the remaining reality gap in multi-source updraft modelling and glider's lateral response. The strongest evidence comes from random fan layouts that were not included in the held-out simulation validation during controller development: compared with open-loop flight, closed-loop success increases from 30.0% to 86.7% in the random three-fan layout and from 20.0% to 93.3% in the random four-fan layout. Replaying the flight results in the simulation environment shows that the simulator does not reproduce every trajectory detail, yet the frozen controller remains viable in flight. Therefore, the thesis demonstrates sim-to-real transfer not as one successful trajectory, but as a traceable evidence chain from modelling and validation to repeated real flight survival through uncertain indoor lift.

Acknowledgements

My four-year MEng at Imperial College London began at the pandemic‘s ebb and end with the first light of general intelligence. In between, the world has turned over more than once. I meant for these acknowledgements to close my chapter in the UK, yet I hesitate the moment I begin to write.

This thesis, in the form it now takes, is not the work of one person alone. It has been carried by the guidance of mentors and friends, and by the kindness of many whom I have never met. First, I would like to thank Dr Urban Fasel for his dedicated supervision over the past year. His insights and advice throughout this work have benefited me immensely. I would also like to thank my second marker, Dr Rhea P. Liem, for her valuable feedback and suggestions. I am grateful to Yao-Cheng (Paul) Li and Adrian-Andrei Buda for their thoughtful, constructive comments in our weekly meetings; to Tian Lan and Krispin Broers for helping me find my footing in the experimental environment; and to Tianyu Zou, Junjie (Jacky) Zhang, and Yuzhe Dun for support in more ways than I can properly list here – I wish you all every success in your doctoral journeys. In addition, my sincere thanks go to Mark Thornton, whose work keeps the day-to-day operations of the Flight Arena running smoothly, and to my roommate, Xu (Michael) Miao, who provided extensive help with setting up the 3D printer.

I also want to thank those scattered across the internet's vast constellation who continue to uphold the spirit of openness, sharing, and collaboration. Tairan He's WhynotTV Podcast first drew me toward control, robotics, and machine intelligence; Peter Sharpe's AeroSandbox showed me the elegance of optimisation-driven design; the countless authors and maintainers of open-source toolboxes in Python and MATLAB kept me from reinventing the wheel in the long trenches of low-level implementation.

My thanks also go to the developers of large language models. Although such systems are not always fully open, their value is undeniable, and helped make up for many of my limitations. Without the assistance of generative AI, I could hardly have managed, within eight months, to carry coursework, examinations, and research at once, and still push this project forward to where it stands today.

I am grateful, too, to those who have been my anchor through the storms of my mind. To my family: if education can be called an investment, you placed your largest stake in me, and even now, I feel my return has been nowhere near enough to repay that faith. To my personal tutor, Professor Robert Hewson: when I first entered university, uncertain and overwhelmed, you listened patiently to my uncertainties and anxieties; over the four years that followed, you repeatedly helped me recalibrate the direction. And to every friend who, whether at campus, in the paddock, on the Internet, or within the game world, has accepted me, put trust in me, and stayed by my side throughout the journey.

Finally, I would like to thank you, the reader who has opened this thesis. No one knows where the tide of this age will carry us, but whenever I feel lost, I always think back to the opening remarks from the first lecture of the freshman year *Introduction to Aerospace* course:

Remember, the sky's the limit.

致谢

The following is a Chinese translation of the preceding English acknowledgements.

帝国理工本硕四年，始于疫情之末，终于通用智能纪元之初，沧桑倏变、世事翻覆，想以本段致谢为自己的英国求学生涯画下句点，却提笔踌躇。

此文终得定稿，非一人之力可成，赖诸多师友襄助，亦承素昧平生者之惠泽。首先，我想致谢 Urban Fasel 副教授在过去一年作为项目导师对我的悉心指导，所赐见解与建议使我受益良多。其次，我想感谢第二评阅人 Rhea P. Liem 教授提出的细致评述与宝贵意见。我还想感谢 Yao-Cheng（Paul）Li 和 Adrian-Andrei Buda 在每周会议中不吝点拨；感谢兰天和 Krispin Broers 协助我熟悉实验环境；感谢邹天宇、张竣杰（Jacky）和敦渝哲在诸多方面给予我的支持——祝诸位博士生涯诸事顺遂，成果丰硕。此外，我想感谢 Mark Thornton，Flight Arena 的日常运转因你而井然有序；也感谢我的舍友缪续（Michael），在调试 3D 打印机的过程中为我提供了莫大的帮助。

我想感谢那些散落于互联网星海，仍在坚守开放、分享与协作精神的人们：何泰然的 WhynotTV 播客将我引入了控制、自主机器与智能系统的世界；Peter Sharpe 的 AeroSandbox 助我领略了优化设计的精妙；更有无数 Python 与 MATLAB 开源工具箱的制作者使我不必深陷重造轮子的苦海。

我想感谢那些大语言模型的开发者们，虽受现实所限，未尽开源，然其功其益，足以补我短拙。若无生成式人工智能相辅，八月之间课业、考试、科研三线并举，恐难至此。

我想感谢那些在心海风浪里予我定锚的人们。我的家人——我的教育或许是你们做过最大的一笔风险投资，而我至今仍愧于回报之微。我的个人导师 Robert Hewson 教授——在我初入大学彷徨之际，愿意倾听我无处安放的困惑与不安，此后四年又屡屡替我校准方向。还有每一位在校园内、围场里、互联网上或游戏世界中接纳我、信任我、并一路伴我前行的朋友。

最后，我想感谢正在阅读这篇论文的你。没有人知道时代的浪潮将把我们拍向何方，但每当我感到迷茫时，总会想起大一《航空航天概论》第一堂课的开场白：

谨记，天空无垠。

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

Latin Letters

Symbol	Units	Description	Formula
a_f		updraft strength scale for fan f	
a_i		finite aspect ratio lift slope of aerodynamic strip i	
$a_{0,f}$	$\mathrm{m\,s^{-1}}$	zeroth Fourier coefficient of the annular fan strength	
$a_{n,f}$	$\mathrm{m\,s^{-1}}$	cosine Fourier coefficient of order n for fan f	
$b_{n,f}$	$\mathrm{m\,s^{-1}}$	sine Fourier coefficient of order n for fan f	
b	m	canted wing span	
b_y	m	projected wing span	$b\cos\Gamma$
b_H	m	horizontal tail span	
h_V	m	vertical tail height	
$\mathcal{B}$		bound set for the glider sizing variables	
$\mathcal{B}_0$		initial empty spatial memory state for a new case	
$B_j(\mathbf{x})$		bounded local flow belief correction	
c	m	wing chord	
c_i	m	chord of aerodynamic strip i	
c_k		discrete state class assigned at decision step k	
$C_{D,i}$		drag coefficient of aerodynamic strip i	
$C_{D0,i}$		profile drag coefficient of aerodynamic strip i	
$C_{L,i}$		lift coefficient of aerodynamic strip i	
C_{entry}		entry-class library-compression coverage term	
C_{speed}		speed-bin library-compression coverage term	
C_{env}		environment library-compression coverage term	
$C_{\text{redundant}}$		library-compression redundancy penalty	
C_k		glider sizing feasibility constraint	
$\mathbf{C}_{wb}(\boldsymbol{\eta})$		body frame to arena frame attitude matrix	
$\mathcal{C}_{\text{exit}}$		allowed transition-validation exit-class set	
$\mathcal{C}_{\text{flight}}$		frozen controller stack	
$\mathbf{d}_{i,b}$		body-axis drag direction of aerodynamic strip i	
$d_j^{\text{safe}}(\mathbf{x})$		predicted minimum safety margin for candidate j	

d_q		distance weight for spatial lift memory cell q	
$\mathbf{D}_z$		fixed conversion between the arena frame z-up convention and body-axis convention	$\mathrm{diag}(1,\,1,\,-1)$
$\mathcal{D}$		dataset used for fitting, calibration, or validation	
$\mathcal{D}_{\text{train}}$		system-identification training dataset	
$\mathcal{D}_{\text{hold}}$		system-identification held-out dataset	
$\mathbf{e}$		environment instance for one finite-horizon simulation	
$\mathbf{e}_n$		local linearisation error state at discrete step n	$\mathbf{x}_n - \mathbf{x}_j^{\text{ref}}$
e_i	$\mathrm{m\,s^{-1}}$	pointwise updraft model error	
e_i		efficiency factor of aerodynamic strip i	
E_j		transition evidence stored for manoeuvre candidate j	
F		number of fans for one updraft configuration	
f_i		nearest-fan index assigned to measurement sample i	
$f_{\boldsymbol{\theta}_{\text{aero}}}$		calibrated glider model	
f_{scale}		annular-Gaussian robust loss scale	
$\mathbf{F}_b$	N	total body-axis aerodynamic force	
$\mathbf{F}_{i,b}$	N	body-axis aerodynamic force from strip i	
$\mathbf{F}_{\text{res}}$	N	compact residual correction to total body-axis force	
g	$\mathrm{m\,s^{-2}}$	gravitational acceleration magnitude	
$\mathbf{g}_w$	$\mathrm{m\,s^{-2}}$	arena-frame gravitational acceleration vector	$[0,\,0,\,-g]^\top$
$\hat{G}_j$	m	estimated specific energy gain for manoeuvre j	
H_{res}	m	terminal specific energy reserve above the floor reference	$H(\mathbf{x}_f) - H_{\text{floor}}$
H_{floor}	m	specific energy reference at the floor boundary	
ΔH_ℓ^+	m	positive useful updraft specific energy gain	
$\mathbf{I}_b$	$\mathrm{kg\,m^2}$	body-axis inertia matrix of the manufactured glider	
$\mathcal{I}$		frozen controller interface tuple	
$J(z)$		glider sizing objective function	
$\mathcal{J}(\boldsymbol{\theta}_{\text{aero}})$		regularised objective for aerodynamic calibration	
J_{sim}		post-run simulation score for complete mission rollout	
J_{fail}		failure penalty term	
J_{route}		route outcome reward	
$J_{H,f}$		terminal specific energy reserve reward	

j		manoeuvre candidate index
$j^\star$		selected manoeuvre candidate
j_g^{medoid}		representative medoid candidate for group g
$k(\cdot,\cdot)$		Gaussian process covariance kernel
$\mathbf{K}$		kernel covariance matrix
$\mathbf{K}_y$		effective training covariance matrix
$\mathbf{K}_j$		local feedback gain for manoeuvre candidate j
$\mathcal{L}(c)$		candidate list retrieved for state class c
$\mathcal{L}_B$		compressed manoeuvre library with budget B
$\mathcal{L}_\tau$		set of candidates supported by current timing model
l_b	m	boom length
l_t	m	tail arm
$\boldsymbol{\ell}_{i,b}$		body-axis lift direction of aerodynamic strip i
M		number of measurement samples
m	kg	glider mass
$M_j(\mathbf{x})$		governor-score mission progress term
$\mathbf{M}_b$	N m	total body-axis aerodynamic moment
$\mathbf{M}_{i,b}$	N m	body-axis aerodynamic moment from strip i
$\mathbf{M}_{\text{res}}$	N m	residual correction to total body-axis moment
$\mathcal{N}_{0.20}(\mathbf{p})$		neighbourhood of stored spatial memory cells near query position $\mathbf{p}$
N_C		number of sizing constraints
N_f		Fourier order used for fan f
$N_j(c)$		number of simulated flight samples for candidate j from entry class c
N_s		number of aerodynamic strips for the panelwise model
n		number of samples, throws, or trials
n_{match}		number of accepted matched transition events
$p_{\mathcal{E}}(\mathbf{e})$		randomised environment distribution
$\hat{p}_j^{\text{hard}}(c)$		estimated hard failure probability for candidate j from class c
P_W		wing-tip deflection penalty
P_H		horizontal-tail tip deflection penalty

P_{95}		95th percentile of an empirical distribution	
P_{99}		99th percentile of an empirical distribution	
$\mathbf{p}$	m	arena-frame glider position	$[x, y, z]^\top$
$\mathbf{p}_i$	m	arena-frame centre position of aerodynamic strip i	
$\mathbf{p}_{V,k}$	m	raw Vicon position measurement at sample k	
$\mathbf{p}^\star$	m	known calibration point used for Vicon alignment	
q_i	Pa	dynamic pressure at aerodynamic strip i	
q_s	Pa	dynamic pressure at the sizing speed	
$\mathbf{Q}_j$		stage state weighting matrix for candidate j	
$\mathbf{Q}_{f,j}$		terminal state weighting matrix for candidate j	
r_f	m	horizontal distance from a point to fan f	
r_i	m	distance from sample i to its assigned fan	
$r_{\text{ring},f}$	m	annular radius of fan f	
$\mathbf{r}_{i,b}$	m	body-frame location of aerodynamic strip i relative to the glider centre of gravity	
R_j^{regime}		poorly calibrated angle of attack regime penalty	
$\mathbf{R}_j$		command weighting matrix for candidate j	
S	m^2	wing planform area	
S_i	m^2	area of aerodynamic strip i	
$S_j(\mathbf{x})$		governor score for admissible candidate j	
$\mathbf{s}_{i,b}$		body-axis span direction of aerodynamic strip i	
T_ℓ	s	useful local-lift dwell time	
$\hat{T}_{\ell,j}$	s	estimated useful lift dwell time for candidate j	
T_j	s	finite execution horizon of manoeuvre candidate j	
$T_{V,\ell}$	s	Vicon capture-to-report latency	
T_f	s	low-pass state-filter delay	
T_x	s	state-feedback delay	
$T_{\max}$	s	maximum allowed runtime threshold	
t_0	s	initial time of a mission segment	
t_f	s	terminal time of a mission segment	
t_{10}	s	control surface response time to 10 percent motion	
t_{50}	s	control surface response time to 50 percent motion	

t_{90}	s	control surface response time to 90 percent motion
$\mathcal{T}_\tau$		timing and latency model used by the controller
$\mathbf{T}(\boldsymbol{\eta})$		Euler angular rate kinematic transformation matrix
u_b	$\mathrm{m\,s^{-1}}$	body-axis forward velocity component
v_b	$\mathrm{m\,s^{-1}}$	body-axis starboard velocity component
w_b	$\mathrm{m\,s^{-1}}$	body-axis downward velocity component
$\mathbf{u}^{\mathrm{cmd}}$		normalised aggregate controller output before executable command mapping
u_i		robust updraft-fitting normalised residual
$\mathcal{U}$		allowable executable command set
$\mathbf{V}_{i,b}$	$\mathrm{m\,s^{-1}}$	body-axis relative airflow at strip i
$\mathbf{V}_{i,b}^{\perp}$	$\mathrm{m\,s^{-1}}$	body-axis relative airflow at strip i after removing the spanwise component
$\mathcal{V}$		validated manoeuvre candidate set
V_s	$\mathrm{m\,s^{-1}}$	representative sizing speed
V_{sink}	$\mathrm{m\,s^{-1}}$	sink rate
$\mathbf{W}$	$\mathrm{m\,s^{-1}}$	arena frame wind velocity
w	$\mathrm{m\,s^{-1}}$	vertical-flow velocity
$w_{\mathbf{e}}$	$\mathrm{m\,s^{-1}}$	vertical-flow field for environment instance $\mathbf{e}$
w_{HAG}	$\mathrm{m\,s^{-1}}$	harmonic annular Gaussian mean updraft surrogate
$w_{\mathrm{HAG-GP}}$	$\mathrm{m\,s^{-1}}$	residual annular Gaussian process updraft surrogate
w_{res}	$\mathrm{m\,s^{-1}}$	residual between measured and residual annular Gaussian process updraft surrogate vertical velocity
$\bar{w}$	$\mathrm{m\,s^{-1}}$	measured mean vertical velocity
z		local sizing decision vector
x	m	arena-frame longitudinal coordinate
y	m	arena-frame lateral coordinate
z	m	arena-frame vertical coordinate
$\mathbf{x}$		15-state vector used by the controller
$\mathbf{x}_0$		initial state of a flight segment
$\mathbf{x}_f$		terminal state of a flight segment
$\mathbf{x}_j^{\mathrm{ref}}$		reference state for manoeuvre candidate j
$\mathcal{X}_{\mathrm{safe}}$		admissible flight volume for controller evaluation

$\mathcal{X}_{\text{launch}}$		launch gate initial condition set
$\mathbf{y}$		system-identification output vector
$z_{\text{fan},k}$	m	height of measurement plane k above the fan outlet
$\mathbf{z}_j$		behaviour and evidence feature vector
$\mathcal{A}(\mathbf{x})$		admissible manoeuvre candidate set
$\mathcal{A}_{\text{shield}}(\mathbf{x})$		shielded admissible manoeuvre candidate set
$\mathbf{m}_i$		control effectiveness vector of aerodynamic strip i
$\mathbf{n}_{i,b}$		body-axis surface normal of aerodynamic strip i
$\mathcal{P}$		set of active manoeuvre families
$\bar{p}_{\text{hard}}$		hard-failure probability acceptance threshold
$\bar{P}_{\text{route}}$		route-probability acceptance threshold
$\mathit{Æ}_i$		aspect ratio of aerodynamic strip i
C_Y		side-force coefficient
C_l		rolling-moment coefficient
C_m		pitching-moment coefficient
C_n		yawing-moment coefficient
C_{mq}		pitch damping derivative with respect to nondimensional pitch rate
k_D		flat-plate strip-model post-stall drag gain
$\bar{q}$	Pa	dynamic pressure at the glider centre of gravity
R_L	N m	roll equation intermediate moment term
R_M	N m	pitch equation intermediate moment term
R_N	N m	yaw equation intermediate moment term
V	m s^{-1}	airspeed magnitude at the glider centre of gravity
$V_{s,i}$	m s^{-1}	spanwise component of local relative airflow at aerodynamic strip i
X_b	N	total body-axis axial force component
Y_b	N	total body-axis side force component
Z_b	N	total body-axis normal force component
L_b	N m	body-axis rolling moment component
M_b	N m	body-axis pitching moment component
N_b	N m	body-axis yawing moment component

$\mathcal{R}_h(\alpha)$		post-stall radial basis function centred at angle of attack α_h	
W_{att}		attached-flow residual blending weight	
W_{tr}		transition residual blending weight	
W_{post}		post-stall residual blending weight	
$X_{i,b}$	N	body-axis axial force component from aerodynamic strip i	
$Y_{i,b}$	N	body-axis side-force component from aerodynamic strip i	
$Z_{i,b}$	N	body-axis normal-force component from aerodynamic strip i	
$L_{i,b}$	N m	body-axis rolling moment component from aerodynamic strip i	
$M_{i,b}$	N m	body-axis pitching moment component from aerodynamic strip i	
$N_{i,b}$	N m	body-axis yawing moment component from aerodynamic strip i	
$X_{\text{LD,res}}$	N	longitudinal post-stall lift-drag residual axial force	
$Z_{\text{LD,res}}$	N	longitudinal post-stall lift-drag residual normal force	
C_L^{res}		post-stall lift residual coefficient	
C_D^{res}		post-stall drag residual coefficient	
$W_{x,i,b}$	$\mathrm{m\,s^{-1}}$	body-axis wind axial velocity component at aerodynamic strip i	
$W_{y,i,b}$	$\mathrm{m\,s^{-1}}$	body-axis wind lateral velocity component at aerodynamic strip i	
$W_{z,i,b}$	$\mathrm{m\,s^{-1}}$	body-axis wind vertical velocity component at aerodynamic strip i	

Greek Letters

Symbol	**Units**	**Description**	**Formula**
α_i	rad	local angle of attack of aerodynamic strip i	
$\alpha_{i,\text{eff}}$	rad	effective angle of attack including zero-lift offset and control incidence of aerodynamic strip i	
$\alpha_{0,i}$	rad	zero-lift angle of attack offset of aerodynamic strip i	
α_q		recency weight for spatial memory cell q	
$\mathit{Æ}$		aspect ratio	
β	rad	sideslip angle	

β		lateral residual variable	
$\boldsymbol{\delta}$	rad	realised control surface state vector	
$\boldsymbol{\delta}^{\mathrm{cmd}}$	rad	commanded surface target after command mapping	
δ_a	rad	aileron surface state	
δ_e	rad	elevator surface state	
δ_r	rad	rudder surface state	
$\delta_{\mathrm{ring},f}$	m	annular thickness parameter for fan f	
$\boldsymbol{\epsilon}_a$	rad	calibrated roll, pitch, and yaw alignment offset	
$\boldsymbol{\epsilon}_\ell$		observed lift memory residual vector	
ϵ_P		allowed continuation-probability reduction for the memory shield	
ϵ_m		allowed path-margin reduction for the memory shield	
$\boldsymbol{\eta}$	rad	attitude vector	$[\phi, \theta, \psi]^\top$
η		harmonic penalty exponent or local spanwise coordinate, according to context	
Γ	$^\circ$	wing dihedral angle	
$\Gamma(\cdot)$		acceptance predicate for held-out transition validation	
$\boldsymbol{\gamma}$		control effectiveness scale vector	
$\boldsymbol{\gamma}$		harmonic annular Gaussian updraft surrogate regularisation hyperparameter vector	
κ_f		spatial width scale for fan f	
κ_σ		inflation scale applied to the updraft fluctuation field	
λ_i		mapping from local surface deflection to strip incidence change of aerodynamic strip i	
λ_ℓ		relative weight on lift exploitation	
λ_{ridge}		harmonic-annular-Gaussian ridge regularisation weight	
λ_{cap}		harmonic-annular-Gaussian soft-cap regularisation weight	
ℓ_d		Gaussian process ARD length scale for feature dimension d	
μ_f		active fan mask for fan f	
μ_{res}	$\mathrm{m\,s^{-1}}$	Gaussian process predictive residual mean	
μ_*	$\mathrm{m\,s^{-1}}$	Gaussian process predictive mean at a query point	
$\boldsymbol{\nu}$		executable aggregate command vector	
ν_a		executable aggregate aileron command	

ν_e		executable aggregate elevator command
ν_r		executable aggregate rudder command
ρ	$\mathrm{kg\,m^{-3}}$	air density
$\rho(\cdot)$		robust loss function
ρ_j		manoeuvre-candidate primitive family label
σ		sample standard deviation
σ_i	$\mathrm{m\,s^{-1}}$	local fluctuation scale
σ_i		smooth stall activation of aerodynamic strip i
$\sigma_{\mathrm{HAG-GP}}$	$\mathrm{m\,s^{-1}}$	total fluctuation field of the residual annular Gaussian process updraft surrogate
σ_{res}	$\mathrm{m\,s^{-1}}$	Gaussian process predictive residual standard deviation
σ_c	$\mathrm{m\,s^{-1}}$	Gaussian process signal standard deviation
σ_w	$\mathrm{m\,s^{-1}}$	Gaussian process white-noise standard deviation
θ_f	rad	azimuth angle about fan f
$\boldsymbol{\theta}_{\mathrm{aero}}$		aerodynamic residual and control surface effectiveness parameter vector
$\boldsymbol{\theta}_m$		pitch moment residual parameter block
$\boldsymbol{\theta}_{\mathrm{lat}}$		lateral directional residual parameter block
$\boldsymbol{\theta}_\delta$		control surface effectiveness scale parameters
$\boldsymbol{\theta}_p$		glider property randomisation parameters
$\boldsymbol{\theta}_\tau$		timing and latency randomisation parameters
$\boldsymbol{\Theta}_g$		closed-loop governor tuning vector
$\boldsymbol{\Theta}_{\mathrm{ring}}$		stacked harmonic annular Gaussian updraft surrogate parameter vector over all fans
τ_δ	s	first-order control surface response time constant
τ_{roll}	s	estimated roll response time constant
ϕ	rad	roll angle
θ	rad	pitch angle
ψ	rad	yaw angle
$\boldsymbol{\phi}_i$		raw Gaussian process input feature vector
$\tilde{\boldsymbol{\phi}}_i$		standardised Gaussian process input feature vector
$\chi(\mathbf{x})$		state classification map
Π_j		complete manoeuvre candidate object

$\boldsymbol{\omega}_b$	$\mathrm{rad\,s^{-1}}$	body angular rate vector	$[p_b,\, q_b,\, r_b]^\top$
ω_f		inverse-distance blending weight	
Δ_I	$\mathrm{kg^2\,m^4}$	determinant-like inertia denominator for the retained I_{xz} rotational dynamics	
$\delta_{s,+}$	rad	positive physical deflection limit for control surface s	
$\delta_{s,-}$	rad	negative physical deflection limit for control surface s	
γ_a		aileron aerodynamic effectiveness scale factor	
γ_e		elevator aerodynamic effectiveness scale factor	
γ_r		rudder aerodynamic effectiveness scale factor	
δ_i	rad	local control surface deflection contribution at aerodynamic strip i	
α_h	rad	post-stall radial basis function angle of attack centre	
$\Delta\alpha_{\mathrm{RBF}}$	rad	post-stall radial basis function angle of attack width	

Superscripts

Symbol	**Description**
$(\cdot)^\star$	optimal, selected, calibrated, or reference value
$(\cdot)^{\mathrm{att}}$	attached flow quantity
$(\cdot)^{\mathrm{cmd}}$	commanded quantity
$(\cdot)^{\mathrm{cons}}$	conservative timing or uncertainty case
$(\cdot)^{\mathrm{cont}}$	continuation route probability or quantity
$(\cdot)^{\mathrm{flight}}$	quantity measured from flight data
$(\cdot)^{\mathrm{hard}}$	hard failure probability or count
$(\cdot)^{\mathrm{post}}$	post-stall quantity
$(\cdot)^{\mathrm{ref}}$	reference quantity for a local manoeuvre candidate
$(\cdot)^{\mathrm{sim}}$	quantity produced by model replay or simulation
$(\cdot)^{\mathrm{term}}$	terminal route probability or quantity
$(\cdot)^{\mathrm{tr}}$	transition regime quantity

Subscripts

Symbol	**Description**
$(\cdot)_{\mathrm{att}}$	attached flow quantity
$(\cdot)_b$	glider body-frame quantity

$(\cdot)_w$	arena-frame quantity
$(\cdot)_V$	raw Vicon measurement quantity
$(\cdot)_{CG}$	quantity evaluated at the centre of gravity
$(\cdot)_a$	aileron components
$(\cdot)_e$	elevator components
$(\cdot)_r$	rudder components
$(\cdot)_f$	fan-indexed quantity
$(\cdot)_i$	measurement sample or aerodynamic strip indexed quantity
$(\cdot)_j$	manoeuvre candidate indexed quantity
$(\cdot)_k$	discrete time sample or measurement plane indexed quantity
$(\cdot)_n$	finite-horizon simulated flight step
$(\cdot)_{\text{post}}$	post-stall quantity
$(\cdot)_s$	flight record, simulated flight, or sizing speed indexed quantity
$(\cdot)_{\text{launch}}$	launch gate quantity
$(\cdot)_{\text{safe}}$	admissible safety volume quantity
$(\cdot)_{\text{ring}}$	annular component of the updraft model
$(\cdot)_{\text{res}}$	residual correction or residual model quantity
$(\cdot)_{\text{tr}}$	transition regime quantity
$(\cdot)_\ell$	lift related quantity
$(\cdot)_\tau$	timing or latency related quantity
$(\cdot)_\delta$	control surface or surface effectiveness related quantity

Abbreviations

Symbol	**Description**
1D	one-dimensional
2D	two-dimensional
3D	three-dimensional
6-DoF	six degrees of freedom
AG	annular Gaussian
ARD	automatic relevance determination
CAD	computer-aided design
CG	centre of gravity

COTS	commercial off-the-shelf
DoF	degrees of freedom
GP	Gaussian process
HAG	harmonic annular Gaussian
HAG-GP	harmonic annular Gaussian with Gaussian process residual correction
IPOPT	Interior Point OPTimiser
LDV	laser Doppler velocimetry
LQR	linear quadratic regulator
MAC	mean aerodynamic chord
MDO	multidisciplinary optimisation
PCHIP	piecewise cubic Hermite interpolating polynomial
PIV	particle image velocimetry
PPM	pulse-position modulation
RBF	radial basis function
RC	radio control
RL	reinforcement learning
ROS	Robot Operating System
ROM	reduced-order modelling
SAE	sum of absolute errors
$SO(3)$	three-dimensional rotation group
UAV	unmanned aerial vehicle
WRMSE	weighted root mean square error
XPS	extruded polystyrene

1 Introduction

Consider a small fixed-wing unmanned aircraft assigned to low-altitude sensing in confined outdoor airspace. It may be mapping terrain or reconstructing a site in 3D [1], [2], monitoring vegetation and coastlines [3], surveying damage after a natural disaster [4], or inspecting infrastructure and terrain boundaries such as bridges, railways, ridgelines, cliffs, valleys, and forest edges [5]. Despite their different objectives, these missions share one flight control difficulty: the aircraft must collect useful data while remaining inside a narrow region shaped by terrain, sensing geometry, and safety limits.

In such places, the aircraft is flying along the boundary between information and risk. Buildings, rooflines, street canyons, bridge decks, slopes, cliffs, and uneven ground can shape the surrounding air into wakes, shear layers, gusts, and local updrafts [6], [7]. For a small fixed-wing vehicle, this flow can be useful and dangerous simultaneously. A lift region may help it reduce energy loss and extend endurance [8], [9]. However, when encountered from the wrong flight state, such a region may erode the recovery margin before the aircraft can return to a safe state.

This thesis studies that trade-off through a controlled indoor setting. A small glider is flown through uncertain lift field inside Imperial College London's Brahmal Vasudevan Multi Terrain Aerial Robotics Arena [1]. The arena is far smaller and cleaner than the outdoor environments that motivate the work, introducing the challenge of recovering the essential flight control difficulty while isolating its sources: state measurement and sensing, aircraft design limitations, command path latency, controller design, and the mismatch between model and reality during controller development.

Accordingly, the above discussion leads to two linked tasks for the present study:

1. Build the experimental workflow: a fixed-wing aircraft, an indoor representation of uncertain lift, an executable command path, and a safety-bounded flight volume in the Flight Arena.
2. Design a control algorithm, together with its development and validation method, for the mission enabled by the experimental workflow: flying a fixed-wing aircraft through the uncertain lift field despite the mismatch between the vehicle model, command latency, sensing feedback, and environment surrogate used during controller development and those encountered in real flight.

The remainder of the thesis follows the evidence chain shown in Fig. 1.1: the problem is motivated by related work, realised in the Flight Arena, modelled to form the simulation environment, tested through controller development, and finally judged using real flight and corresponding simulation replay.

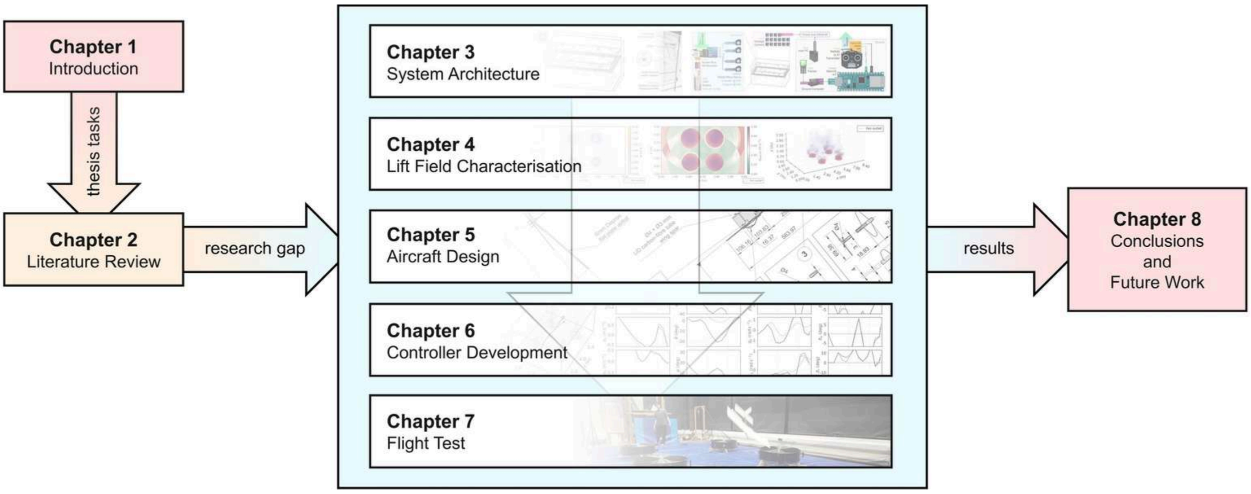


Fig. 1.1. Thesis roadmap from research gap to experimental results.

[1] To avoid repetition, this thesis uses the informal shortened form "Flight Arena" when referring to this facility.

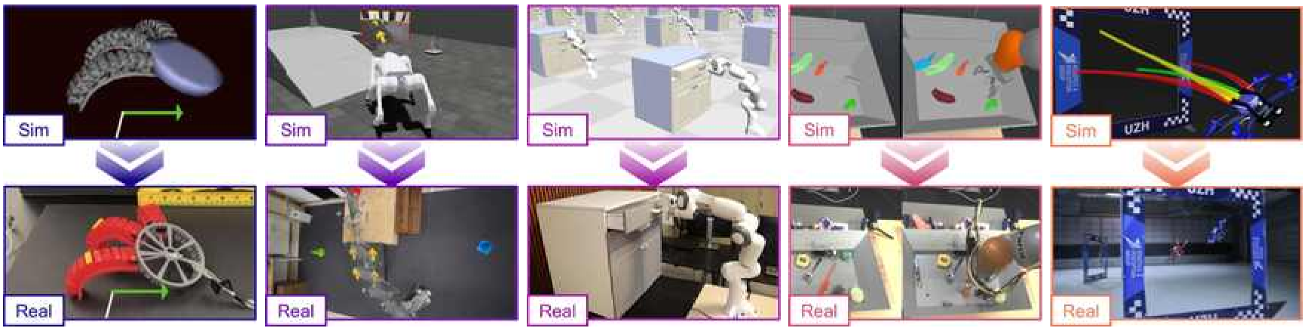
2 Background and Related Work

This chapter reviews the literature needed to formulate the thesis problem. It first defines sim-to-real transfer for fixed-wing flight under aerodynamic forcing, then narrows the discussion to controlled environment modelling, small glider design, manoeuvre primitive control, and the resulting research gap.

2.1 Sim-to-Real Transfer Under Aerodynamic Forcing

Sim-to-real[2] transfer refers to using models, controllers, or policies developed in simulation on physical hardware [10]. The main difficulty of sim-to-real is the reality gap: the mismatch between simulated and real dynamics, sensing, actuation, timing, and environmental interaction [11], [12]. In robotics, this gap has been addressed through system identification [13], [14], domain and dynamics randomisation [15], [16], [17], adaptive randomisation [18], [19], and real-to-sim-to-real loops [20], [21], [22].

Aerial sim-to-real work has extended these methods to both multirotor and fixed-wing vehicles. Studies of agile multirotor racing have used large-scale simulation, randomisation, and systematic testing to transfer learned perception and control policies to reality [23], [24]. Fixed-wing examples are fewer, but recent work shows that high-fidelity modelling and parameter randomisation can improve transfer in pitch control and attitude regulation [25], [26]. These studies support simulation as a useful development tool and show that success of transfer should be assessed using measurable validation tasks.



(a) locomotion [16]. **(b)** navigation [20]. **(c)** manipulation [21]. **(d)** perception [22]. **(e)** agile flight [23].

Fig. 2.1. Representative sim-to-real transfer examples in robotics and aerial autonomy.

When the surrounding flow is dynamically coupled to the vehicle, the reality gap should also include the aerodynamic forcing. Fixed-wing studies in wind and turbulence demonstrated that small aircraft are sensitive to non-uniform flow and gusts [27], [28]. Research on flow and dispersion in urban areas, including wind around buildings, further shows that structures can create shear layers, recirculation zones, and vertical motion [29], [30], [31]. Recent work on WindSeer also frames the need for sparse 3D wind prediction for small UAV operation in complex terrain [32]. For small fixed-wing vehicles, such spatially varying flow disturbances change the relative airflow at the lifting surfaces, affecting vehicle's energy state, flight path, control authority, and recoverability [33]. A simulator with an accurate aircraft model may still produce misleading behaviour if the flow field is poorly represented.

Beyond treating wind as a disturbance to be predicted or tolerated, autonomous soaring provides the closest fixed-wing precedent for exploiting useful lift as a partially observable environmental resource [8]. Early field-tested studies demonstrated that a small motorised sailplane could estimate and exploit thermal structure in flight [9], [34]. Subsequent work on atmospheric flow field modelling [35], biologically inspired soaring strategies [36], and thermalling control under limited flow observations [37] has advanced the representation, estimation, and use of uncertain lift fields. Recent orographic soaring experiments further show that fixed-wing MAVs can hover using local rising flow, while also revealing practical limitations in convergence to the lift region, circling behaviour, and roll stability [38], [39].

[2] Simulation-to-reality, sometimes also written as "sim2real" for brevity in paper titles, literature, and codebases.

Fig. 2.2. Representative simulation-to-flight progression in autonomous thermal soaring research, illustrated through Allen's NASA Dryden studies as an early flight-tested example [9], [34], [40].

The present thesis does not attempt to reproduce nature thermals, orographic lift, or urban canyons directly. Instead, it uses indoor updrafts to study a narrower sim-to-real problem: can feedback-stabilised glider manoeuvre primitives be selected and transferred consistently across repeated launches when vehicle dynamics, implementation timing, and aerodynamic forcing differ between simulation and reality?

2.2 Environment Design, Characterisation, and Updraft Surrogates

A controlled sim-to-real study requires an environment that is repeatable enough for comparison, measurable enough for model construction, and variable enough across space to expose transfer degradation. Outdoor thermals [35], [41], terrain-induced lift [38], [39], and wind around buildings [29], [31] are relevant to low altitude fixed-wing operation and endurance extension, but they are difficult to localise or reproduce [42]. Heat-driven indoor thermals are also poorly matched to the present facility because their length scales and heat input are impractical for the available volume [43], [44]. Hence, mechanically generated updrafts are used in this thesis as controlled laboratory forcing fields.

Within this controlled setting, measurements are mainly required to build a useful updraft surrogate from finite samples. They must capture the dominant vertical velocity structure, resolve variations over the glider scale, and provide local fluctuation estimates [45]. Particle image velocimetry (PIV) [46] and laser Doppler velocimetry (LDV) [47] offer high-fidelity flow information, but their requirements for optical access, seeding, illumination, and setup effort can limit coverage in indoor test environments [48]. Pointwise anemometry with spatial scanning provides a lower-dimensional flow characterisation [49], but can still produce practical mean field maps and local fluctuation estimates when probe orientation, sampling duration, repeatability, and uncertainty are reported carefully [50], [51], [52].

To use measured samples in simulation, the finite flow data must be converted into updraft surrogates. This motivates the modelling framework developed in Chapter 4. Analytic Gaussian plume models provide compact descriptions for early guidance and energy exchange studies [53], [54]. Rotorcraft induced-flow and outwash theory offer physical intuition for induced velocity, source strength, and ground or boundary interaction effects [55], [56], [57]. Data-driven surrogate models, including Gaussian processes (GPs) [58], radial basis functions (RBFs) [59], and other response-surface methods [60], can interpolate measured data and, in some cases, provide uncertainty estimates. These models support validation and robustness studies, but should not be treated as exact descriptions of the physical flow, since finite measurements may miss temporal fluctuations, lateral velocities, or local wind shear.

2.3 Small Gliders and Control-Oriented Modelling

Small fixed-wing gliders are a natural benchmark for lift-aware studies, since their sink rate, recoverability, and terminal energy depend directly on aerodynamic energy exchange [61]. In the present thesis, the vehicle is also part of the closed-loop system: geometry, wing loading, control surface authority, structural tolerance, and mass budget determine the attainable flight envelope, the available recovery margin, and the modelling assumptions that can be used for controller design [33].

This coupling between airframe design and control is especially restrictive in a confined indoor arena with spatially varying inflow. Reducing wing loading and mass improves low-speed operation and reduces impact energy, but it also makes the aircraft more sensitive to inflow and less structurally tolerant. Increasing mass or wing loading can improve disturbance tolerance, but requires more space for turning and recovery. The representative platforms in Fig. 2.3 and Table 2.1 illustrate the relevant compromise and support the lightweight foam and carbon construction adopted for the present glider.

(a) ChiChi [62]. **(b)** CeePee [63]. **(c)** Feather^2 [64]. **(d)** FireFly 2.0 [65]. **(e)** Cory's glider [66].

Fig. 2.3. Representative small fixed-wing gliders in related work and hobby literature.

Table 2.1 Key characteristics of the representative small fixed-wing gliders shown in Fig. 2.3.

Parameter	**Symbol**	Unit	**ChiChic**	**CeePee** [63]	**Feather^2** [64]	**FireFly 2.0** [65]	**Cory's glider** [66]
Primary materials	-	-	balsa, plywood, heat-shrink film		balsa, CFRP tube, XPS foam	XPS foam, CFRP tube	Depron [67], CFRP tube
Control surface(s)	-	-	ailerons, ruddervators	ailerons, rudder, elevator	rudder, elevator	elevons	elevator
Mass	m	g	115 – 130	200	43 – 50	80 – 100	77
Wing span	b	m	0.600	0.750	0.736	0.812	0.813
Mean chord	$\bar{c}$	m	0.098	0.115	0.099	0.168	0.098
Wing area	S	m^2	0.0585	0.0860	0.0729	0.1368	0.0797
Aspect ratio	AR	-	6.15	6.54	7.43	4.82	8.30
Wing loading	$W\,S^{-1}$	$\mathrm{N\,m^{-2}}$	19.3 – 21.8	22.8	5.79 – 6.73	5.74 – 7.17	9.48

The dimensions and wing loadings in Table 2.1 also indicate the aerodynamic regime where such vehicles operate. For small fixed-wing gliders, the combination of short chord and low flight speed typically places the Reynolds number in the range between 10^4 and 10^5. In this regime, lift, drag, and stall behaviour are highly sensitive to laminar separation, surface roughness, aerofoil thickness, and manufacturing error [68], [69], [70].

Designing a small glider for this setting involves coupled trade-offs among aerodynamic efficiency, structural weight and stiffness, trim and stability, control authority, manufacturability, and mission performance [71]. These trade-offs may be explored through iterative design built on existing configurations, multidisciplinary design optimisation (MDO), or a combination of both [72] [73]. Open-source MDO frameworks such as OpenMDAO [74], SUAVE [75], and GPkit [76] support this type of design study. In this thesis, AeroSandbox [77] is selected as a lightweight differentiable framework for customised constraints and rapid iteration, giving a transparent and repeatable glider design workflow in Chapter 5.

The same low-Reynolds-number sensitivity discussed above also shapes the vehicle model used later for control. Since the relevant aerofoil geometry and Reynolds number range are not experimentally validated for the as-built aircraft, the aerodynamic model should not rely on a precise aerofoil database. Instead, it must remain numerically robust over the expected operating envelope while supporting trim, local linearisation, and repeated closed-loop simulation [78], [79].

Quasi-steady flat-plate aerodynamic models provide a practical solution to this requirement. For thin lifting surfaces, they give a compact representation that remains usable over a wider incidence range than conventional aerofoil models [80], [81]. Previous studies on small fixed-wing perching have shown that averaged high-incidence lift and drag behaviour can be estimated using flat-plate models, although individual manoeuvres remain affected by unsteady vortex dynamics and repeatability limits [66], [82]. However, since the flat-plate model only provides the nominal force structure, system identification is introduced to calibrate its parameters using flight test data. It estimates aerodynamic derivatives, damping terms, and control surface effectiveness from measured input-output time histories, and assesses predictive quality on a separate validation dataset [83], [84], [85].

The indoor updrafts add a separate modelling demand through the aerodynamic loading imposed on the lifting surfaces. As discussed in Section 2.2, the forcing field is spatially non-uniform and is represented by finite measured surrogates in simulation. If the vertical velocity varies significantly over the wing span, representing the flow only at the centre of gravity may miss asymmetric changes in local relative airflow and the resulting roll, pitch, and yaw moments. This leads to the panel-wise aerodynamic model in Chapter 6, where each panel is evaluated using its own local relative velocity.

2.4 Guidance and Control with Manoeuvre Primitives

Conventional fixed-wing control architectures separate reference generation from stabilisation. Cascaded PID controllers [86], linear quadratic regulators (LQR) [87], robust control [88], and model predictive control (MPC) [89] are standard tools for stabilising aircraft around trim conditions or tracking feasible references [90]. Path-following methods then convert geometric errors into course, bank, altitude, or flight path angle commands, [27], [91], [92]. This separation is effective when the aircraft has enough space to follow smooth references and when those references remain inside the attainable flight envelope.

For the present indoor problem, the same decomposition is restrictive. The useful safety volume is small relative to the turn radius and recovery distances of a fixed-wing glider, and the glider may encounter strong local aerodynamic forcing while already close to the wall, height, speed, or actuator limits. Hence, the guidance problem is better posed as the selection of short, recoverable manoeuvres [93].

Manoeuvre-based methods have a long history in robotics and aerospace applications. Sequential composition [94], manoeuvre automata [95], LQR-Trees [96], and funnel libraries [97] decompose nonlinear control problems into locally valid manoeuvre primitives, each with admissible entry regions and expected exit conditions. Fixed-wing perching studies are particularly relevant because they combine reduced-order high-incidence aerodynamic modelling, time-varying feedback, and experimental validation of short aggressive manoeuvres [66], [82]. Therefore, a manoeuvre primitive is treated here as a short section of flight controlled by its own local feedback law, with defined entry conditions, a nominal reference and command profile, a finite time horizon, exit checks, and outcome metrics. In this setting, LQR is used as a local stabiliser for individual primitives, where each primitive has its own reference state, nominal command, linearisation, and performance objective [96].

Once control is framed as choosing between candidate manoeuvres, constraints must be checked before execution. Reference and command governors provide established methods for modifying or rejecting proposed commands before constraint violation occurs [98], [99]. Viability theory [100] and set invariance [101] treat safety as a state space property, identifying admissible states, recoverable regions, and trajectories that preserve constraints. In this thesis, the same logic is applied to manoeuvre primitives: a candidate primitive is rejected if its entry conditions are not satisfied, or if its predicted execution violates the safety volume, actuator limits, flight envelope, or recovery requirements.

The primitive library is the finite set of candidate manoeuvres made available to this governor. Its size creates a coverage problem: adding primitives can enlarge the reachable behaviour set, but each new primitive requires its own entry checks, uncertainty tests, exit criteria, and rejection conditions before it can be used [66], [96]. Related motion-planning work shows that richer primitive sets can improve coverage or motion quality, but also increase validation and computational burden [97], [102]. Instead of assuming that increasing the library size automatically improves sim-to-real transfer, this thesis treats library design as a trade-off between behavioural coverage and robustness of individual primitives.

With the primitive library and viability governor defined, one guidance question remains: which locally promising lift region should be targeted next, and which primitive objective should be proposed to reach it? This problem is partially observed because the aircraft cannot measure the full lift field during one flight. Partially Observable Markov Decision Processes (POMDPs) [103], [104], belief-space planning [105], [106], informative planning [109], [110], and reinforcement learning (RL) [111], [112], [113] provide general tools for action selection under incomplete information. In autonomous soaring research, related ideas appear in heuristic thermalling [114], online updraft estimation [34], stochastic decision making [115], and RL-based lift exploitation [116], [117], [118]. As discussed in Chapter 6, these ideas are implemented here through a spatial lift belief, which is updated after each launch and suggests the next lift-seeking target, viability governor then decides whether the candidate primitive can be executed safely. The resulting control architecture is evaluated in Chapter 7 through primitive acceptance, rejection, degradation, and improvement across repeated launches in the presence of the reality gap.

Primitive selection defines the guidance logic, but closed-loop behaviour also depends on the sensing, computation, and command path linking the aircraft and controller [119], [120], [121]. Indoor flight experiments commonly use motion capture systems for repeatable state estimation, but calibration, marker visibility, update timing, and communication delay must be managed carefully [122], [123], [124]. Networked and off-board control studies further show that delay and packet timing can degrade stability and tracking performance [125], [126]. These implementation effects are quantified in Chapter 3.

2.5 Research Gap and Thesis Positioning

Taken together, the reviewed literature places this thesis at the intersection of several established but only partly connected lines of research. Robotics sim-to-real studies provide mature tools for transfer evaluation [10], [12], system identification [13], [14], and randomisation [15], [17], but most examples do not treat external aerodynamic forcing as a dominant source of mismatch between simulation and reality. Research on fixed-wing wind-aware guidance [27], [28] and airflow around buildings [29], [31] explains why spatial wind variation around obstacles and structures should be represented in the control problem. Autonomous thermalling [8], [35] and orographic soaring [38], [39] show that fixed-wing aircraft can exploit useful lift even when the surrounding flow field is partially observable. Fixed-wing perching studies [66], [82], together with LQR-Trees [96] and funnel library methods [97], illustrate how short feedback-stabilised manoeuvres can be associated with admissible entry regions and safety constraints. What remains missing is a fixed-wing sim-to-real transfer setting that treats the uncertain flow field as part of the reality gap, instead of separating the vehicle, controller, experimental setup, and aerodynamic environment into independent problems.

This gap cannot be closed by a guidance and control method alone. Recent fixed-wing research platforms and autopilot software provide reusable aircraft designs, simulation tools, and controller interfaces, but not the full experimental setting required for sim-to-real transfer [127], [128]. Robotics testbed and benchmarking studies further show that experiments are easier to reproduce when they use common interfaces, recorded data, shared procedures, and clearly defined hardware requirements [129], [130], [131]. For fixed-wing sim-to-real transfer studies, the missing element is a reusable, reproducible, and extensible workflow for indoor aerial robotics laboratories that use motion capture, a setup increasingly common in universities and research institutions [132], [133], [134]. This workflow must specify the test volume, state measurement, sensing and command latency, aircraft design and hardware, test environment design and characterisation, and the data needed to rerun each flight in simulation.

Accordingly, this thesis addresses these gaps in two coupled parts. It first develops the required experimental workflow, then proposes a viability-guided primitive selection algorithm for lift-aware guidance and control of a small fixed-wing glider in uncertain indoor updrafts. The algorithm is evaluated through the same workflow to test whether it can transfer from simulation to flight despite a reality gap caused by a combined mismatch in the vehicle, sensing, command path, and flow field.

3 Experimental Platform and System Architecture

Building on the research gap, this chapter turns the required experimental workflow into a concrete infrastructure for an indoor aerial robotics arena instrumented with a motion capture system. It defines the physical test setup, computation and state measurement architecture, command interface, and end-to-end system latency, providing the experimental basis used throughout the rest of the thesis.

3.1 Experimental Platform

The flight experiments were conducted in Imperial College London's Flight Arena, an indoor test space with dimensions of $10.0 \times 6.2 \times 5.5\,\mathrm{m}$. The Flight Arena is equipped with 16 Vicon Vantage cameras connected through a Power over Ethernet network that supplies camera power and transfers data to the Vicon synchronisation hardware and host PC [135]. Vicon Tracker software provides real-time 6-DoF rigid-body measurements with millimetre accuracy from at least three reflective markers fixed in a known geometry on each tracked object [122], [123].

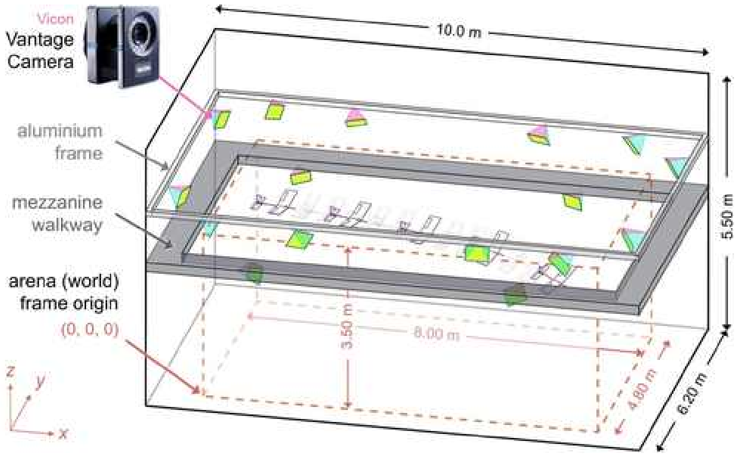


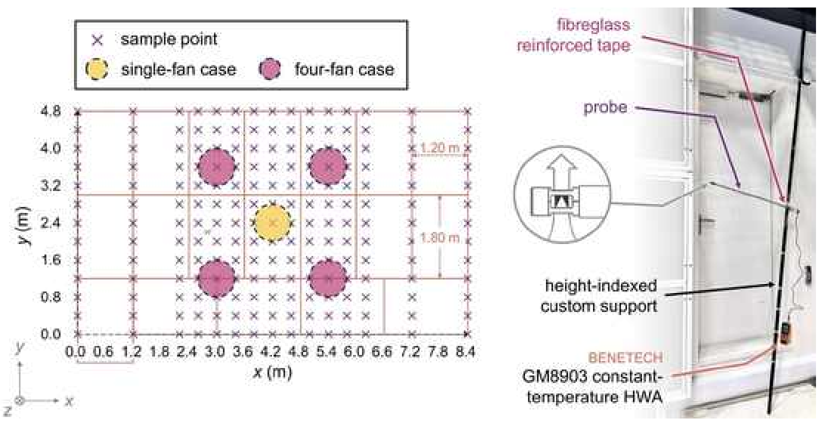
Fig. 3.1. Flight Arena with Vicon motion capture system.

Within this facility, flight tests were limited to an $8.0 \times 4.8 \times 3.5\,\mathrm{m}$ volume protected by the recovery net and floor padding, which defined the arena frame (world frame) coordinate system shown in Fig. 3.1. The test vehicle is a small fixed-wing glider, whose design and manufacture are presented in Chapter 5. For controller development in Chapter 6, a smaller operational region inside the arena frame, $x \in [1.2, 6.6]\,\mathrm{m}$, $y \in [0.0, 4.4]\,\mathrm{m}$, and $z \in [0.4, 3.5]\,\mathrm{m}$, is used so that the reflective markers mounted on the glider, shown later in Fig. 3.5 and Fig. 3.6, remain visible to at least 75% of the Vicon cameras.

3.1.1 Updraft Generation and Measurement Setup

To create repeatable indoor updraft environments for flow characterisation, modelling, and subsequent controller studies, experiments were carried out using either a single Clarke CAMAX30 axial drum fan [136] or four fans. The single-fan case provided an isolated baseline for updraft field characterisation and sensitivity tests, while the four-fan case produced a multi-source updraft for evaluating surrogate reconstruction quality and subsequent controller performance in a more spatially complex flow.

The updraft field was sampled on a locally refined Cartesian grid using a BENETECH GM8903 hot-wire anemometer mounted on a height-indexed rigid support [137]. Ambient temperature was held between 18.5 °C and 20.0 °C to minimise temperature-induced sensor drift [138], while the probe orientation was kept fixed to reduce errors caused by lateral flow in shear or swirl regions [49]. This setup provided a consistent basis for the dataset introduced in Chapter 4.



(a) Measurement grid. **(b)** Probe arrangement.

Fig. 3.2. Fan-generated updraft measurement setup.

3.2 Sensing, Computation, and Command Architecture

Sensing, computation, and command transmission determine how the state is measured, where the controller runs, what delays are introduced, and whether each experiment can be replayed in simulation. Since the full airframe design is introduced in Chapter 5, this section treats the test vehicle only through four servo channels relevant to the actuation link: left aileron, right aileron, elevator, and rudder.

3.2.1 Computation and State Measurement Architecture

Table 3.1 compares computation architectures by integration difficulty, real-time performance, onboard size, weight, and power (SWaP) burden, reliability, modularity, and extra cost relative to the available infrastructure. The ground computer is selected to keep the glider mechanically simple by handling control, logging, and safety supervision off-board.

Table 3.2 scores state estimators by integration, accuracy, robustness, real-time performance, setup ease, and additional cost. Here, robustness refers to sensitivity to test conditions, and setup ease covers calibration and maintenance effort. Vicon motion capture system is chosen since it provides fast and accurate 6-DoF tracking without adding sensors, wiring, or processing load to glider, supporting the off-board control selected above.

Table 3.1 Computation architecture decision matrix.

Architecture	**Integration**	**Real-time**	**Low SWaP**	**Reliability**	**Modularity**	**Low cost**	**Score**
Weight	10%	20%	30%	20%	10%	10%	100%
Ground Computer	8	9	10	4	9	10	**8.3**
Onboard HW	5	7	3	6	5	5	**5.0**
Onboard LW	3	4	9	9	7	7	**7.0**
Split Compute	9	9	8	7	3	7	**7.5**

Table 3.2 State estimator decision matrix.

Estimator	**Integration**	**Accuracy**	**Robustness**	**Real-time**	**Setup ease**	**Low cost**	**Score**
Weight	10%	30%	20%	20%	10%	10%	100%
IMU	9	3	8	10	9	7	**7.0**
UWB	7	6	7	7	5	4	**6.2**
UWB + IMU	8	7	8	8	5	3	**6.9**
Vicon	9	10	9	9	4	10	**8.9**
Vicon + IMU	7	10	10	10	3	7	**8.7**
Camera + IMU	7	6	5	6	7	6	**6.0**
LiDAR + IMU	6	8	8	7	6	5	**7.1**
Optical Flow	7	3	5	8	8	6	**5.6**

3.2.2 Command Interface Selection

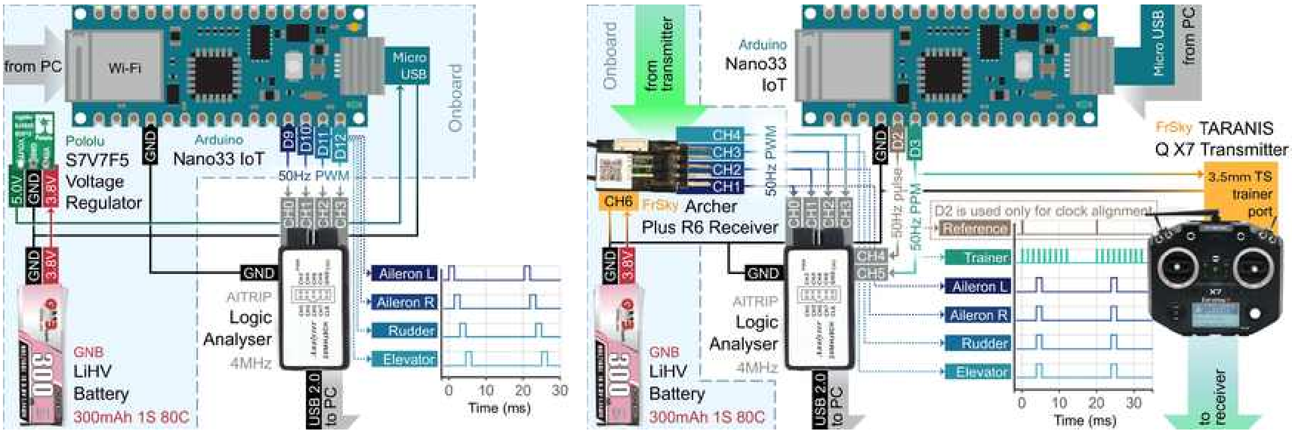
With the ground computer and Vicon motion capture system selected, a compatible command interface is needed to carry control commands to the onboard actuators. It should be inexpensive, lightweight, simple to integrate, stable, durable, reproducible, and extensible to future aircraft or actuator changes.

These requirements leave two practical routes. In direct servo command path (Fig. 3.3a), an onboard microcontroller receives commands over Wi-Fi and directly generates servo PWM signals [3]. In commercial off-the-shelf (COTS) radio control (RC) command chain (Fig. 3.3b), the microcontroller sends a PPM signal [4] to a transmitter [139] via its trainer port, the transmitter then relays the commands through the radio link to an onboard receiver [140], which outputs separate servo PWM signals.

[3] Pulse-width modulation (PWM) represents a command by varying the width of repeated pulses. For standard RC servos, the pulse width sets the demanded servo position.

[4] Pulse-position modulation (PPM) encodes multiple RC channel commands as a sequence of pulses within a repeating timing frame, allowing several channels to be carried on a single signal line.

As discussed in Section 2.4, previous studies have shown that implementation timing affects closed-loop control [120], but stage-by-stage latency measurements of command paths are rarely reported. To address this gap, a repeatable electrical timing test was designed for two candidate routes, each driven by a sequence of step changes in the four control surface commands. For a given run, the ground computer logged the scheduled command and dispatch time, while the Arduino Nano 33 IoT [141] recorded the board receive and update times for the same command sequence. A logic analyser [142] was connected to the relevant pulse lines to captured the first electrical output change after each command step [5].



(a) Direct servo command path. The logic analyser records the servo PWM pulses directly produced by the Arduino Nano 33 IoT, which define the final electrical command endpoint.

(b) COTS RC transmitter-receiver command chain. The logic analyser records the reference pulse, trainer-side PPM signal, and receiver-side servo PWM pulses. The receiver PWM transition is used as the final electrical command endpoint.

Fig. 3.3. Electrical command latency test setups for the two candidate interfaces.

Five timing runs were processed for each command interface. Each run lasted 59.0 s and contained 2952 command updates for four control surfaces. The command logs and pulse traces were then aligned offline to compute the electrical delay at each stage of the command path and the final endpoint latency.

Table 3.3 Stage-by-stage electrical timing comparison for the two command interfaces.

Path segment	Unit	Direct servo				COTS RC			
		median	P_{95} [6]	P_{99} [6]	σ	median	P_{95} [6]	P_{99} [6]	σ
Ground command to dispatch	ms	0.25	0.99	4.03	0.89	0.16	0.51	1.17	0.34
Ground dispatch to board receive	ms	22.9	32.4	34.1	5.68	6.83	7.38	7.63	0.74
		over Wi-Fi				via micro USB			
Board receive to firmware update	ms	0.20	0.23	0.30	0.02	11.6	16.8	17.1	5.08
		servo PWM updated				trainer PPM updated			
Firmware update to electrical output	ms	16.2	28.0	30.5	6.99	3.01	20.1	20.1	7.83
		next servo PWM pulse				next trainer PPM frame			
Electrical output to receiver output	ms	-	-	-	-	6.89	33.6	45.6	12.8
		-				receiver PWM updated			
Ground command to final endpoint	ms	**40.4**	**55.0**	**58.3**	**9.46**	**33.9**	**57.8**	**64.8**	**11.6**

Table 3.3 shows that the direct servo path has slightly higher median ground computer command to electrical command endpoint latency, but lower P_{95}, P_{99}, and standard deviation. Based on latency

[5] The analyser records digital voltage signals over time. In these tests, it was sampled at 4 MHz, corresponding to a 0.25 μs interval, so the start, end, and width of millisecond-scale command pulses can be measured across several channels.

[6] P_{95} and P_{99} are the delay values below which 95% and 99% of matched events fall. They describe rare high-latency cases, which matter because the controller must remain stable beyond the typical delay.

alone, it would be preferable for fast closed-loop control. However, the interface also had to support repeatable flight tests, safe intervention, practical integration, and future extension. During the latency test, the COTS RC path ran for 35 – 50 min, compared with 7 – 15 min for the direct servo setup, while preserving transmitter pilot override as a safety fallback, remaining compatible with standard RC servos without modification, and reducing custom onboard integration work. The direct servo route also heavily depends on local Wi-Fi configuration, making exact reproduction across laboratories less reliable.

Therefore, the COTS RC path is adopted as the experimental command interface used in this thesis and, together with the previously selected sensing and computation architecture, forms the final flight test system illustrated in Fig. 3.4. The system was completed with DOMAN S0020 micro servos [143], chosen for low mass and sufficient torque, with limited positioning precision accepted as the trade-off.

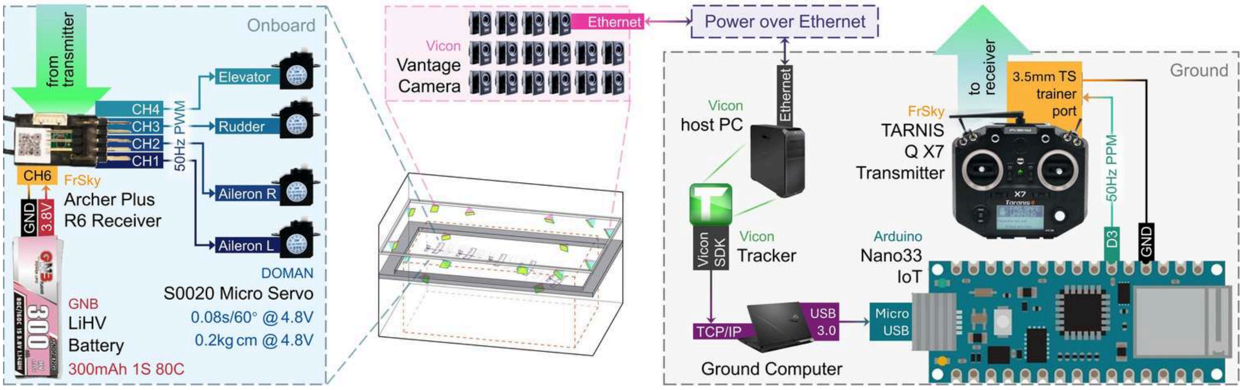


Fig. 3.4. Selected flight test sensing, computation, and command architecture.

Hardware specifications, detailed timing methods, and per-run statistics are given in Appendix B.1.

3.3 State Processing and Closed-Loop Timing

The sensing and command architecture above forms an off-board feedback loop, but only the electrical command timing has been characterised. This section then describes state construction and completes the latency model used for controller design by combining the sensing, filtering, and actuator delays.

3.3.1 Vicon-Based State Processing and Filtering

Vicon system was configured to stream rigid-body pose at 200 Hz [135]. For consistency during flight analysis and replay, all processed poses are aligned with the arena frame shown in Fig. 3.1, with airframe body axes following the flight dynamics convention in Fig. 3.5. At discrete Vicon sample k, the raw position, $\mathbf{p}_{V,k} = [x_{V,k},\, y_{V,k},\, z_{V,k}]^\top$, and attitude, $\boldsymbol{\eta}_{V,k} = [\phi_{V,k},\, \theta_{V,k},\, \psi_{V,k}]^\top$, are converted using offsets calibrated by holding the glider at a known point $\mathbf{p}^\star$ in the arena frame with zero roll, pitch, and yaw:

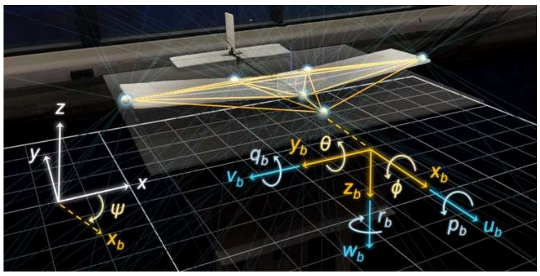


Fig. 3.5. Airframe rigid-body tracking.

$$\mathbf{p}_k \;=\; \mathbf{p}_{V,k} + \mathbf{p}^\star - \bar{\mathbf{p}}_V, \qquad \boldsymbol{\eta}_k \;=\; \mathrm{wrap}_\pi\left(\mathrm{diag}(1,\, -1,\, -1)\, \boldsymbol{\eta}_{V,k} + \boldsymbol{\epsilon}_a\right), \tag{3.1}$$

where $\bar{\mathbf{p}}_V$ is the mean raw position recorded while the glider is held at $\mathbf{p}^\star$, $\boldsymbol{\epsilon}_a = [\,\epsilon_\phi,\, \epsilon_\theta,\, \epsilon_\psi\,]^\top$ contains the calibrated roll, pitch, and yaw alignment, and $\mathrm{wrap}_\pi(\cdot)$ maps each angle to the interval $[-\pi,\, \pi)$.

The transformed position $\mathbf{p}_k = [x_k,\, y_k,\, z_k]^\top$ and attitude $\boldsymbol{\eta}_k = [\phi_k,\, \theta_k,\, \psi_k]^\top$ in (3.1) form the first two blocks of the 15-state vector used by the flight controller described in Chapter 6:

$$\mathbf{x}_k = \left[\,\mathbf{p}_k^\top, \quad \boldsymbol{\eta}_k^\top, \quad \mathbf{v}_{b,k}^\top, \quad \boldsymbol{\omega}_{b,k}^\top, \quad \boldsymbol{\delta}_k^\top\,\right]^\top. \tag{3.2}$$

The remaining blocks are derived as follows. The body-axis velocity $\mathbf{v}_{b,k} = [u_{b,k},\, v_{b,k},\, w_{b,k}]^\top$ is obtained by finite differencing adjacent transformed positions, applying an 8 Hz one-pole low-pass filter, and rotating the result into the body frame. The body angular rate $\boldsymbol{\omega}_{b,k} = [p_{b,k},\, q_{b,k},\, r_{b,k}]^\top$ is estimated from successive attitude matrices using an $SO(3)$ rotation-delta observer [144], avoiding jumps when Euler angles wrap around $\pm\pi$, then filtered with the same 8 Hz low-pass filter. The control surface state $\boldsymbol{\delta}_k = [\delta_{a,k},\, \delta_{e,k},\, \delta_{r,k}]^\top$ is propagated from command history using the response model introduced below.

The state processing procedures and associated algorithms are detailed in Appendix B.2.

3.3.2 Control Surface Response and Implementation Timing

The control interface in Fig. 3.4 receives three normalised commands, $\mathbf{u}_k^{\mathrm{cmd}} = [u_{a,k}^{\mathrm{cmd}},\, u_{e,k}^{\mathrm{cmd}},\, u_{r,k}^{\mathrm{cmd}}]^\top$, for aileron, elevator, and rudder action. Before execution in both simulation and reality, each command is rounded to the nearest value in $\{-1.0,\, -0.8, \ldots,\, 0.8,\, 1.0\}$, reducing sensitivity to small changes that cannot be reproduced reliably due to servo deadband, linkage looseness, and imperfect centering.

To measure the control surface response latency, reflective markers were temporarily attached to the individual surfaces and tracked by Vicon system, while repeated step commands were sent through the selected COTS RC command chain. Since the marker-loaded surfaces tend to oscillate after abrupt step commands, each measured response was scaled between its initial average position and final settled position. The times to reach 10% and 50% of normalised surface movement were then extracted.

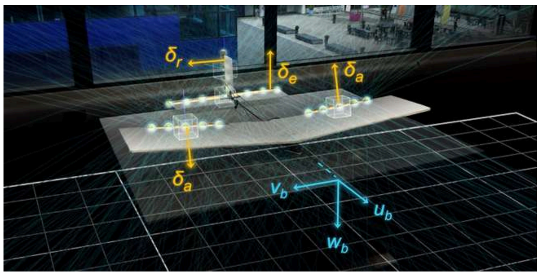


Fig. 3.6. Control surface tracking.

This latency test was repeated five times, producing 800 valid command to control surface events. Vicon capture to report latency was also measured in the same runs from the logged difference between each Vicon frame time and the host time when the corresponding pose became available.

Table 3.4 Adopted sensing, filtering, and actuator timing statistics.

Quantity	Symbol	Unit	median	P_{95}	P_{99}	max	σ
Vicon capture to report latency	$T_{V,\ell}$	ms	14.9	16.9	17.9	36.9	0.92
8 Hz low-pass state filter delay	T_f	ms	20.0	-	-	-	-
Surface initial-response	t_{10}	ms	72.8	107.3	119.6	137.0	32.0
Surface half-response	t_{50}	ms	107.6	130.8	140.5	150.8	16.4

For simulation environment in Chapter 6, the median and conservative timing cases are defined to evaluate controller robustness to latency mismatch before flight testing. In both cases, the delay in state feedback is treated separately from actuator response [145]. The Vicon-based state feedback delays are

$$\tilde{T}_x = \tilde{T}_{V,\ell} + T_f = 34.9\,\mathrm{ms}\,, \qquad T_x^{\mathrm{cons}} = T_{V,\ell}^{P95} + T_f = 36.9\,\mathrm{ms}\,. \tag{3.3}$$

The measured surface response is modelled using the median initial-response delay, $\tilde{t}_{10}$, followed by first-order motion toward the aggregate surface command from d samples earlier, denoted $\boldsymbol{\delta}_{k-d}^{\mathrm{cmd}}$:

$$\boldsymbol{\delta}_{k+1} = \boldsymbol{\delta}_k + \left[1 - \exp\left(-\Delta t_k/\tau_\delta\right)\right]\left(\boldsymbol{\delta}_{k-d}^{\mathrm{cmd}} - \boldsymbol{\delta}_k\right), \qquad d\Delta t_k \approx \tilde{t}_{10}\,, \tag{3.4}$$

where the time constant τ_δ is chosen so that the first-order response reaches exactly 50% at t_{50}:

$$\tilde{\tau}_\delta = \frac{\tilde{t}_{50} - \tilde{t}_{10}}{\ln 2} = 50.5\,\mathrm{ms}\,, \qquad \tau_\delta^{\mathrm{cons}} = \frac{t_{50}^{\max} - \tilde{t}_{10}}{\ln 2} = 112.5\,\mathrm{ms}\,. \tag{3.5}$$

The full response extraction procedure and per-run timing tables are provided in Appendix B.3.

4 Updraft Characterisation and Modelling

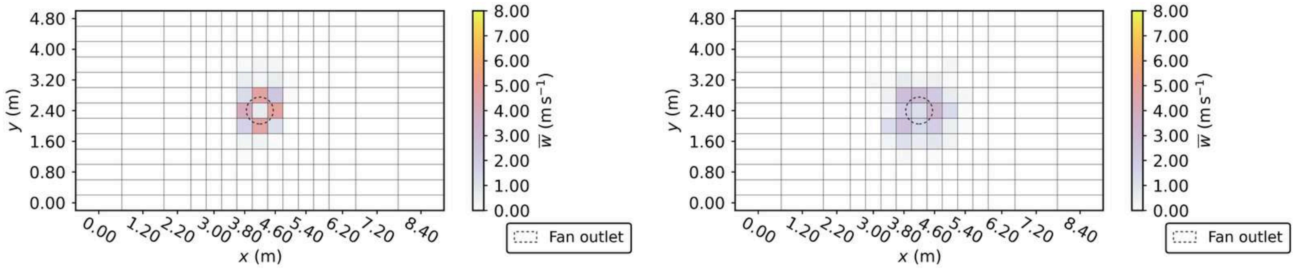
In this thesis, the dominant contributor to the reality gap is the indoor updraft, which strongly affects the glider's vertical energy exchange. This chapter characterises the measured fan-generated flow and introduces the modelling framework used later for controller development in simulation.

4.1 Updraft Measurement and Dataset

Using the artificial updraft and measurement setup introduced in Section 3.1.1, the local vertical velocity $w(t)$ was characterised for both fan configurations, with the quasi-steady mean $\bar{w}$ used for modelling. For each configuration, $\bar{w}(x_i, y_i, z_{\mathrm{fan},k})$ was recorded on seven horizontal planes at $z_{\mathrm{fan}} \in \{0.20, 0.35, 0.50, 0.75, 1.10, 1.60, 2.20\}$ m above the fan outlet plane. Measurements were taken at 195 grid points on each plane, giving 1365 spatial locations per configuration across the seven planes.

Fig. 4.1. Time-lapse composite of anemometer measurements of the fan-generated updraft.

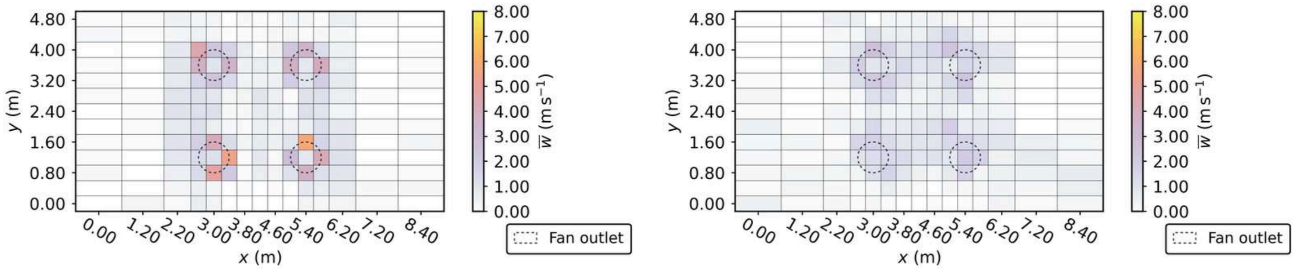
Representative mean-field maps at $z_{\mathrm{fan},2} = 0.35$ m and $z_{\mathrm{fan},6} = 1.60$ m are shown in Fig. 4.2 and 4.3, while annotated mean-field maps for all horizontal planes in both layouts are provided in Appendix C.2.



(a) 0.35 m above fan outlet plane.

(b) 1.60 m above fan outlet plane.

Fig. 4.2. Spatial distribution of mean vertical velocity at representative heights (single fan).



(a) 0.35 m above fan outlet plane.

(b) 1.60 m above fan outlet plane.

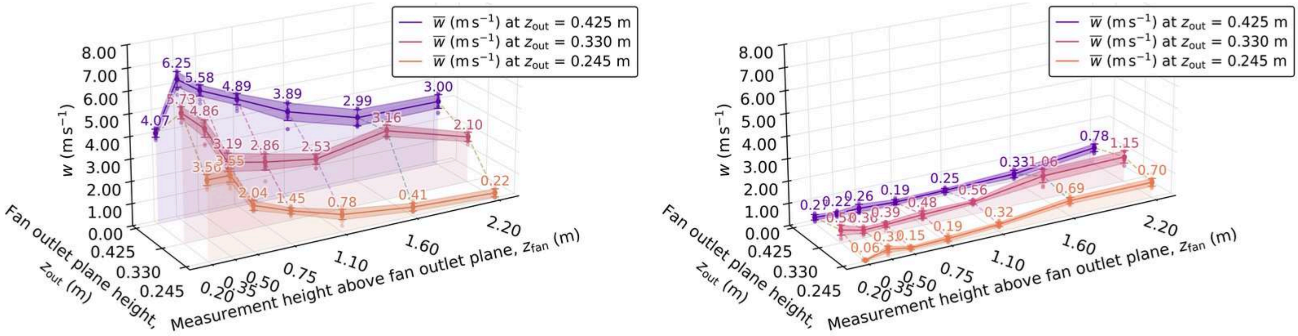
Fig. 4.3. Spatial distribution of mean vertical velocity at representative heights (four fans).

In addition to the mean-field maps, temporal variability was assessed at six representative points spanning the near-outlet, off-outlet, and far-field regions. Each record consisted of 15 samples at 1 Hz, with duplicate runs to check repeatability. In the single-fan case, fan outlet height and speed were varied for sensitivity tests, while the four-fan case was measured only at the nominal operating condition.

4.2 Experimental Updraft Field Characterisation

The fan field shows strong height dependence and multi-source interaction. In single-fan case (Fig. 4.2), the peak occurs on an asymmetric annular ring, that spreads and weakens with height. The peak decreasing from 4.86 m/s at $z_{\mathrm{fan},2} = 0.35$ m to 3.16 m s^{-1} at $z_{\mathrm{fan},6} = 1.60$ m. In the four-fan case (Fig. 4.3), four distinct asymmetric cores are observed at low altitude, which weaken and overlap with height. The local peaks reduce from $4.02-5.80\,\mathrm{m\,s^{-1}}$ at $z_{\mathrm{fan},2} = 0.35\,\mathrm{m}$ to $2.04-2.60\,\mathrm{m\,s^{-1}}$ at $z_{\mathrm{fan},6} = 1.60\,\mathrm{m}$, while the overlap raises the background lift in the central region.

Ground effect is then evaluated by varying the fan outlet height across low (0.245 m), baseline (0.330 m), and high (0.425 m) configurations. As shown in Fig. 4.4a, the effect is strongest near-core, where rising outlet height increases the lower-profile updraft and delays its decay with height, consistent with rotor-induced ground effect behaviour [56], [146]. The off-core response in Fig. 4.4b is weaker and less monotonic, showing local redistribution rather than a uniform change across the field. Fig. 4.4 further shows that the fluctuation level is spatially non-uniform: near-core point 1 exhibit $\sigma = 0.26-0.41\,\mathrm{m\,s^{-1}}$ over $z_{\mathrm{fan}} = 0.20-1.60\,\mathrm{m}$, while off-core point 4 show smaller absolute fluctuations, about $0.09-0.30\,\mathrm{m\,s^{-1}}$, but larger relative variability as the mean velocity is low [7]. Overall, the baseline outlet height was adopted as a compromise between updraft strength and reduced sensitivity to ground effects.

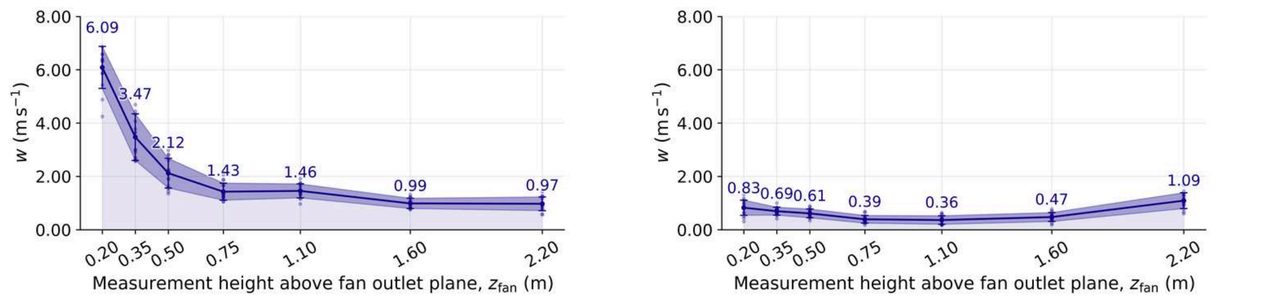


(a) Point 1 (near-core) at $(x,\, y) = (4.60,\, 2.40)$ m. **(b)** Point 4 (off-core) at $(x,\, y) = (3.80,\, 1.60)$ m.

Fig. 4.4. Vertical velocity profiles at representative sampling points for three fan outlet plane heights (single fan). The solid line shows the mean, semi-transparent markers show the 15 samples (1 Hz over 15 s) at each measurement height, and shaded bands show $\pm 1\sigma$.

Fan speed sensitivity is also examined at 8800 rpm and 960 rpm. The higher speed was adopted, since the lower one led to occasionally blade-guard contact, likely due to insufficient centrifugal stiffening.

In the four-fan configuration, plume overlap produces non-zero mean velocity and measurable variance at points between the individual cores, as shown in Fig. 4.5. The largest absolute fluctuations occur in the near-core and interaction region. For example, at $z_{\mathrm{fan},2} = 0.35\,\mathrm{m}$, point 1 give $\bar{w} = 3.47\,\mathrm{m\,s^{-1}}$ with $\sigma = 0.87\,\mathrm{m\,s^{-1}}$, while the point 4 gives $\bar{w} = 0.69\,\mathrm{m\,s^{-1}}$ with $\sigma = 0.15\,\mathrm{m\,s^{-1}}$. These measurements clearly show that the uncertainty is highly non-uniform and varies with both height and position.



(a) Point 1 (near-core) at $(x,\, y) = (4.60,\, 2.40)$ m. **(b)** Point 4 (interaction) at $(x,\, y) = (3.80,\, 1.60)$ m.

Fig. 4.5. Vertical velocity profiles at representative sampling points (four fans). The solid line shows the mean, semi-transparent markers show the 15 samples (1 Hz over 15 s) at each measurement height, and shaded bands show $\pm 1\sigma$.

The complete set of corresponding profiles for all six sampling points is provided in Appendix C.3.

[7] Since these records span only 15 s at 1 Hz, the reported σ only represent local short horizon fluctuation levels.

In conclusion, the measurements show that the updraft field is non-axisymmetric, height-dependent, and subject to spatially varying uncertainty. It is also sensitive to outlet height and fan speed, as observed from the single-fan sensitivity sweeps, while the four-fan case is additionally affected by plume overlap and redistribution under multi-source interaction.

4.3 Updraft Modelling Framework

These observed features motivate the simulation surrogates developed in this section, which are designed to preserve the measured field structure while retaining a representation of spatial uncertainty.

For a configuration with F fans, with fan centres $(x_f,\, y_f)$ for $f = 1, \ldots, F$, the horizontal distance and azimuth angle of an arbitrary point $(x,\, y,\, z)$ relative to fan f are

$$r_f\,(x,\, y) \;=\; \sqrt{(x - x_f)^2 \,+\, (y - y_f)^2}\,, \qquad \theta_f\,(x,\, y) \;=\; \operatorname{atan2}(y - y_f,\, x - x_f)\,. \tag{4.1}$$

Each sampling point $(x_i,\, y_i)$ is then assigned to its nearest fan f, with relative radius r_i:

$$f_i \;=\; \arg \min_{f \in \{1,\ldots,F\}} \, r_f\,(x_i,\, y_i)\,, \qquad r_i \;=\; r_{f_i}\,(x_i,\, y_i)\,. \tag{4.2}$$

At measurement height $z_{\text{fan},k}$, the fluctuation scale is mapped from six representative time-series measurements in Section 4.1. For each fan, these data define an empirical radial fluctuation profile, $\sigma_f(r_f,\, z_{\text{fan},k})$. The fluctuation assigned to sample i is then obtained by inverse-distance blending:

$$\sigma_i = \sum_{f=1}^{F} \omega_f\, \sigma_f(r_f,\, z_{\text{fan},k})\,, \qquad \omega_f = \frac{(r_f + \epsilon)^{-p}}{\sum_{g=1}^{F} (r_g + \epsilon)^{-p}}\,. \tag{4.3}$$

The corresponding fluctuation field plots at all measurement heights are given in Appendix D.1.

4.3.1 Error Metrics for Model Fitting

Model performance is quantified by comparing the fitted updraft field with the measured mean vertical velocity. For each sample (x_i, y_i) at height $z_{\text{fan},k}$, the pointwise error is defined as

$$e_i(z_{\text{fan},k}) \;=\; w_{\text{AG}}(x_i,\, y_i,\, z_{\text{fan},k}) \;-\; \bar{w}(x_i,\, y_i,\, z_{\text{fan},k})\,. \tag{4.4}$$

Two complementary error measures are used. To summarise the overall mismatch, the per-height sum of absolute errors (SAE) [147] and its aggregate over all heights are computed:

$$\text{SAE}(z_{\text{fan},k}) \;= \sum_i |e_i(z_{\text{fan},k})|\,, \qquad \text{SAE}_{\text{total}} \;= \sum_k \sum_i |e_i(z_{\text{fan},k})|\,. \tag{4.5}$$

To account for measurement variability, weighted root mean square error (WRMSE) [147] is also used:

$$\text{WRMSE}(z_{\text{fan},k}) \;= \sqrt{\frac{\sum_i \left(\frac{e_i(z_{\text{fan},k})}{\sigma_i(z_{\text{fan},k})}\right)^2}{\sum_i \left(\frac{1}{\sigma_i(z_{\text{fan},k})}\right)^2}}\,, \qquad \text{WRMSE}_{\text{total}} \;= \sqrt{\frac{\sum_k \sum_i \left(\frac{e_i(z_{\text{fan},k})}{\sigma_i(z_{\text{fan},k})}\right)^2}{\sum_k \sum_i \left(\frac{1}{\sigma_i(z_{\text{fan},k})}\right)^2}}\,. \tag{4.6}$$

Since the reference data are time-averaged, the reported SAE and WRMSE quantify only how closely each model reproduces the measured mean field, and do not represent the unsteady flow experienced by the real glider [148]. Therefore, they are used mainly for relative comparison between models.

4.3.2 Harmonic Annular Gaussian Surrogate

As discussed in Section 4.2, the fan-generated updraft exhibits two dominant structural features: a ring-shaped peak and clear azimuthal asymmetry. A control-oriented surrogate should therefore preserve the compact annular structure of the mean field while remaining flexible enough to represent the observed directional variation. Inspired by the harmonic inflow representations in rotorcraft induced-flow modelling [56] and motivated by the common use of Gaussian plume fits in soaring simulation [53], a harmonic annular Gaussian model is designed to represent the measured field:

$$w_{\text{HAG}}(x,\, y\,;\, z) \;=\; \sum_{f=1}^{F} \left[w_{0,f}\,(z) \;+\; A_{\text{ring},f}\,(\theta_f\,(x,\, y);\, z)\, \exp\!\left(- \left(\frac{r_f\,(x,\, y) - r_{\text{ring},f}(z)}{\delta_{\text{ring},f}(z)} \right)^{2} \right) \right], \tag{4.7}$$

where w_0 is the background vertical velocity, r_{ring} and δ_{ring} are the ring radius and thickness, and A_{ring} is the azimuth-dependent ring strength, written as a constant term plus an N-th order Fourier series:

$$A_{\text{ring},f}\,(\theta_f\,(x,\, y);\, z) \;=\; a_{0,f}(z) \;+\; \sum_{n=1}^{N_f} \Big(a_{n,f}(z)\cos(n\,\theta_f) \;+\; b_{n,f}(z)\sin(n\,\theta_f) \Big)\,. \tag{4.8}$$

For fan f, the corresponding parameter block of the model in (4.4) is given by

$$\boldsymbol{\theta}_{\text{ring},f}\,(z) = \Big[\, w_{0,f}(z),\, r_{\text{ring},f}(z),\, \delta_{\text{ring},f}(z),\, a_{0,f}(z),\, \{a_{n,f}(z),\, b_{n,f}(z)\}_{n=1}^{N_f} \,\Big]^{\top}, \tag{4.9}$$

and the full parameter vector used for fitting is obtained by stacking the parameter blocks for all fans:

$$\boldsymbol{\Theta}_{\text{ring}}(z) \;=\; \big[\, \boldsymbol{\theta}_{\text{ring},1}^{\top}(z),\, \ldots,\, \boldsymbol{\theta}_{\text{ring},F}^{\top}\,(z) \,\big]^{\top}. \tag{4.10}$$

At each measurement height $z_{\text{fan},k}$, $\boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k})$ is estimated by solving a least-squares optimisation problem with robust loss function $\rho\,(\cdot)$ [149]. To reduce sensitivity to local minima and improve the chance of reaching a good solution, the optimisation is initialised from multiple starting points [150]:

$$\begin{aligned} \underset{\boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k})}{\text{minimise}} \quad & \sum_{i=1}^{M} \rho_{f_i}(u_i) \;+\; \mathcal{R}(\boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k}),\, \boldsymbol{\gamma})\ , \\ \text{subject to} \quad & a_{0,f}(z_{\text{fan},k}) \geq 0\,, \quad \delta_{\text{ring},f}\,(z_{\text{fan},k}) \in [\,0.12,\, 1\,]\,, \\ & w_{0,f}\,(z_{\text{fan},k}) \in [\,\bar{w}_{\min}(z_{\text{fan},k}) - 1,\, \bar{w}_{\max}(z_{\text{fan},k}) + 1\,]\,. \end{aligned} \tag{4.11}$$

Here, M is the total number of measurements, u_i denotes the normalised residual, and $\mathcal{R}(\cdot,\,\cdot)$ is a regularisation term introduced to limit overfitting of the Fourier coefficients [151].

The normalised residual in (4.10) is constructed such that the denominator introduces variance-aware weighting, so samples with larger measured fluctuations have lower weight in the fit:

$$u_i = \frac{w_{\text{HAG}}(x_i,\, y_i;\, z_{\text{fan},k},\, \boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k})) - \bar{w}_i}{f_{\text{scale},\, f_i}\,\sigma_i(z_{\text{fan},k})}. \tag{4.12}$$

The regularisation term in (4.10) with hyperparameters $\boldsymbol{\gamma} = (\lambda_{\text{ridge}},\, \lambda_{\text{cap}},\, \kappa,\, \eta,\, N)$ is defined as

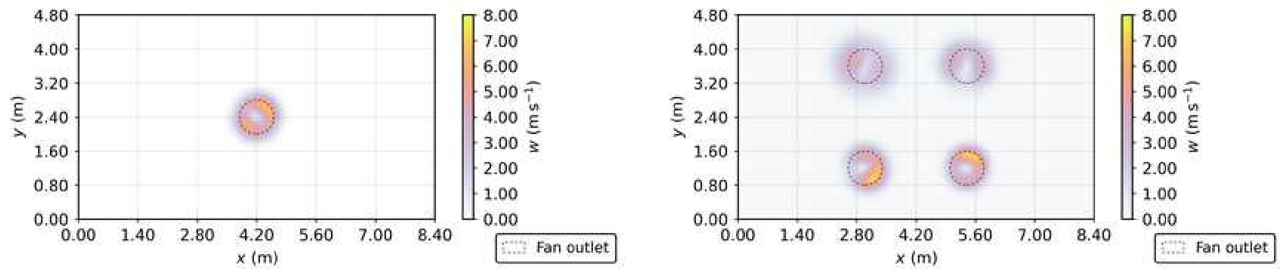
$$\begin{aligned} \mathcal{R}(\boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k}),\, \boldsymbol{\gamma}) \;=\; \sum_{f=1}^{F}\sum_{n=1}^{N_f} n^{\eta_f} \Big\{ & \lambda_{\text{ridge},f} \|[a_{n,f},\, b_{n,f}]^{\top}\|_2^2 \\ & + \lambda_{\text{cap},f} \Big[\max\!\big(0,\, \|[a_{n,f},\, b_{n,f}]^{\top}\|_2 - \kappa_f\, a_{0,f}\big) \Big]^2 \Big\}, \end{aligned} \tag{4.13}$$

which combines: (i) an L_2 term that penalises large harmonic coefficients [151], and (ii) a soft “cap” that activates only when the harmonic magnitude exceeds a fixed fraction of a_0 [152]. Both penalties are weighted by n^{η} to filter out higher-order harmonics that are not clearly present in the measurements [153].

Accordingly, three aspects of the fitting setup are evaluated to select the configuration that minimises total SAE and WRMSE across all measurement heights: (i) the robust loss function $\rho\,(\cdot)$ in (4.10), including soft-L_1 [154], Huber [148], and Cauchy [155]; (ii) the corresponding scale parameter f_{scale} in

(4.11); and (iii) the regularisation hyperparameters $\boldsymbol{\gamma}$ in (4.12).

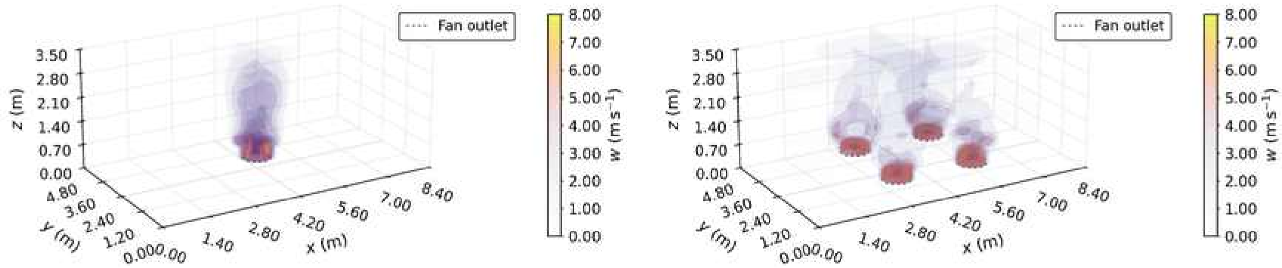
The selected configurations use the same robust loss, soft-L_1, but different scale and regularisation settings. For single-fan fit, $f_{\text{scale}} = 1.5$ and $\boldsymbol{\gamma} = (0.05,\ 0.20,\ 0.70,\ 1.5,\ 2)$. For four-fan case, the left pair of fans uses $f_{\text{scale}} = 1.5$ while the right pair uses $f_{\text{scale}} = 1$, with same $\boldsymbol{\gamma} = (0.03,\ 0.10,\ 0.80,\ 1.5,\ 2)$ [8]. The weaker harmonic penalties in four-fan fit allow stronger azimuthal correction, while the fan-dependent robust scale indicates that the right-side flow exhibits larger fluctuations.



(a) single-fan case. **(b)** four-fan case.

Fig. 4.6. Fitted mean updraft map of the harmonic annular Gaussian updraft surrogate at a single measurement plane 0.35 m above the fan outlet plane.

Finally, to obtain an uncertainty-aware continuous model for simulation, $\{\boldsymbol{\Theta}_{\text{ring}}(z_{\text{fan},k})\}_{k=1}^{7}$ and σ_i are interpolated over height, z, using shape-preserving piecewise cubic Hermite interpolation (PCHIP) [156]. Near and above the highest measured plane, extrapolation is blended to a separate upper-plane PCHIP fitted only through $z_{\text{fan},5}$, $z_{\text{fan},6}$, and $z_{\text{fan},7}$. Below the lowest measured plane, the parameters are fixed to their values at $z_{\text{fan},1}$ to reduce sensitivity to unsteadiness near the outlet [157].



(a) single-fan case. **(b)** four-fan case.

Fig. 4.7. 3D rendering of the harmonic annular Gaussian updraft surrogate mean vertical velocity field.

4.3.3 Residual Correction Using Gaussian Process Regression

Although the harmonic annular Gaussian model captures the dominant non-axisymmetric ring-shaped structure, small local discrepancies remain, especially where plume interaction and measurement noise are not fully captured by the parametric fit. To model these residual errors, a Gaussian process (GP) regressor is introduced as a residual corrector to refine the harmonic annular Gaussian updraft representation [158], [159]. GP regression is a non-parametric Bayesian method that treats the unknown mapping $g(\cdot)$ as a random function whose values at any finite set of inputs are jointly Gaussian [58].

In the present case, the GP is trained on the residual dataset with M measurement points,

$$\mathcal{D} = \{(x_i,\ y_i,\ z_i,\ w_{\text{res},i},\ \sigma_i)\}_{i=1}^{M}, \qquad z_i \in \{z_{\text{fan},k}\}_{k=1}^{7}, \qquad w_{\text{res},i} = \bar{w}_i - w_{\text{HAG},i}\,. \tag{4.14}$$

Using polar inputs, standardised by the sample mean μ_d and standard deviation s_d [159],

$$\boldsymbol{\phi}_i = [\, r_i,\ \cos\theta_i,\ \sin\theta_i,\ z_i \,]^{\top}, \qquad \tilde{\phi}_{i,d} = \frac{\phi_{i,d} - \mu_d}{s_d}, \qquad d \in \{1, 2, 3, 4\}, \tag{4.15}$$

[8] Since $N = 2$, both selected harmonic annular Gaussian models include a second-order azimuthal Fourier correction.

the GP observation model for the residual field with sample-wise regularisation is then written as [160]

$$w_{\mathrm{res},i} \;=\; g(\tilde{\boldsymbol{\phi}}_i) + \varepsilon_i\,, \qquad g(\cdot) \sim \mathcal{GP}\left(0,\, k(\cdot,\, \cdot)\right), \qquad \varepsilon_i \sim \mathcal{N}(0,\, \alpha_i)\,, \tag{4.16}$$

where $\alpha_i = \max(\alpha_{\mathrm{scale}}\, \sigma_i^2,\, 10^{-8})$ is the sample-dependent observation noise variance with a minimum floor for numerical robustness. In the final selected model, the single-fan case uses $\alpha_{\mathrm{scale}} = 1.2$, while the four-fan selects $\alpha_{\mathrm{scale}} = 1.0$, indicating that the single-fan fit favours slightly stronger variance inflation.

The covariance between GP function values at two inputs is determined by the kernel function $k(\cdot,\, \cdot)$ [58]. For training inputs $\{\tilde{\boldsymbol{\phi}}_i\}_{i=1}^{M}$, the effective training covariance is

$$\mathbf{K}_y \;=\; \mathbf{K} + \mathbf{V}\,, \qquad K_{ij} \;=\; k(\tilde{\boldsymbol{\phi}}_i,\, \tilde{\boldsymbol{\phi}}_j), \qquad \mathbf{V} \;=\; \mathrm{diag}(\alpha_1,\, \ldots,\, \alpha_M). \tag{4.17}$$

Kernel type, including radial basis function (RBF) [59], Matérn kernels with $\nu \in \{3/2,\, 5/2\}$ [161], and rational quadratic [160], is selected using grouped cross-validation [162], [163]: samples are split by measurement plane $z_{\mathrm{fan},k}$, with three planes held out from training for validation, and the choice is made by minimising the validation WRMSE. This procedure selected a Matérn-3/2 kernel for single-fan and a Matérn-5/2 kernel for four-fan case [161], both with Automatic Relevance Determination (ARD) length scales [164] and an added white noise term to prevent overfitting by forcing exact interpolation [165]:

$$k(\tilde{\boldsymbol{\phi}},\, \tilde{\boldsymbol{\phi}}') \;=\; \sigma_c^2\, k_\nu(q) \;+\; \sigma_w^2 \delta_{\tilde{\boldsymbol{\phi}}\tilde{\boldsymbol{\phi}}'}\,, \qquad k_\nu(q) \;=\; \begin{cases} (1+\sqrt{3}\,q)\exp(-\sqrt{3}\,q)\,, & \nu = 3/2\,, \\ \left(1+\sqrt{5}\,q + 5q^2/3\right)\exp(-\sqrt{5}\,q)\,, & \nu = 5/2\,, \end{cases} \tag{4.18}$$

where $q = \sqrt{\sum_{d=1}^{4} (\tilde{\phi}_d - \tilde{\phi}'_d)^2/\ell_d^2}$ is ARD-scaled distance, $\delta_{\tilde{\boldsymbol{\phi}}\tilde{\boldsymbol{\phi}}'}$ equals 1 when $\tilde{\boldsymbol{\phi}} = \tilde{\boldsymbol{\phi}}'$ and 0 otherwise.

The kernel hyperparameters, $\boldsymbol{\theta} = (\sigma_c,\, \ell_1,\, \ell_2,\, \ell_3,\, \ell_4,\, \sigma_w) \equiv (\sigma_c,\, \ell_r,\, \ell_{\cos\theta},\, \ell_{\sin\theta},\, \ell_z,\, \sigma_w)$, are estimated by maximising the GP log marginal likelihood [160],

$$\boldsymbol{\theta}^* \;=\; \arg\max_{\boldsymbol{\theta}}\; \log p\, \left[-\frac{1}{2}\mathbf{w}_{\mathrm{res}}^\top\, \mathbf{K}_y^{-1}\mathbf{w}_{\mathrm{res}} - \frac{1}{2}\log|\mathbf{K}_y| - \frac{M}{2}\log(2\pi)\right], \tag{4.19}$$

with a multi-start procedure to reduce sensitivity to local minima [150].

After training, the GP surrogate is evaluated on each measured grid to produce maps of the predictive mean and standard deviation [160]. For a query input $\tilde{\boldsymbol{\phi}}_*$, the predictive residual mean and variance are

$$\mu_{\mathrm{res},*} \;=\; \mathbf{k}_*^\top \mathbf{K}_y^{-1}\mathbf{w}_{\mathrm{res}}\,, \qquad \sigma^2_{\mathrm{res},*} \;=\; k_{**} - \mathbf{k}_*^\top \mathbf{K}_y^{-1}\mathbf{k}_*\,, \tag{4.20}$$

where $\mathbf{w}_{\mathrm{res}} = [w_{\mathrm{res},1}, \ldots, w_{\mathrm{res},M}]^\top$, $\mathbf{k}_* = [k(\tilde{\boldsymbol{\phi}}_*,\, \tilde{\boldsymbol{\phi}}_1), \ldots, k(\tilde{\boldsymbol{\phi}}_*,\, \tilde{\boldsymbol{\phi}}_M)]^\top$, and $k_{**} = k(\tilde{\boldsymbol{\phi}}_*,\, \tilde{\boldsymbol{\phi}}_*)$.

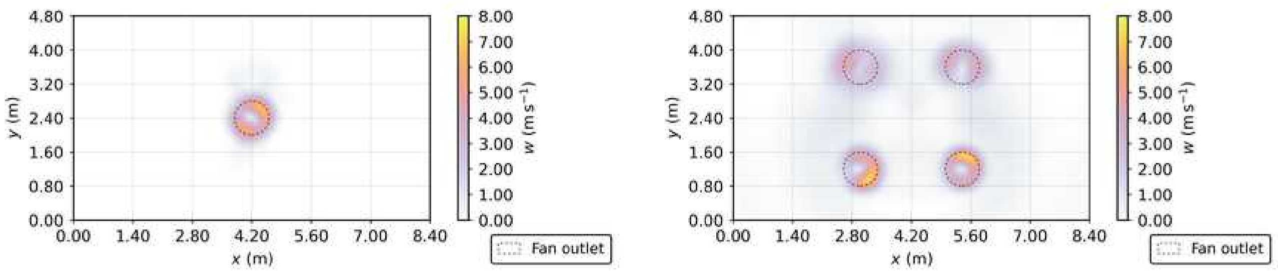
The final harmonic annular Gaussian model with GP residual correction [9] is then obtained by

$$w_{\mathrm{HAG-GP}}(x,\, y,\, z) \;=\; w_{\mathrm{HAG}} \;+\; \mu_{\mathrm{res}}\,, \qquad \sigma_{\mathrm{HAG-GP}}(x,\, y,\, z) \;=\; \sqrt{\sigma_{\mathrm{i}}^2 \;+\; \sigma_{\mathrm{res}}^2}\,. \tag{4.21}$$

The training procedure described above yields $\boldsymbol{\theta}^* = (2.29,\, 0.716,\, 0.659,\, 3.70,\, 0.371,\, 10^{-6})$ for the single-fan configuration and $\boldsymbol{\theta}^* = (0.789,\, 0.486,\, 100,\, 0.681,\, 1.96,\, 0.640)$ for the four-fan case. Relative to the single-fan case, the four-fan fit selects a smaller σ_c and a much larger white-noise level, σ_w, consistent with increased unresolved variability arising from multi-fan interactions [166]. The ARD length scales, $\{\ell_d\}_{d=1}^4$, further suggest that the residual field in the four-fan case is more radially localised, more anisotropic between the two angular feature directions, and shows little variation with height.

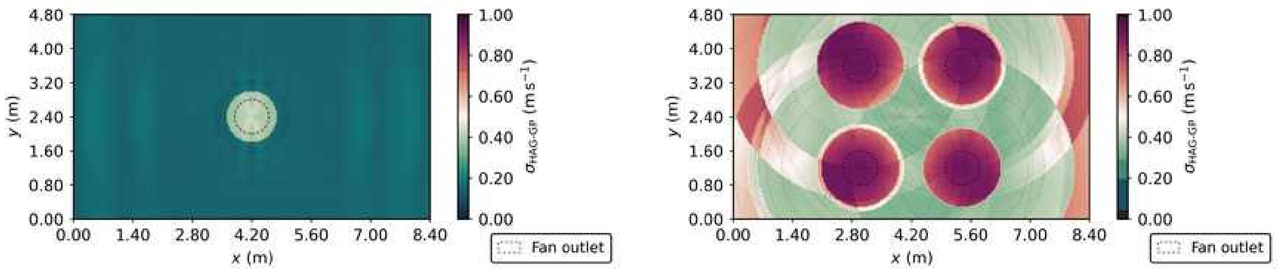
This observation is supported by the total fluctuation field shown in Fig. 4.9. The field is dominated by the empirical fluctuation term, σ_i, while a much smaller σ_{res} mainly reflects model uncertainty, and remains nearly height-invariant. In the single-fan case, $\sigma_{\mathrm{HAG-GP}}$ remains concentrated around the fan-centred region, whereas in the four-fan case it is more spatially distributed and consistently higher.

[9] For simplicity, this model is called the residual annular Gaussian process surrogate in the remainder of the thesis.



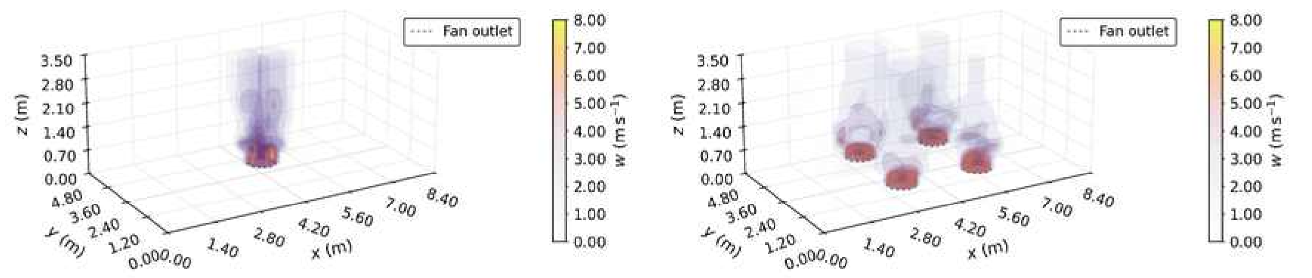
(a) single-fan case. **(b)** four-fan case.

Fig. 4.8. Fitted mean updraft map of the residual annular Gaussian process surrogate at a single measurement plane 0.35 m above the fan outlet plane.



(a) single-fan case. **(b)** four-fan case.

Fig. 4.9. Total fluctuation field, $\sigma_{\mathrm{HAG-GP}}$, of the residual annular Gaussian process surrogate at a measurement plane 0.35 m above the fan outlet plane. The grey dotted rings indicate the annular regions used to assign the empirical fluctuation level, σ_i, as described at the beginning of Section 4.3.



(a) single-fan case. **(b)** four-fan case.

Fig. 4.10. 3D rendering of the residual annular Gaussian process surrogate mean vertical velocity field.

4.4 Model Comparison and Selection

Alongside the harmonic annular Gaussian and residual annular GP surrogates, two simpler baselines are constructed: an axisymmetric annular Gaussian model, obtained by removing the Fourier series terms from (4.7), and an axisymmetric Gaussian plume, formed by further removing the ring radius term from (4.6). These four candidate updraft surrogates are evaluated using SAE and WRMSE metrics to compare their reconstruction quality and support the choice of surrogates used in controller development.

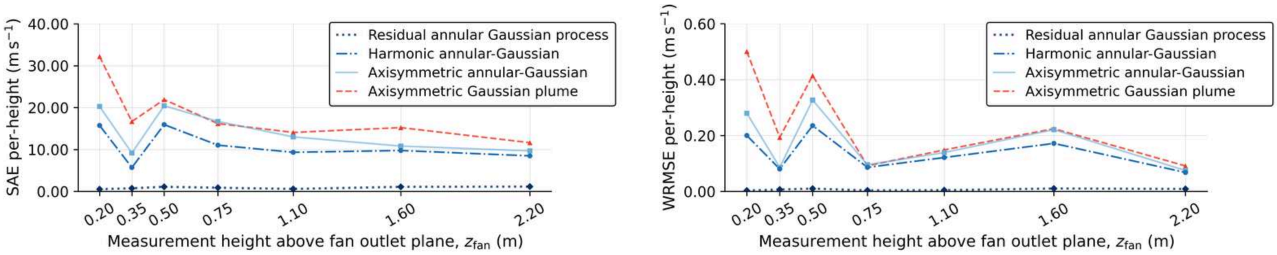
The fitted updraft maps for these two simpler baseline models are not shown in the main text for brevity. The complete set across all seven measurement planes is provided in Appendices D.2 to D.5.

For the single-fan configuration, Table 4.1 and Fig. 4.11 show a clear progression in fit quality from the Gaussian plume to the axisymmetric annular Gaussian, harmonic annular Gaussian, and residual annular GP surrogates, indicating that both the ring-shaped peak and azimuthal asymmetry are important features of the measured field. Once this dominant structure is captured, the remaining mismatch is weak, smooth, and spatially coherent, making it well-suited to residual GP correction.

For the four-fan case, the residual annular GP again yields the lowest SAE and WRMSE, but by a much smaller margin, as shown in Table 4.1 and Fig. 4.12. This is consistent with the stronger and less coherent residual field left by plume interaction, fan-to-fan asymmetry, and height-dependent merging.

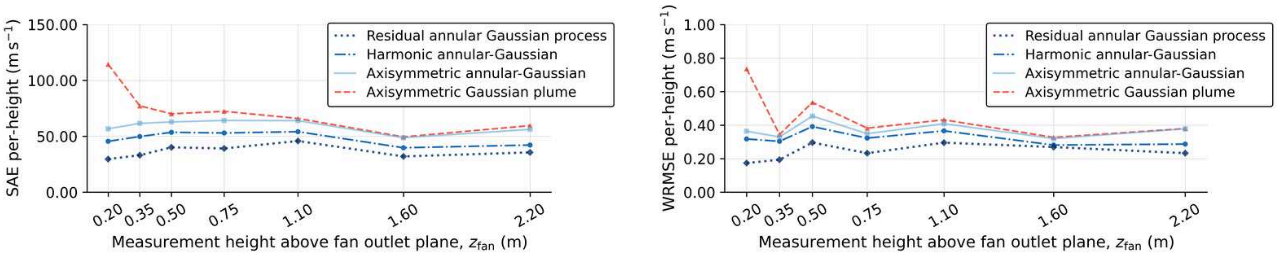
Table 4.1 Aggregate error metrics (SAE and WRMSE) for evaluated updraft models.

Setup	Error metric	Unit	Axisymmetric Gaussian plume	Axisymmetric annular Gaussian	Harmonic annular Gaussian	Residual annular Gaussian process
Single-fan	SAE	$\mathrm{m\,s^{-1}}$	128	100	76.1	6.14
	WRMSE	$\mathrm{m\,s^{-1}}$	0.210	0.152	0.120	0.00607
Four-fan	SAE	$\mathrm{m\,s^{-1}}$	509	414	338	255
	WRMSE	$\mathrm{m\,s^{-1}}$	0.419	0.367	0.321	0.241



(a) sum of absolute errors (SAE). **(b)** weighted root mean square error (WRMSE).

Fig. 4.11. Per-height error metrics for the evaluated updraft models (single fan).



(a) sum of absolute errors (SAE). **(b)** weighted root mean square error (WRMSE).

Fig. 4.12. Per-height error metrics for the evaluated updraft models (four fans).

Although the residual annular GP gives the most accurate reconstruction of the measured mean field, it remains an in-sample reconstruction rather than a complete description of the underlying unsteady flow. This mainly arises from two limitations. First, the physical updraft is time-varying, while the reconstructed mean field is fitted to time-averaged measurements at a finite set of spatial points. The residual correction is flexible enough to reduce interpolation error, but an improved fit does not necessarily imply better physical generalisation, it may also reflect overfitting to the available data [160], [165]. Second, temporal fluctuation is sampled more sparsely. The local fluctuation levels used in the model are estimated from short time-series records at only six representative locations, and then mapped across the rest of the field by blending in (4.3). Therefore, the resulting total fluctuation field in (4.21) is an approximate simulation uncertainty map, and may differ substantially from reality.

Instead of collecting additional data or adopting more accurate measurement methods to refine a single updraft model, the subsequent controller study in Chapter 6 adopts a more practical simulation strategy based on the principle of domain randomisation in robotics sim-to-real transfer [10], [12], [15]. The residual annular GP is selected as the most detailed measured surrogate, while its strength, spatial scale, centre location, number of active fans, and uncertainty level are randomised during simulation. According, controller design and validation are carried out across imperfect but physically related updraft families, reducing dependence on a single fitted reconstruction that may overfit the available measurements, while allowing reality to be treated as another variation of the simulated cases.

5 Glider Design and Optimisation

With the experimental apparatus, system architecture, and fan-generated updraft established in Chapters 3 and 4, the next requirement is the physical vehicle that must operate within this controlled setting. This chapter introduces the glider used as the experimental control plant. Its design supports one of the thesis objectives: a reproducible and extensible workflow for indoor fixed-wing sim-to-real experiments.

5.1 Design Requirements and Sizing Methodology

The glider needs to operate within the confined usable motion capture volume of the Flight Arena defined in Section 3.1, carry the selected onboard components of the control interface shown in Fig. 3.4, and remain controllable when flying through the aggressive local updrafts characterised in the previous chapter. Together, these requirements framed the task as a sizing problem: the airframe should have a low sink rate, moderate roll response, adequate longitudinal trim authority, sufficient directional stability, enough structural stiffness to survive failed launches, wall contact, and abnormal landings, and be repairable after minor damage. It also needed to be modular enough for future studies or applications to adapt the payload, airframe components, or operating assumptions without redesigning the airframe.

Building on the small fixed-wing glider design and construction review in Section 2.3, the material and modelling choices were made to keep the aircraft light, stiff, repairable, and easy to modify. Small glider construction commonly falls between plywood structures with thin covering and carbon-fibre tube airframes with foam lifting surfaces. The former can be very light, allow carefully shaped airfoils, but is more vulnerable to impact damage, much difficult to repair, and harder to modularise. Hence, the present design adopted the latter architecture: a 4 mm outer diameter, 2 mm inner diameter unidirectional (UD) carbon-fibre tube [167] was used as the fuselage, and 4 mm outer-diameter, 3 mm inner-diameter tubes were added as wing spars. Depron [67], a lightweight extruded polystyrene foam sheet, was used with 6 mm and 3 mm thicknesses for the wing and tails respectively. The lifting surface joints and avionics supports were 3D-printed from PLA-CF [168] and PAHT-CF [169]. Detailed 3D-printed parts design, adhesive reinforcement, and manufacturing are discussed in Section 5.3.

The glider sizing workflow was designed to support rapid iteration through a design-manufacture-measure loop, allowing manufacturing constraints and early flight test observations that were not fully captured in the initial design considerations to feed back into later airframe iterations. AeroSandbox [77] was used as the open-source Python MDO framework for this sizing study. It enabled the geometry, mass model, trim conditions, stability checks, simple stiffness estimates, and handling constraints to be coupled in a single differentiable constrained optimisation problem. Aerodynamic loads and coefficients were evaluated using AeroSandbox's AeroBuildup analysis [170], a low-order component buildup method that estimates aircraft forces, moments, induced drag, neutral point, and stability derivatives. Through AeroSandbox's backend, CasADi [171] supplied automatic differentiation for the coupled model, and Interior Point OPTimiser (IPOPT) [172] solved the resulting constrained nonlinear programme.

The preliminary sizing model implemented in AeroSandbox was intentionally simplified so that the main design parameters could be selected quickly during the iterative workflow. The lifting surfaces were modelled using thin, symmetric NACA 0002 aerofoils as finite thickness surrogates for the nearly flat Depron lifting surfaces, avoiding numerical difficulties associated with zero thickness flat-plate models while preserving the intended thin plate behaviour at low Reynolds number [69], [78], [173]. The mass model combined the materials discussed above, the onboard avionics selected in Section 3.2.2, and an additional glue allowance to account for adhesive mass and small manufacturing variations.

5.2 Optimisation Formulation and Design Selection

With the materials selected and a simplified glider model defined, the sizing optimisation problem could be formulated to select the main geometric and trim parameters. Robustness and sensitivity checks were then used to examine the design response to geometric parameter changes during manufacturing.

5.2.1 Sizing Problem Formulation

The sizing problem first separated fixed platform choices from optimisation variables, holding weakly constrained secondary parameters constant so that the optimiser could focus on dominant design drivers.

All lifting surfaces were kept rectangular, unswept, and uncambered since rectangular flat plates are easier to manufacture and repair, while wing sweep is mainly used to manage high-speed compressibility effects [72]. The wing dihedral angle was fixed at $\Gamma = 10°$ as a practical choice for passive lateral stability [174]. The horizontal and vertical tails used fixed aspect ratios, $AR_H = 4$ and $AR_V = 2$, consistent with common preliminary empennage sizing ranges [71]. Control surface geometry was also fixed: the ailerons occupied $30\% - 85\%$ of the wing semispan with a 30% chord fraction, the elevator used a 30% horizontal tail chord fraction, and the rudder used a 35% vertical tail chord fraction, to preserve a manufacturable actuator layout and sufficient control authority for low-speed flight[61].

The optimisation problem was then formulated around the variables that directly affected flight performance, airframe layout, and trim. The decision vector contained the main sizing and trim variables,

$$z = [\alpha_{\text{trim}},\, \delta_{a,\text{trim}},\, \delta_{e,\text{trim}},\, \delta_{r,\text{trim}},\, b_y,\, c,\, l_t,\, b_H,\, h_V]^\top \in \mathcal{B}\,, \tag{5.1}$$

where $b_y = b\cos\Gamma$ is the projected wing span for canted span b, c is the wing chord, l_t is the tail arm, b_H is the horizontal tail span, h_V is the vertical tail height, and $\mathcal{B}$ is the bound set on these variables.

In the optimiser, each candidate airframe was evaluated under the same representative flight conditions, chosen to reflect operation within the confined usable volume of the indoor Flight Arena. The level-flight sizing point used $6.5\,\mathrm{m\,s^{-1}}$, while a lower-speed turning check at $4.0\,\mathrm{m\,s^{-1}}$ evaluated whether the aircraft could generate sufficient roll response within the available space. This check used a $50°$ bank target, a $0.2\,\mathrm{s}$ roll entry time window, and a $0.2\,\mathrm{m}$ wall clearance allowance.

The objective function then balanced the main design trade-offs as

$$\begin{aligned} J(z) \;=\; & \frac{V_{\text{sink}}}{1.00\,\mathrm{m\,s^{-1}}} \;+\; 0.05\,\frac{m}{0.10\,\mathrm{kg}} \;+\; 0.25\,\frac{\tau_{\text{roll}}}{0.45\,\mathrm{s}} \\ & \;+\; 2.15\,P_W \;+\; 0.08\,P_H \;+\; 0.01\,\frac{\delta_{e,\text{trim}}^2 + 0.30\delta_{r,\text{trim}}^2 + 0.15\delta_{a,\text{trim}}^2}{(10.0°)^2}. \end{aligned} \tag{5.2}$$

Here, P_W and P_H are the dimensionless tip deflection penalties for the wing and horizontal tail. Each half of the lifting surface was treated as a root-fixed cantilever beam under a distributed load [175]:

$$P \;=\; \frac{\mathrm{sp}_\kappa(\delta - \delta_{\text{allow}})}{\delta_{\text{allow}}}\,, \qquad \mathrm{sp}_\kappa(x) = \kappa^{-1}\ln(1 + \exp(\kappa x))\,, \qquad \delta \;=\; \frac{\lambda \ell^4}{8EI}\,, \qquad \delta_{\text{allow}} = 0.08\ell\,. \tag{5.3}$$

The roll response term, τ_{roll}, was estimated from the standard linear roll-damping relation [33],

$$\tau_{\text{roll}} \;=\; \frac{2I_{xx}V_s}{q_s S b_y^2 |C_{l_p}|}, \qquad q_s \;=\; \frac{1}{2}\rho V_s^2. \tag{5.4}$$

Accordingly, the complete sizing problem was written as

$$z^\star \;=\; \arg\min_{z\in\mathcal{B}}\, J(z;\,\theta,\,\xi_s) \qquad \text{subject to} \quad C_k(z;\,\theta,\,\xi_s) \le 0\,, \quad k = 1,\,\ldots,\,N_C\,, \tag{5.5}$$

where θ collects the fixed material, actuator, avionics, and build assumptions introduced in Sections 3.2.2 and above, ξ_s contains the optimiser flight condition, and C_k are the feasibility constraints chosen for the indoor flight task in this thesis, covering lift balance, pitching moment trim, static margin, tail volume, control authority, wing loading, minimum Reynolds number, roll damping, and structural limits.

The full numerical setup is not expanded here for brevity. Appendix E.1 gives the complete bounds, feasibility limits, objective weights, solver settings, and workflow algorithms.

5.2.2 Uncertainty-Aware Design Selection

A preliminary glider design can appear optimal at one sizing point, but still be sensitive to local solver convergence or small build variations [73], [176]. To reduce this risk, the nonlinear problem was solved from multiple initial guesses to expose different feasible local designs [150]. Near-identical solutions were removed, and the remaining candidates were evaluated under 100 uncertainty scenarios [177]. These scenarios separated possible manufacturing errors from the local updraft conditions experienced by the glider, based on the flow characterisation in Chapter 4. The former varied mass growth up to 10%, longitudinal CG shift by $\pm 0.06\,\mathrm{MAC}$, incidence by $\pm 2°$, control effectiveness by $0.85-1.00$, control surface zero offset by $\pm 3°$, inertia by $0.90-1.10$, wing and horizontal tail stiffness by $0.80-1.30$, and drag by $1.00-1.25$, while the latter varied the vertical-flow speed at optimisation point by $\pm 0.30\,\mathrm{m\,s^{-1}}$.

Accordingly, the final glider design was selected from the uncertainty evaluation metrics using a lexicographic ranking rule, where lower-priority criteria were considered only when candidates had similar higher-priority performance [178], [179]. In this problem, candidates were ranked first by the fraction of uncertainty cases that remained feasible, then by the mean sink rate over the worst successful cases, and finally by the original sizing objective.

The selected candidate was then carried into the design-manufacture-measure loop discussed in Section 5.1. Five iterations were completed before the final design was selected. During this loop, the sizing model was updated to reflect practical manufacturing constraints, including the tip-deflection penalties in (5.3), revised avionics mass and placement, component locations that avoided mutual interference, and iterative refinement of the 3D-printed mounting design discussed later in Section 5.3.

The final selected preliminary design is summarised in Table 5.1. It used the full allowable projected span, $b_y = 0.750\,\mathrm{m}$, since a larger span reduced sink rate and improved roll response without violating the static margin or mass constraints. The corresponding uncertainty evaluation is shown in Fig. 5.1, where the design remained feasible in all 100 uncertainty scenarios. Manufacturing perturbations mainly increased sink rate and roll response, while updraft-only cases primarily reduced the trim angle of attack margin. The compound cases combined these effects and produced the tightest trim margin.

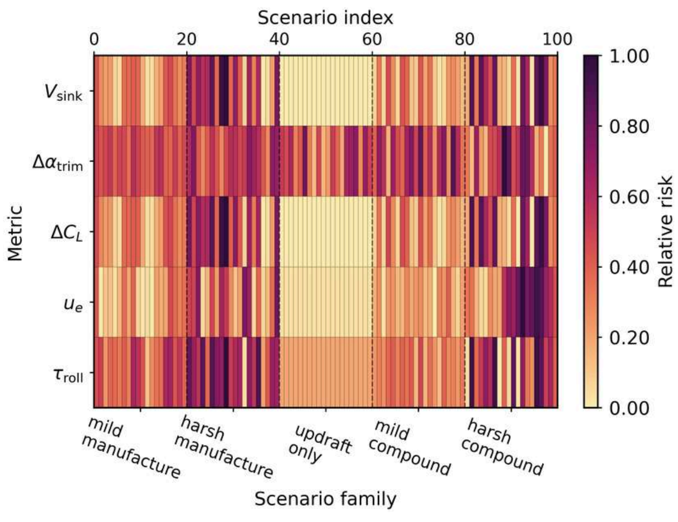


Fig. 5.1. Uncertainty scenario evaluation map for the selected candidate in fifth iteration. The scenario families separate manufacturing perturbations, updraft perturbations, and compound cases where both are applied together, "mild" and "harsh" denote the lower and higher uncertainty ranges.

Table 5.1 Final selected glider design.

Quantity	**Symbol**	**Unit**	**Value**
Projected wing span	b_y	m	0.750
Canted wing span	b	m	0.762
Wing chord	c	m	0.165
Tail arm	l_t	m	0.400
Boom length	l_b	m	0.564
Horizontal tail span	b_H	m	0.366
Vertical tail height	h_V	m	0.119
Design mass	m	g	117.5
Wing loading	$W\,S^{-1}$	$\mathrm{N\,m^{-2}}$	9.20
Lift-to-drag ratio	$L\,D^{-1}$	-	9.70
Static margin	SM	%MAC	10.0
Sizing sink rate	V_{sink}	$\mathrm{m\,s^{-1}}$	0.670
Worst sink rate	$V_{\mathrm{sink}}^{\mathrm{tail}}$	$\mathrm{m\,s^{-1}}$	0.852

The iteration history, geometry, and mass breakdown diagrams are provided in Appendix E.2, while the detailed optimiser inputs and selected design properties are provided in Appendix E.3.

5.2.3 Sensitivity Analysis

To separate the effect of individual geometry choices from the combined uncertainty cases above, local normalised sensitivities were evaluated about the selected design [170], [180]. For an output quantity q and design parameter p, the sensitivity at the selected design, denoted by $(\cdot)^\star$, is written as

$$\bar{S}_{q,p} = \frac{p^\star}{|q^\star|} \left.\frac{\partial q}{\partial p}\right|_\star . \tag{5.6}$$

Representative local sensitivities are summarised in Table 5.2. Since small changes in lifting surface geometry significantly affect glider sink rate, roll response, and static margin, the glider properties must be remeasured after manufacturing before being used to build the control-oriented model in Chapter 6.

Table 5.2 Representative normalised local sensitivities to geometry increases about the selected design.

Perturbed quantity	Symbol	V_{sink}	m	τ_{roll}	SM	Dominant effect
Wing span	b	-0.535	0.305	-1.334	-8.377	Lower sink and faster roll response, but reduced static margin
Wing chord	c	0.001	0.231	-0.417	-6.723	Mainly increases mass and reduces static margin
Tail arm	l_t	0.017	0.048	0.113	3.177	Improves longitudinal stability, with small mass and roll penalties
Horizontal tail span	b_H	0.046	0.093	-0.189	8.608	Strongly improves static margin but adds mass
Vertical tail height	h_V	0.038	0.019	0.001	-0.621	Weak effect on these longitudinal and roll metrics

5.3 Detailed Design and Manufacture

The fifth iteration sizing result in Table 5.1 defined the main glider geometry, while the material choices in Section 5.1 set the basic structural architecture. To turn this preliminary design into a physical glider, the mounting structure, landing protection, and control linkages still needed to be designed in detail, bringing together the lifting surfaces, wing spar, fuselage tube, Vicon reflective markers, onboard receiver and servos from the selected control interface in Fig. 3.4 into one airframe.

5.3.1 Modular Hardware Interface Design

As shown in Fig. 5.3, the main hardware interfaces were built as 3D-printed modules that slid onto the carbon-fibre fuselage tube and were fixed with removable adhesive, so their positions could be adjusted during assembly or later modification. The upper part of the centre module provided the wing and wing spar mounting points, with chamfered and tapered supports used to improve local stiffness while avoiding concentrated loads at sharp corners. The lower plate carried the onboard avionics and was perforated to reduce mass, provide wiring access, and leave sufficient space for future avionics and payload expansion. The landing gear attached to this centre module acted as the primary impact-absorbing structure during normal landings, protecting the avionics mounted below the centre plate. The tail module fixed the horizontal and vertical tails to the fuselage rod, while the tail gear acted as a rear skid.

To balance printability, stiffness, mass, and part function, PLA-CF [168] was used for the geometrically complex centre module and landing gear, where reliable printing and sufficient local stiffness were prioritised. PAHT-CF [169] was used for the smaller tail mount and tail skid, where the simpler geometry made more demanding filament practical and its lower density helped reduce rear-end mass.

The glider material properties and hardware interfaces print settings are listed in Appendix E.4.

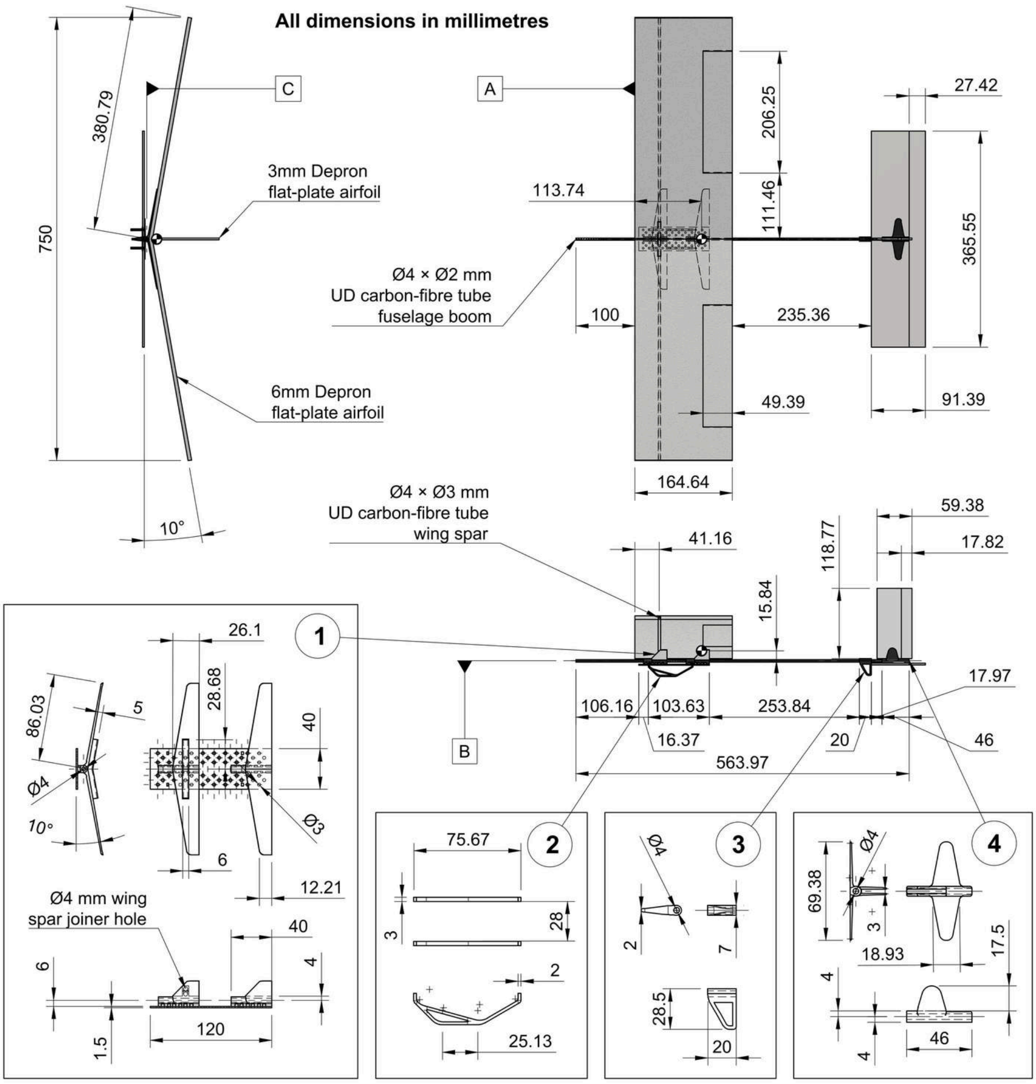


Fig. 5.2. General arrangement and key dimensions of the fifth iteration glider assembly.

5.3.2 Manufacture and Assembly

With the hardware interface modules designed and 3D-printed, the Depron [67] lifting surfaces were bonded to them using epoxy [181]. Cross-weave reinforced transparent filament tape [182] was then applied along the spar, hinge lines, and tail joints to improve local stiffness and damage tolerance. Hot-melt adhesive was used to attach the Vicon reflective markers, allowing them to be quickly removed and reattached when changing the tracking configuration between Fig. 3.5 and 3.6 for different tests.

The control surfaces were cut directly into the lifting surfaces. Along each hinge line, the moving edge was bevelled so that the surface could rotate freely, and transparent tape was used as a simple, flexible hinge. The ailerons were driven by short 0.8 mm diameter 304 stainless-steel wire [183] pushrods, since their servos were directly mounted on the wing. For the elevator and rudder, the servos were placed on the centre module to help control the CG position, so longer pushrods were required. These longer linkages used 1.5 mm diameter carbon-fibre rods [167] as the main members, with the same stainless-steel wire bonded to each end using hot-melt adhesive and covered with heat-shrink tubing [10].

[10] This linkage layout was adapted from an online RC aircraft construction tutorial, available at https://www.bilibili.com/video/BV11XLUzKEpc.

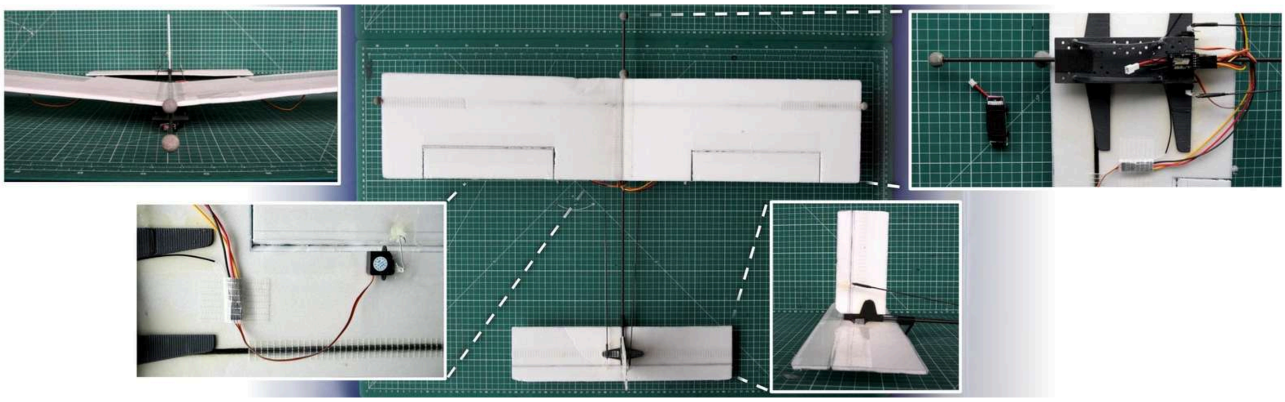


Fig. 5.3. Manufactured fifth iteration glider and key assembly details.

5.3.3 Manufacturing Deviations and Flight Configuration

After assembly, the glider was first checked through open-loop launches, which showed a pitch-up stall tendency. Hence, a 15 g ballast was added near the nose to move the longitudinal CG forward.

As shown in Table 5.3, the geometry of the manufactured glider remained close to the preliminary sizing output, but the measured mass and longitudinal CG changed significantly. This type of variation was already considered in the uncertainty-aware evaluation in Section 5.2.2, but the measured mass growth and CG shift were larger than the assumed limits even before adding ballast. Such a mismatch reveals the main weakness of the sizing workflow in this chapter, even with the design-manufacture-measure loop: it largely underestimated the practical mass and placement of wiring and soldered joints, which were only installed during final assembly, while leaving too little margin for potential ballast.

Table 5.3 Measured glider configurations, with percentage changes relative to the optimised value.

Parameter	Symbol	Unit	Optimised	Manufactured	Δ	Ballasted	Δ
Mass	m	g	117.5	133.6	+13.6%	148.6	+26.4%
Wing span	b	m	0.762	0.764	+0.3%	0.764	+0.3%
Wing chord	c	m	0.165	0.165	0%	0.165	0%
Dihedral	Γ	deg	10.0	9.28	-7.2%	9.28	-7.2%
Boom length	l	m	0.564	0.568	+0.7%	0.568	+0.7%
Horizontal tail span	b_H	m	0.366	0.364	-0.5%	0.364	-0.5%
Vertical tail height	h_V	m	0.119	0.119	0%	0.119	0%
Wing loading	$W\,S^{-1}$	$\mathrm{N\,m^{-2}}$	9.20	10.4	+13.0%	11.6	+26.1%
Longitudinal CG	$x_{\mathrm{CG}}\,c^{-1}$	%MAC	69.1	76.4	+10.6%	63.9	-7.5%

Nevertheless, after ballast was added, the manufactured glider met the experimental requirements. During the subsequent flight testing reported in Chapter 6 and 7, it completed 1042 launch attempts across still air and strong updraft conditions, providing a stable platform for real-world controller evaluation. Even after occasional hard wall or floor impacts, damage was limited to the wing edges and could be repaired quickly with transparent tape, leaving no noticeable change in flight behaviour.

The completed glider design and manufacture, together with the sensing and command architecture, latency quantification, and environment design and characterisation established in the preceding chapters, now form a repeatable, reproducible, and extensible platform for indoor fixed-wing sim-to-real experiments. Building on this platform, the thesis can proceed to develop a lift-aware guidance and control method for flying the glider through uncertain indoor updrafts.

6 Flying Through Uncertainty with Viable Manoeuvre Primitives

This chapter returns to the control problem introduced in Chapter 1: how should a small fixed-wing aircraft operate near terrain, structures, or other flow-shaping boundaries [1], [3], [5] when the local vertical airflow can be useful or harmful depending on the flight condition [6], [7], and how can the aircraft explore and exploit that flow when it repeatedly flies through the same operating region?

6.1 Problem Formulation

In outdoor flight, sensing delays, flow structure, and safety constraints vary together over space and time, making the control problem difficult to study. Therefore, a controlled laboratory setting is needed, so the effects of these coupled factors can be reproduced, isolated, measured, and examined systematically.

The preceding chapters provide this setting in the Flight Arena, allowing the broader control problem to be studied as a repeatable flight segment: a hand-launched glider enters the bounded volume, meets fan-generated updrafts, and aims to safely reach the opposite side while using any lift along its path.

6.1.1 Mission Definition and Launch Gate

As discussed in Section 3.1, the Flight Arena defines an arena frame, with x along the intended flight direction, y lateral, and z upward. The body frame follows the flight dynamics convention shown in Fig. 3.5, with its origin taken at the glider CG. Expanding the 15-state vector in (3.2) gives

$$\mathbf{x} = [x,\ y,\ z,\ \phi,\ \theta,\ \psi,\ u_b,\ v_b,\ w_b,\ p_b,\ q_b,\ r_b,\ \delta_a,\ \delta_e,\ \delta_r]^\top . \tag{6.1}$$

The first twelve components are obtained from the processed Vicon motion capture stream [123], while the final three are propagated from the command history. Both procedures are described in Section 3.3.

Section 3.1 also defines the smaller flight volume used for controller evaluation, selected to maintain visibility of the reflective markers attached to the glider by the Vicon motion capture system:

$$\mathcal{X}_{\text{safe}} = \{\mathbf{x} : x \in [1.2,\ 6.6]\ \text{m}\,, \quad y \in [0.0,\ 4.4]\ \text{m}\,, \quad z \in [0.4,\ 3.5]\ \text{m}\}\,. \tag{6.2}$$

To ensure that the glider enters this flight volume within a repeatable range, while still allowing the unavoidable variation of a hand launch, the launch gate is defined as the admissible initial condition set:

$$\begin{aligned} \mathcal{X}_{\text{launch}} = \big\{\mathbf{x} : \ & x \in [1.2,\ 1.4]\ \text{m}\,, \quad y \in [1.8,\ 2.2]\ \text{m}\,, \quad z \in [1.3,\ 1.8]\ \text{m}\,, \\ & |\phi| \le 20\ \text{deg}\,, \quad -10\ \text{deg} \le \theta \le 20\ \text{deg}\,, \quad |\psi| \le 20\ \text{deg}\,, \\ & 4.0 \le u_b \le 8.0\ \text{m}\,\text{s}^{-1}\,, \quad |v_b| \le 1.5\ \text{m}\,\text{s}^{-1}\,, \quad |w_b| \le 0.9\ \text{m}\,\text{s}^{-1}\,, \\ & |p_b| \le 1.2\ \text{rad}\,\text{s}^{-1}\,, \quad |q_b| \le 1.2\ \text{rad}\,\text{s}^{-1}\,, \quad |r_b| \le 1.8\ \text{rad}\,\text{s}^{-1}\big\}\,. \end{aligned} \tag{6.3}$$

Because the glider may cross the launch gate between two processed motion capture frames, the launch trigger is interpolated to the plane $x = 1.3$ m and accepted only after two consecutive frames satisfy $\mathcal{X}_{\text{launch}}$. This approach reduces sensitivity to both frame timing and single-frame measurement noise.

As illustrated in Fig. 6.1, mission success is defined as the glider reaching the downstream side of the admissible flight volume $\mathcal{X}_{\text{safe}}$ through the intended exit face. The mission failure occurs if the glider hits the floor, ceiling, or lateral walls.

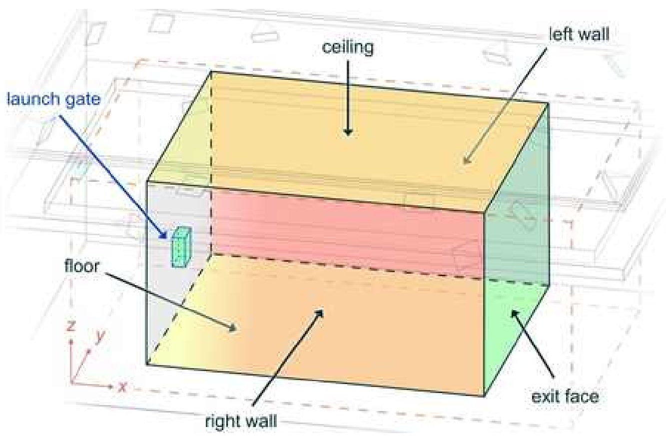


Fig. 6.1. Mission geometry.

The local updrafts used in this mission is generated by Clarke CAMAX30 axial drum fans [136], as characterised in Chapter 4. The mission-specific fan placements are introduced later in this chapter.

6.1.2 Controller Design Objective

The main controller design difficulty is guiding the glider through the uncertain updrafts while completing the mission. The glider must balance the useful lift provided by the updraft against the risk of crossing the safety boundary. This requirement can be expressed as a constrained finite-horizon control problem:

$$\begin{aligned} \underset{\pi}{\text{maximise}} \quad & \mathbb{E}_{\pi}\left[H(\mathbf{x}(t_f)) - H(\mathbf{x}(t_0)) + \lambda_\ell T_\ell\right] \\ \text{subject to} \quad & \mathbf{x}(t) \in \mathcal{X}_{\text{safe}}\,, \quad \boldsymbol{\nu}(t) \in \mathcal{U}\,, \quad t \in [t_0,\, t_f]\,. \end{aligned} \tag{6.4}$$

where $H = z + \|\mathbf{v}_b\|^2/(2g)$ is glider's specific energy in metres, T_ℓ is the time spent in useful lift, λ_ℓ sets weight on lift exploitation, $\boldsymbol{\nu} = [\nu_a,\, \nu_e,\, \nu_r]^\top$ is the executable aggregate command for aileron, elevator, and rudder, and $\mathcal{U}$ is the allowable command set defined by the command interface in Section 3.3.2.

6.2 Control-Oriented Glider Model

The model used in this chapter must be accurate enough to support local feedback design and expose the effect of uncertain updrafts, while remaining simple enough to run repeatedly across many candidate controllers and environment settings. These requirements lead to a control-oriented 6-DoF model with panelwise aerodynamic loading and simple residual calibration through system identification.

6.2.1 Glider Dynamics

The continuous-time rigid-body and actuator model uses the 15-state notation in (6.1). It is written as

$$\dot{\mathbf{p}} \;=\; \mathbf{D}_z\,\mathbf{C}_{wb}(\boldsymbol{\eta})\,\mathbf{v}_b\,, \tag{6.5}$$

$$\dot{\boldsymbol{\eta}} \;=\; \mathbf{T}(\boldsymbol{\eta})\,\boldsymbol{\omega}_b\,, \tag{6.6}$$

$$\dot{\boldsymbol{\delta}} \;=\; \boldsymbol{\tau}_\delta^{-1}\left(\boldsymbol{\delta}^{\text{cmd}} - \boldsymbol{\delta}\right), \tag{6.7}$$

$$\dot{\mathbf{v}}_b \;=\; \frac{1}{m}\mathbf{F}_b(\mathbf{x},\,\mathbf{W}) \;+\; \mathbf{C}_{wb}^\top(\boldsymbol{\eta})\,\mathbf{D}_z\,\mathbf{g}_w \;-\; \boldsymbol{\omega}_b \times \mathbf{v}_b\,, \tag{6.8}$$

$$\dot{\boldsymbol{\omega}}_b \;=\; \mathbf{I}_b^{-1}\left[\mathbf{M}_b(\mathbf{x},\,\mathbf{W}) \;-\; \boldsymbol{\omega}_b \times \mathbf{I}_b\boldsymbol{\omega}_b\right], \tag{6.9}$$

where $\mathbf{F}_b$ and $\mathbf{M}_b$ are the total aerodynamic force and moment in body axes, and $\mathbf{g}_w = [0,\,0,\,-g]^\top$ is the gravitational acceleration vector. The matrices $\mathbf{C}_{wb}(\boldsymbol{\eta})$ and $\mathbf{D}_z = \operatorname{diag}(1,\,1,\,-1)$ follow the Vicon-based state processing convention detailed in Appendix B.2. The measured mass m and body-axis inertia matrix $\mathbf{I}_b$ come from the manufactured glider mass properties in Appendix E.5. The commanded control surface target $\boldsymbol{\delta}^{\text{cmd}}$ and response time constant $\boldsymbol{\tau}_\delta$ follow the model introduced in (3.4) and (3.5).

6.2.2 Panelwise Aerodynamic Loading

The updraft can vary significantly across the lifting surfaces, producing uneven aerodynamic loading and changing the net force and moment on the glider. To represent this effect, the control-oriented glider model divides the lifting surfaces into 24 aerodynamic strips: 12 on the wing, 8 on the horizontal tail, and 4 on the vertical tail. Each strip is approximated as a local 2D flat-plate section, so lift and drag are computed from the part of the relative airflow that crosses the strip chordwise [66], [82].

For a representative strip indexed by i, the model stores its body frame location $\mathbf{r}_{i,b}$, area S_i, chord c_i, span direction $\mathbf{s}_{i,b}$, surface normal $\mathbf{n}_{i,b}$, aspect ratio Æ_i, zero-lift offset $\alpha_{0,i}$, profile drag coefficient $C_{D0,i}$, and control effectiveness vector $\mathbf{m}_i$. The local airflow is evaluated at the centre of the strip:

$$\mathbf{V}_{i,b} \;=\; \mathbf{v}_b \,+\, \boldsymbol{\omega}_b \times \mathbf{r}_{i,b} \;-\; \mathbf{C}_{wb}^\top\,\mathbf{D}_z\,\mathbf{W}(\mathbf{p}_i,\,t)\,, \qquad \mathbf{p}_i \;=\; \mathbf{p} \,+\, \mathbf{D}_z\,\mathbf{C}_{wb}\,\mathbf{r}_{i,b}\,. \tag{6.10}$$

where $\mathbf{p}_i$ is the corresponding centre position in the arena frame, and $\mathbf{W}(\mathbf{p}_i, t)$ is the arena frame wind velocity at that position. The strip dynamic pressure and effective angle of attack are then computed as

$$q_i \;=\; \frac{1}{2}\rho\,\|\mathbf{V}_{i,b}^\perp\|^2, \qquad \alpha_{i,\text{eff}} \;=\; \alpha_i - \alpha_{0,i} \,+\, \lambda_i\,\mathbf{m}_i^\top\boldsymbol{\delta}\,, \qquad \mathbf{V}_{i,b}^\perp \;=\; \mathbf{V}_{i,b} \;-\; \left(\mathbf{V}_{i,b}^\top\,\mathbf{s}_{i,b}\right)\mathbf{s}_{i,b}\,, \tag{6.11}$$

where $\mathbf{V}^{\perp}_{i,b}$ is the relative airflow after removing the spanwise component, which does not define the local section angle of attack in strip approximation, and $\lambda_i\, \mathbf{m}_i^{\top}\boldsymbol{\delta}$ is the control-induced local incidence change.

Using these local quantities, the attached flow lift slope is corrected for finite aspect ratio [184]:

$$a_i \;=\; \frac{2\pi \mathit{AR}_i}{\mathit{AR}_i + 2}\,, \qquad C^{\text{att}}_{L,i} \;=\; a_i\, \alpha_{i,\text{eff}}\,, \qquad C^{\text{att}}_{D,i} \;=\; C_{D0,i} \,+\, \frac{(C^{\text{att}}_{L,i})^2}{\pi e_i \mathit{AR}_i}\,. \tag{6.12}$$

This attached flow model is reliable only at moderate angles of attack. To provide a smooth transition to post-stall behaviour, the attached coefficients are blended into a flat-plate post-stall approximation [66]:

$$C_{L,i} \;=\; (1-\sigma_i)\, C^{\text{att}}_{L,i} \,+\, \sigma_i C^{\text{post}}_{L,i}\,, \qquad C^{\text{post}}_{L,i} \;=\; 2\sin\alpha_{i,\text{eff}}\cos\alpha_{i,\text{eff}}\,, \tag{6.13}$$

$$C_{D,i} \;=\; (1-\sigma_i)\, C^{\text{att}}_{D,i} \,+\, \sigma_i C^{\text{post}}_{D,i}\,, \qquad C^{\text{post}}_{D,i} \;=\; C_{D0,i} \,+\, k_D \sin^2\alpha_{i,\text{eff}}\,, \tag{6.14}$$

where the smooth activation $\sigma_i \in [0,\,1]$ is a function of $|\alpha_{i,\text{eff}}|$. Values near $\sigma_i = 0$ correspond to attached flow, values near $\sigma_i = 1$ represent post-stall flow, and intermediate values define the transition regime.

Finally, the total aerodynamic force and moment are obtained by summing the strip contributions and adding compact residual corrections identified in the following section:

$$\mathbf{F}_b \;=\; \sum_{i=1}^{N_s} \mathbf{F}_{i,b} \,+\, \mathbf{F}_{\text{res}}(\mathbf{x},\boldsymbol{\delta})\,, \qquad \mathbf{F}_{i,b} \;=\; q_i S_i \left(C_{L,i}\, \boldsymbol{\ell}_{i,b} - C_{D,i}\, \mathbf{d}_{i,b}\right), \tag{6.15}$$

$$\mathbf{M}_b \;=\; \sum_{i=1}^{N_s} \mathbf{M}_{i,b} \,+\, \mathbf{M}_{\text{res}}(\mathbf{x},\boldsymbol{\delta})\,, \qquad \mathbf{M}_{i,b} \;=\; \mathbf{r}_{i,b} \times \mathbf{F}_{i,b}\,, \tag{6.16}$$

where the lift and drag directions of each strip, $\boldsymbol{\ell}_{i,b}$ and $\mathbf{d}_{i,b}$, are defined from $\mathbf{V}^{\perp}_{i,b}$ and $\mathbf{n}_{i,b}$.

6.2.3 System Identification

To match the model with the manufactured glider described in Chapter 5, a calibration step is required. The calibration follows the standard aircraft system identification sequence [83], [84], using hand-launched flight records in still air. Open-loop throws first refine the rigid-body glider model, and flights with control surface deflections then adjust the model response to commanded inputs.

Because flight test time is limited and hand launches inevitably introduce variation in the initial conditions, only a compact set of aerodynamic correction parameters is identified:

$$\boldsymbol{\theta}_{\text{aero}} \;=\; \left[\boldsymbol{\theta}_m^{\top},\quad \boldsymbol{\theta}_{\text{lat}}^{\top},\quad \boldsymbol{\theta}_{\delta}^{\top}\right]^{\top}. \tag{6.17}$$

Here, $\boldsymbol{\theta}_m = [C^{\text{att}}_m,\, C^{\text{tr}}_m,\, C^{\text{post}}_m,\, C^{\text{post}}_{mq}]^{\top}$ corrects pitch moment in the attached, transition, and post-stall regimes introduced above, with an additional post-stall pitch damping term. $\boldsymbol{\theta}_{\text{lat}} = [\boldsymbol{\theta}_Y^{\top},\, \boldsymbol{\theta}_l^{\top},\, \boldsymbol{\theta}_n^{\top}]^{\top}$ contains side force, roll moment, and yaw moment coupling corrections, where $\boldsymbol{\theta} = [C_0,\, C_\beta,\, C_p,\, C_r]^{\top}$. $\boldsymbol{\theta}_\delta = [\boldsymbol{\gamma}_{\text{att}}^{\top},\, \boldsymbol{\gamma}_{\text{tr}}^{\top},\, \boldsymbol{\gamma}_{\text{post}}^{\top}]^{\top}$ contains conservative control surface effectiveness scales, where $\boldsymbol{\gamma} = [\gamma_a,\, \gamma_e,\, \gamma_r]^{\top}$.

The parameter vector in (6.17) is fitted by replaying the model from the measured launch condition under the logged control inputs, then minimising the error between the replayed and recorded flights:

$$\mathcal{J}(\boldsymbol{\theta}_{\text{aero}}) \;=\; \sum_{s\in\mathcal{D}_{\text{train}}} \sum_{k=1}^{N_{\text{fit},s}} \left\| \mathbf{y}^{\text{flight}}_{s,k} - \mathbf{y}^{\text{sim}}_{s,k}(\boldsymbol{\theta}_{\text{aero}}) \right\|^2_{Q_y} \,+\, \|\boldsymbol{\theta}_{\text{aero}} - \boldsymbol{\theta}_{\text{aero},0}\|^2_{R_\theta}\,. \tag{6.18}$$

Validation is performed on held-out throw records separate from those used for fitting. A calibration is accepted only if it improves or preserves the objective in (6.18) on these held-out records. This validation stage reduces the sensitivity of the calibrated model to unrepresentative launch records.

The baseline glider parameters and calibration results are summarised in Appendix F.1. The dataset contains 157 open-loop launches and 214 launches with control inputs. Longitudinal corrections are applied with higher confidence because sink and pitch are clearly observed, while transition regime, post-stall, and lateral corrections are treated more cautiously since launch variation in attitude, roll, yaw, and sideslip makes their effects harder to isolate. This modelling limitation reappears in Chapter 7, when replay results are compared with real flights with the controller developed in the following sections.

6.3 Randomised Simulation Environment

A lift-aware controller must be tested across launch variation, uncertain updraft conditions, aerodynamic mismatch, and system latency before flight testing. As discussed in Section 2.1, simulation provides a controlled sim-to-real development stage, where controllers can be designed and stress-tested without consuming limited flight test opportunities or exposing the physical glider to avoidable failures.

In this thesis, each finite-horizon simulated flight is defined by the environment instance

$$\mathbf{e} \;=\; (w_{\mathbf{e}}, \quad \boldsymbol{\theta}_p, \quad \boldsymbol{\theta}_\tau, \quad \boldsymbol{\theta}_\delta, \quad \mathbf{x}_0)\,, \tag{6.19}$$

where $w_{\mathbf{e}}$ is the vertical-flow field, with $w_{\mathbf{e}}(\mathbf{p}) = 0$ denoting still air, and fan-generated updrafts are presented using the residual annular GP surrogates developed in Chapter 4. The remaining terms define the glider properties $\boldsymbol{\theta}_p$, latency parameters $\boldsymbol{\theta}_\tau$, control surface effectiveness $\boldsymbol{\theta}_\delta$, and initial state $\mathbf{x}_0$.

As discussed in Section 4.4, domain randomisation is introduced to avoid designing the controller around a single imperfect updraft surrogate. By varying the surrogate position and strength, the real updraft is more likely to fall within the family of simulated cases [10], [12], [15]. The same principle is applied to other parts of the present problem, where glider mass, inertia, aerodynamic coefficients, control surface effectiveness, system latency, and launch condition also varied during controller development:

$$\mathbf{e} \sim p_{\mathcal{E}}(\mathbf{e}) \;=\; p(w_{\mathbf{e}})\, p(\boldsymbol{\theta}_p)\, p(\boldsymbol{\theta}_\tau)\, p(\boldsymbol{\theta}_\delta)\, p(\mathbf{x}_0)\,. \tag{6.20}$$

The detailed randomisation ranges, including the initial state sampling rule, are listed in Appendix F.2. They are chosen from the measured platform variation reported in the preceding chapters and refined using the launch and replay observations from the system identification procedure discussed above.

6.4 Viability-Guided Manoeuvre Selection

With the randomised simulation environment in place, the control problem for a small glider crossing a tight, flow-disturbed volume is reduced to a question of local commitment. Following the primitive, funnel, and governor ideas reviewed in Section 2.4, the controller is formulated as a lightweight real-time viability governor: at each decision step in flight, it selects a short, locally stabilised manoeuvre primitive from a finite pre-built library according to the current flight condition. The flight path then emerges as a sequence of admissible manoeuvres that collectively guide the glider toward mission completion.

6.4.1 Manoeuvre Primitive Library

In the present problem, manoeuvre primitives use a short execution horizon of $T = 0.10\,\mathrm{s}$, so that the controller can replan frequently within the confined arena. Each primitive is designed for a group of entry conditions, identified from the current 15-state vector in (6.1) through an entry class:

$$c \;=\; \chi(\mathbf{x}) \;\in\; \{\text{launch, post-launch, stable, boundary, recovery, terminal, failure}\}\,. \tag{6.21}$$

The primitive library is organised by these entry classes and the following manoeuvre families:

$$\mathcal{P} \;=\; \left\{ \begin{array}{l} \text{glide, safety margin recovery, updraft entry, updraft exploitation,} \\ \text{mild left turn, mild right turn, energy-retaining bank, safe exit} \end{array} \right\}. \tag{6.22}$$

Each family contains multiple candidate primitives, allowing the same manoeuvre type to be adapted to different entry conditions. For candidate manoeuvre primitive j, the nonlinear control-oriented glider model in Section 6.2 is linearised around a local operating point:

$$\mathbf{e}_{n+1} = \mathbf{A}_j \mathbf{e}_n + \mathbf{B}_j \Delta\boldsymbol{\nu}_n\,, \qquad \mathbf{e}_n = \mathbf{x}_n - \mathbf{x}_j^{\mathrm{ref}}\,, \qquad \Delta\boldsymbol{\nu}_n = \boldsymbol{\nu}_n - \boldsymbol{\nu}_j^{\mathrm{ref}} = -\mathbf{K}_j \hat{\mathbf{e}}_n\,, \tag{6.23}$$

where the reference state and aggregate command, $(\mathbf{x}_j^{\mathrm{ref}}, \boldsymbol{\nu}_j^{\mathrm{ref}})$, define that operating point, and $\hat{\mathbf{e}}_n$ is the state error used by the feedback controller after applying the latency model developed in Section 3.3.2. The local feedback gain $\mathbf{K}_j$ is obtained from the finite-horizon LQR problem

$$\min_{\Delta\boldsymbol{\nu}_0,\ldots,\Delta\boldsymbol{\nu}_{N-1}} \sum_{n=0}^{N-1} \left(\mathbf{e}_n^\top \mathbf{Q}_j \mathbf{e}_n \;+\; \Delta\boldsymbol{\nu}_n^\top \mathbf{R}_j \Delta\boldsymbol{\nu}_n \right) \;+\; \mathbf{e}_N^\top \mathbf{Q}_{f,j} \mathbf{e}_N\,. \tag{6.24}$$

To construct a rich library that enables the controller to select manoeuvres across the range of scenarios the glider may encounter, each candidate is generated and tested in simulation by combining a manoeuvre family from (6.22), an entry class from (6.21), and an environment instance from the randomised simulation distribution in (6.20). A generated candidate first becomes eligible if its local LQR can be built and a 0.10 s simulated execution from the sampled entry state gives enough information to classify the exit state, check the safety margin, and measure the energy change. The construction process then keeps a smaller retained set by removing candidates that are numerically invalid, end in an undefined flight condition, or duplicate an already covered manoeuvre. Each retained candidate is then expanded across the admissible subset of local speed-bin from 2.5 to $8.0\,\mathrm{m\,s^{-1}}$ in $0.5\,\mathrm{m\,s^{-1}}$ increments.

Within the selected entry class, the initial state is sampled over a wide range of positions, airspeeds, attitudes, body rates, and control surface deflections, detailed in Table F.18. The corresponding environment families are summarised in Table 6.1. Following the domain randomisation logic introduced in Section 6.3, this construction exposes the library to updraft, plant, timing, implementation, and launch uncertainty before it is brought to the physical system.

Table 6.1 Environment families used in the controller development and validation ladder.

Environment	Updraft representation	Introduced randomisation
Still air	No updraft, $w_{\mathbf{e}}(\mathbf{p}) = 0$	Isolates glider, launch, timing, and controller behaviour without fan forcing.
Fixed single-fan	Nominal single-fan residual annular GP surrogate	Tests response to one measured updraft region with fixed fan placement and strength.
Fixed four-fan	Nominal four-fan residual annular GP surrogate	Tests the controller in the multi-source lift field without randomised fan parameters.
Fan parameter variation	Randomised four-fan residual annular GP surrogate	Varies fan strength, width, and local fluctuation level while keeping the nominal fan positions.
Local fan-position variation	Randomised four-fan residual annular GP surrogate with shifted layout	Applies local placement shifts to test sensitivity to updraft displacement in the arena.
Active-fan-count variation	Randomised residual annular GP surrogate with different active fans	Varies the number of useful lift sources from no active fan to all four fans.
Full-environment variation	Randomised residual annular GP surrogate with arena-wide fan placement	Combines fan parameter, active-fan-count, and fan placement variation with fixed plant and timing.
Full-domain variation	Randomised residual annular GP surrogate combining the above variations	Combines full-environment variation with plant, launch, and latency uncertainty.

The expanded speed-bin variants are passed to the held-out validation introduced later in Section 6.5. The outcomes across the sampled validation cases are stored in $E_j(c)$, which contains the probability that execution from entry class c exits into the next manoeuvre entry class c^+, the hard failure[11] rate, the useful updraft exposure time, the mean specific energy change, and the minimum safety margin:

$$\begin{aligned} E_j(c) = \Big\{ \hat{P}_j(c^+|c) \;=\; \frac{N_j(c \to c^+)}{N_j(c)},\; \hat{p}_j^{\text{hard}}(c) \;=\; \frac{N_j^{\text{hard}}(c)}{N_j(c)},\; \hat{T}_{\ell,j}(c) \;=\; \frac{1}{N_j(c)} \sum_{s=1}^{N_j(c)} T_{\ell,s}\,, \\ \hat{G}_j(c) \;=\; \frac{1}{N_j(c)} \sum_{s=1}^{N_j(c)} \left[H_{s,\text{exit}} - H_{s,\text{entry}} \right],\; d_j^{\text{safe}}(\mathbf{x}_k) \;=\; \min_{\tau \in [0,T_j]} d_{\text{safe}}\left(\hat{\mathbf{x}}_j(\tau; \mathbf{x}_k) \right) \Big\}. \end{aligned} \tag{6.25}$$

Together with the manoeuvre family, entry class, local reference, command, feedback gain, and execution horizon, this validation record defines one candidate primitive in the library:

$$\Pi_j \;=\; \Big(\rho_j \in \mathcal{P},\; c_j^{\text{in}},\; \mathbf{x}_j^{\text{ref}},\; \boldsymbol{\nu}_j^{\text{ref}},\; \mathbf{K}_j,\; T_j,\; E_j \Big). \tag{6.26}$$

6.4.2 Spatial Lift Memory

The validation record in (6.25) is fixed before flight and only indicates which primitives performed well in simulation. During operation, repeated visits to nearby regions of an uncertain flow field provide new local evidence about useful or risky lift. Therefore, the controller further adds a bounded spatial memory, allowing later decisions to exploit observed benefit while still exploring admissible alternatives. After a selected primitive is executed, the controller records the residual lift evidence along its path:

$$\boldsymbol{\epsilon}_\ell \;=\; \left[\Delta H_{\text{obs}} - \Delta H_{\text{pred}},\; T_{\ell,\text{obs}} - T_{\ell,\text{pred}} \right]. \tag{6.27}$$

The residuals are stored in the $0.1 \times 0.1 \times 0.1$ m grid cells, and the local belief is computed from

$$\bar{\boldsymbol{\epsilon}}(\mathbf{p}) \;=\; \frac{\sum_{q \in \mathcal{N}_{0.20}(\mathbf{p})} \alpha_q d_q^{-1} \boldsymbol{\epsilon}_q}{\sum_{q \in \mathcal{N}_{0.20}(\mathbf{p})} \alpha_q d_q^{-1}}, \qquad \alpha_q \;=\; 2^{-\Delta n_q / n_{1/2}}, \tag{6.28}$$

where $\mathcal{N}_{0.20}(\mathbf{p})$ is the set of cells within a 0.20 m radius of position $\mathbf{p}$, d_q is the distance from $\mathbf{p}$ to cell q, and the factor α_q downweights older observations so the memory can adapt to changing flow conditions.

6.4.3 Viability Governor

The validation record and spatial lift memory make each stabilised primitive selectable during flight. Let $\mathcal{L}(c)$ be the primitives whose entry class matches the current state $c = \chi(\mathbf{x})$. The viability governor first removes candidates that do not support safe execution, leaving the admissible set

$$\mathcal{A}(\mathbf{x}) \;=\; \Big\{ j \in \mathcal{L}(c) : c_j^{\text{in}} = c,\; \hat{p}_j^{\text{hard}}(c) \le \bar{p}_{\text{hard}},\; \hat{P}_j^{\text{cont}}(c) + \hat{P}_j^{\text{term}}(c) > 0,\; d_j^{\text{safe}}(\mathbf{x}) \ge 0,\; \ell_j \in \mathcal{L}_\tau \Big\}, \tag{6.29}$$

where $\ell_j \in \mathcal{L}_\tau$ requires the candidate to be supported by the active latency model. The remaining admissible candidates are then scored as

$$\begin{aligned} S_j(\mathbf{x}) \;=\; & w_c \hat{P}_j^{\text{cont}} \;+\; w_t \hat{P}_j^{\text{term}} \;-\; w_h \hat{p}_j^{\text{hard}} \;+\; w_\ell \hat{T}_{\ell,j} \\ & +\; w_g \hat{G}_j \;+\; w_m M_j(\mathbf{x}) \;+\; w_b B_j(\mathbf{x}) \;-\; w_r R_j^{\text{regime}}(\mathbf{x}). \end{aligned} \tag{6.30}$$

Here, $M_j(\mathbf{x})$ is the predicted mission progress toward the exit face in Fig. 6.1, $B_j(\mathbf{x})$ is the bounded correction obtained by querying the spatial lift memory in (6.28) along the candidate path, and $R_j^{\text{regime}}(\mathbf{x})$ penalises candidates that rely on poorly calibrated flow regimes, as discussed in Section 6.2.3.

To prevent the spatial lift memory from degrading the selection, the governor accepts a memory-driven choice only when it does not weaken performance relative to the no-memory choice, this gives

$$\mathcal{A}_{\text{shield}}(\mathbf{x}) \;=\; \{ j \in \mathcal{A}(\mathbf{x}) : \Delta S_{\text{mem}}(j) > 0,\; \Delta \hat{p}_j^{\text{hard}} \le 0,\; \Delta \hat{P}_j^{\text{cont}} \ge -\epsilon_P,\; \Delta m_{\text{path},j} \ge -\epsilon_m \}. \tag{6.31}$$

[11] Hard failure in the present problem is defined as a floor or ceiling violation, numerical failure of the simulated trajectory, or a physically impossible flight state.

Finally, the governor selects the remaining manoeuvre primitive with the highest score in (6.30):

$$j^\star(\mathbf{x}) = \arg \max_{j \in \mathcal{A}_{\text{shield}}(\mathbf{x})} S_j(\mathbf{x}), \tag{6.32}$$

6.4.4 Library Compression and Decision Loop

The dense primitive library constructed in Section 6.4.1 gives the governor many possible local commitments. As discussed in Section 2.4, a larger library can improve behavioural coverage, but it also increases validation burden and online search cost. This trade-off is especially important here because each decision must be made within the 0.10 s primitive horizon. Hence, after validation, candidates are grouped by primitive family and entry class, and each group is represented by a medoid:

$$j_g^{\text{medoid}} = \arg\min_{j \in g} \sum_{i \in g} d(\mathbf{z}_i, \mathbf{z}_j), \tag{6.33}$$

where $\mathbf{z}_j$ collects behaviour, validation, airspeed, and the range of randomised environments where the primitive was tested. The medoid is always chosen as an existing validated primitive, ensuring that compression does not create synthetic controller objects that were never tested.

Combining all the components introduced above, the in-flight decision loop can be summarised as

$$\mathbf{x}_k \xrightarrow{\text{classify}} c_k \xrightarrow{\text{retrieve}} \mathcal{L}(c_k) \xrightarrow{\text{govern}} \mathcal{A}(\mathbf{x}_k) \xrightarrow{\text{memory check}} \mathcal{A}_{\text{shield}}(\mathbf{x}_k) \xrightarrow{\text{score}} j_k \xrightarrow{\text{execute}} \mathbf{x}_{k+1}. \tag{6.34}$$

6.5 Validation Ladder for Controller Development

This section assembles the plant model, randomised environments, and viability guided controller into a repeatable validation ladder for controller development.

After library construction in Section 6.4.1, the primitives are replayed using the same environment families introduced in Table 6.1, without retuning their local feedback gains. The validation record in (6.25) is then used to retain only primitives that, from sampled initial states in their assigned entry class, keep hard failure risk below the accepted threshold and exit into either a terminal class at the final exit face or a continuation class from which another validated primitive can be selected.

The validated set is then compressed by the medoid procedure in (6.33) into five executable library tiers of decreasing size: no-cluster, super-light, light, balanced, and heavy. Launch entry groups in (6.21) are given a larger representative budget to account for unavoidable human release variation.

All five library tiers are then used to tune the viability governor: its thresholds in (6.29), scoring weights in (6.30), memory shield margins in (6.31), and lift belief settings in (6.28). Tuning is performed by simulating the complete mission in Section 6.1.1 under the full-domain randomised setting in Table 6.1, aiming to maximise the controller design objective in (6.4) while reducing sensitivity to library size.

The final simulation stage evaluates the completed controller over the full mission using the same held-out strategy as the primitive validation above. A zero command open-loop case is included as a comparison baseline, and the spatial memory policy is tested with repeated launch histories: before each final held-out simulation across the same launch gate, the controller may observe $h = 3$, 10, 30 previous launches from the same held-out case, while the no-memory baseline uses $h = 0$. This stage provides the evidence needed to decide whether controller is safe and robust enough to transfer to reality.

This structure gives two practical advantages. First, it supports rapid diagnosis and iteration by separating primitive generation, library compression, and viability-governor tuning. Hence, development can resume from the failed stage without rerunning the full ladder. Second, because the basic decision algorithm is unchanged throughout the workflow, the ladder is not limited to the present problem: a different plant model, environment surrogate, or mission objective can be adapted by regenerating the primitive library and rerunning validation, without redesigning the controller architecture.

7 Crossing the Reality Gap from Simulation to Real Flight

The controller development workflow is designed to be efficient and extensible, but these advantages are only meaningful if the resulting controller can survive sim-to-real transfer. Therefore, this chapter asks the final question of this thesis: whether a controller built in simulation can be carried into real flight and still guide the glider safely through uncertain local updrafts, maximise useful lift to improve its specific energy, and balance exploitation with exploration over repeated launches, even when the vehicle, launch, command path, sensing, and flow field differ from the simulation.

7.1 Simulation Evidence for Controller Readiness

The first answer is sought in simulation, where the controller can be stress-tested before real flight.

7.1.1 Primitive Library Construction

The simulation results begin with the dense manoeuvre library generated by the construction procedure in Section 6.4.1. Each manoeuvre family starts from the same construction budget: 128 candidate primitives assigned to entry classes and tested over 12800 simulated replays. The construction removes many candidates while preserving manoeuvre diversity across all eight families.

Table 7.1 Dense manoeuvre primitive library.

Manoeuvre family	Eligible objects	Retained objects	Speed-bin variants
Glide	98	20	200
Safety margin recovery	98	20	199
Updraft entry	99	21	206
Updraft exploitation	97	19	193
Mild left turn	99	21	207
Mild right turn	97	19	193
Energy-retaining bank	98	20	200
Safe exit	99	21	207
Total	785	161	1605

7.1.2 Held-Out Primitive Validation

Held-out primitive validation applies the procedure in Section 6.5 to the speed-bin variants in Table 7.1. All 1605 speed-bin variants remain eligible after validation, but each carries a conservative validation record in (6.25). These records are summarised over 256800 held-out simulations in Table 7.2.

Table 7.2 Held-out primitive validation summary.

Validation result	Count	Rate	Interpretation
Usable transition outcome	64460	25.1%	Manoeuvre exits into a class that can continue or complete the route.
Controlled terminal boundary exit	46355	18.1%	Manoeuvre hit the wall or exit face with bounded attitude and body rates, without floor or ceiling violation.
Weak outcome	145964	56.8%	Manoeuvre completes the replay, but its energy gain, probability of continuation to the next manoeuvre, or terminal evidence is insufficient for acceptance.
Hard wall failure	21	0.008%	Manoeuvre hit the wall with excessive attitude or body rate.
Floor or ceiling failure	0	0%	Manoeuvre violate the floor or ceiling boundary.

7.1.3 Library Compression

As shown in Table 7.3, the validated variants are compressed into five library tiers using the medoid procedure in (6.33). This creates the independent cases used later in mission simulation to study the effect of library size.

Table 7.3 Compressed primitive library tiers after medoid selection.

Library	Total	Boundary	In flight	Launch	Recovery	Kept
No-cluster	1605	527	528	432	118	100.0%
Super-light	374	96	96	96	86	23.3%
Light	208	48	48	64	48	13.0%
Balanced	112	24	24	40	24	7.0%
Heavy	40	8	8	16	8	2.5%

7.1.4 Viability Governor Tuning

The viability governor is tuned using all five library tiers, each is tested over 200 full domain randomised mission simulations. To directly compare complete missions, a post-run score is defined:

$$J_{\text{sim}} \;=\; J_{\text{fail}} \;+\; \mathbb{I}_{J_{\text{fail}}=0}\left(J_{\text{route}} + \text{clip}(20\,\Delta H_\ell^+,\,0,\,40) \;+\; \text{clip}(5\,T_\ell,\,0,\,20) + J_{H,f}\right), \tag{7.1}$$

where

$$J_{H,f} \;=\; \mathbb{I}_{\text{mission}}\,\text{clip}\left(10\,H_{\text{res}}^+,\,0,\,20\right), \qquad H_{\text{res}} \;=\; H(\mathbf{x}_f) - H_{\text{floor}}\,. \tag{7.2}$$

Here, H_{res} is the glider's terminal specific energy, contributes to J_{sim} only when the mission is marked successful, otherwise sets to zero. The reward term J_{route} is 100 for mission completion, 30 for useful updraft capture without completing the mission, and 10 for a safe simulated flight that avoids failure but neither completes the route nor captures useful lift. The failure term J_{fail} is -100 for hard failure in (6.26), -70 for no viable primitive at launch, -40 in flight, -50 for lateral wall exit.

Table 7.4 Viability governor tuning result across five library tiers.

Library	Safe	Exit face	Hard failure	No viable	Lateral wall	Mean H_{res}	Mean J_{sim}
No-cluster	58.0%	54.0%	16.0%	6.0%	8.0%	2.22 m	47.2
Super-light	58.0%	54.0%	14.0%	8.0%	8.0%	2.22 m	48.4
Light	58.0%	54.0%	16.0%	6.0%	8.0%	2.21 m	47.2
Balanced	58.0%	54.0%	14.0%	8.0%	8.0%	2.22 m	48.4
Heavy	54.0%	50.0%	24.0%	8.0%	10.0%	2.15 m	33.6

As intended in Section 6.5, the tuned viability governor remains effective across a wide range of library sizes. Balanced, light, super-light, and no-cluster libraries give similar mission success rates, showing that increasing the library beyond a moderate size gives only marginal benefit. However, the heavy tier loses performance because it is too small. The same trend is further confirmed in next section.

This marks the end of controller development, since the remaining simulation stages are used to evaluate the frozen design. Across repeated development runs, the workflow consistently finished in under three hours on an older laptop, demonstrating strong computational efficiency and giving the controller design an additional practical advantage beyond what was discussed in Section 6.5.

7.1.5 Mission Simulation

The full mission simulation evaluates the completed controller over the environment families in Table 6.1. The dataset contains 36000 final mission runs and 384795 executed 0.100 s primitive flight segments.

The first marginal effect observed is launch speed, because the initial kinetic energy determines how much authority the first few primitives have before the glider approaches the arena boundary.

Table 7.5 Mission simulation results at different launch speeds magnitudes $\|\mathbf{v}(t_0)\|$ in $\mathrm{m\,s^{-1}}$.

$\|\mathbf{v}(t_0)\|$	Runs	Segments	Safe	Exit face	Main limit	Mean H_{res}	Mean J_{sim}
4.0–5.0	5600	63777	15.6%	14.5%	No viable primitive: 19.6%	1.29 m	10.5
5.0–6.0	7200	86690	64.6%	51.2%	Hard failure: 27.1%; wrong wall: 17.0%	1.51 m	30.8
6.0–8.0	16000	151788	90.9%	83.7%	Hard failure: 5.9%; wrong wall: 8.2%	2.56 m	99.2

Since the low-speed failures are dominated by insufficient energy, the remaining analysis focuses on runs with $\|\mathbf{v}(t_0)\| \geq 5.0\,\mathrm{m\,s^{-1}}$ to isolate controller behaviour within the feasible launch energy envelope. Across the environment cases defined in Table 6.1, Table 7.6 shows how controller performance degrades as the updraft and uncertainty model become less ideal. The still air case is kept as an isolated evidence case. It is expected to have a higher failure rate because there is no energy source for glider to exploit.

Table 7.6 Mission simulation across the held-out environment ladder for $\|\mathbf{v}(t_0)\| \geq 5.0\,\mathrm{m\,s^{-1}}$.

Environment	Runs	Segments	Safe	Exit face	Hard failure	No viable	Lateral wall	Mean H_{res}	Mean J_{sim}
Still air	2900	27534	67.0%	59.7%	22.6%	10.4%	7.3%	2.21 m	40.4
Fixed single-fan	2900	29832	86.1%	77.3%	10.6%	3.3%	8.8%	2.29 m	83.5
Fixed four-fan	2900	31146	97.2%	87.3%	2.1%	0.7%	10.6%	2.48 m	115.4
Fan parameter variation	2900	31092	92.7%	80.7%	5.5%	1.8%	14.8%	2.39 m	99.4
Local fan-position variation	2900	30941	87.6%	79.0%	9.3%	3.1%	14.2%	2.41 m	94.0
Active-fan-count variation	2900	29161	81.0%	71.9%	14.2%	4.8%	10.1%	2.29 m	72.0
Full-environment variation	2900	29275	81.2%	71.9%	14.1%	4.7%	10.2%	2.29 m	72.1
Full-domain variation	2900	29497	69.4%	60.8%	21.2%	9.4%	11.4%	2.24 m	46.9

The spatial memory policy is also evaluated. Its aggregate effect is insignificant, fewer than 10% of runs change the selected primitive because of memory, and the mission performance is almost unchanged.

Table 7.7 Mission simulation with different spatial memory history lengths for $\|\mathbf{v}(t_0)\| \geq 5.0\,\mathrm{m\,s^{-1}}$.

History h	Runs	Segments	Safe	Exit face	Hard failure	No viable	Lateral wall	Memory switch	Mean H_{res}	Mean J_{sim}
0	5800	59628	82.7%	73.5%	12.4%	4.9%	11.0%	0%	2.33 m	77.9
3	5800	59609	82.8%	73.6%	12.4%	4.8%	10.9%	7.5%	2.33 m	78.1
10	5800	59612	82.8%	73.6%	12.4%	4.8%	10.9%	7.9%	2.33 m	78.1
30	5800	59629	82.7%	73.5%	12.5%	4.8%	10.9%	8.4%	2.33 m	77.9

In the present arena, most controller authority is spent stabilising the glider and maintaining a viable route through the updraft field. The glider has limited freedom to choose a longer or more aggressive path purely to exploit remembered lift, especially when that manoeuvre would cost more energy than the expected lift benefit. Therefore, the spatial memory component should be treated as a questionable addition in this experiment: its aggregate benefit is small compared with the added controller development complexity. Inspection of individual launches gives the same interpretation.

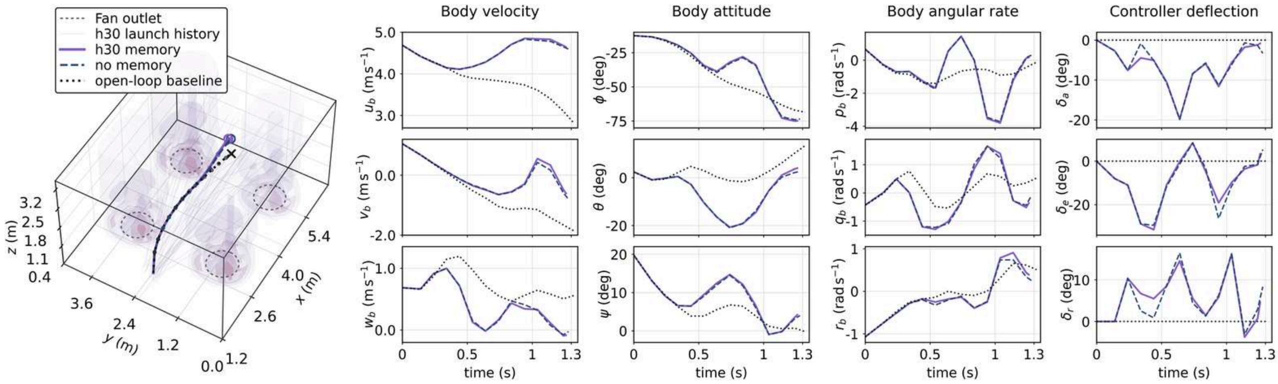


Fig. 7.1. Mission run 2322 in the local fan-position variation environment with the light cluster, showing that the memory component has little effect. Both the memory and no-memory flights survive and exploit the updraft, while the open-loop launch hits the ground.

The final marginal effect is library compression, because it changes the number of primitives the governor must inspect at each decision. The mission performance trend agrees with the tuning result in Section 7.1.4: Despite the heavily compressed one, no-cluster gives only a small benefit over the compressed libraries. The runtime columns make the practical trade-off clear. Balanced keeps near no-cluster mission performance, reduces the mean evaluated set to 7.0 candidates, and completes 85.2% of selector decisions within the 0.100 s primitive horizon. Therefore, it is used for the real flight controller.

Table 7.8 Effect of library compression on mission simulation and selector runtime. Mission outcomes use feasible launch runs with $\|\mathbf{v}(t_0)\| \geq 5.0\,\mathrm{m\,s^{-1}}$. Runtime is the online controller compute time.

Library	**Size**	**Runs**	**Safe**	**Exit face**	**Hard failure**	**Mean J_{sim}**	**Mean candidates**	**Mean time**	**$T <$ 0.100 s**
No-cluster	1605	4640	83.8%	75.1%	11.9%	81.0	110.1	258.1 ms	44.6%
Super-light	374	4640	82.7%	73.5%	12.9%	77.6	27.9	122.8 ms	70.8%
Light	208	4640	83.3%	74.3%	12.1%	79.5	14.4	79.4 ms	74.9%
Balanced	112	4640	84.0%	74.7%	10.8%	80.9	7.0	50.9 ms	85.2%
Heavy	40	4640	80.0%	70.3%	14.5%	71.1	2.9	39.4 ms	94.3%

The complete stage-by-stage controller development statistics are provided in Appendix G.1.

7.2 Real Flight Transfer

Since the flight test opportunities are limited, the sim-to-real transfer is first evaluated in cases already represented by the controller development ladder. This makes the expected failure modes easier to diagnose and allows the physical flights to be compared directly with the simulation evidence above.

The real-flight outcomes are reported with the same launch speed split used in simulation, $\|\mathbf{v}(t_0)\| = 6.0\,\mathrm{m\,s^{-1}}$. Below this threshold, failures are dominated by insufficient launch energy, as expected. Above it, still-air and single-fan flights transfer reliably, while the four-fan case remains limited by floor impacts. As in the simulation study, spatial memory does not form a separate success case in these validated transfer flights. Above $6.0\,\mathrm{m\,s^{-1}}$, still-air and single-fan flights already reach the exit face reliably without memory, leaving little room for memory to improve the outcome. In the four-fan case, memory reduces the exit face rate from 77.3% to 68.1% and increases floor impacts from 18.2% to 31.9%. Hence, in the validated flight cases, spatial memory is still not the main mechanism behind transfer.

Table 7.9 Real-flight transfer summary separated by launch speed magnitude $\|\mathbf{v}(t_0)\|$ in $\mathrm{m\,s^{-1}}$.

Flight condition	$\|\mathbf{v}(t_0)\|$	Closed-loop				Closed-loop with memory			
		throws	exit face	floor impact	mean H_{res}	throws	exit face	floor impact	mean H_{res}
Still-air	5.0–6.0	7	71.4%	28.6%	1.54 m	14	35.7%	64.3%	1.62 m
	6.0–8.0	23	100.0%	0.0%	1.81 m	16	100.0%	0.0%	1.90 m
Single-fan updraft	5.0–6.0	16	62.5%	37.5%	1.53 m	51	68.6%	27.5%	1.62 m
	6.0–8.0	14	100.0%	0.0%	1.76 m	9	100.0%	0.0%	1.67 m
Four-fan updraft	5.0–6.0	8	0.0%	87.5%	1.60 m	13	15.4%	84.6%	1.47 m
	6.0–8.0	22	77.3%	18.2%	1.84 m	47	68.1%	31.9%	1.64 m

To diagnose the source of the remaining reality gap, selected open-loop and closed-loop flights with similar launch states are replayed in simulation from the measured launch condition using the logged control input sequence. The representative four-fan case in Fig. 7.2 shows that the glider's longitudinal behaviour is reproduced well, but the lateral trajectory diverges in both open-loop and closed-loop replay. This divergence is consistent with two limitations discussed earlier: first, the weaker lateral system identification confidence, launch state sensitivity, and post-stall coupling identified in Section 6.2.3; and second, the larger four-fan surrogate error caused by plume interaction, unequal fan strength and alignment, height-dependent flow merging, and time variation not captured by the fixed measured surrogate, as discussed in Section 4.4.

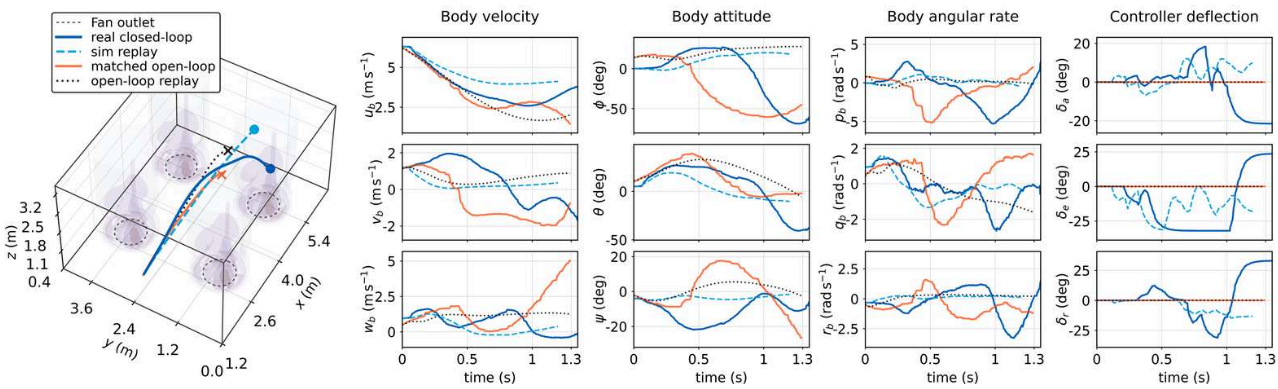


Fig. 7.2. Four-fan updraft replay. Closed-loop transfer flight with memory, throw 13, and the open-loop failure, throw 8, are selected with similar launch states.

7.3 Transfer Beyond the Validated Environments

The final experiments challenge the developed controller with two randomised fan layouts that were not included in the held-out simulation validation.

As demonstrated in Table 7.10, with representative cases in Fig. 7.3 and Fig. 7.4, these layouts give stronger sim-to-real evidence because open-loop flight fails frequently, mainly through floor impact after the updraft increases pitch and angle of attack beyond the glider's passive recovery margin, while the closed-loop controller reaches the exit face in most throws. However, memory again does not form a separate success mechanism: it reduces the exit face success in both random layouts, although the four-fan memory case gives the highest mean terminal energy among successful flights. This further motivates treating spatial memory as optional, or removing it from the present controller, because its small and inconsistent benefit does not justify the additional tuning and validation burden in this study.

Table 7.10 Real flight transfer beyond the validated environment layouts.

Layout	Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Random three-fan	Open-loop	10	30.0%	60.0%	10.0%	0.0%	1.63 m
	Closed-loop	30	86.7%	13.3%	0.0%	0.0%	1.90 m
	- with memory	30	70.0%	16.7%	6.7%	6.7%	1.85 m
Random four-fan	Open-loop	10	20.0%	70.0%	10.0%	0.0%	1.74 m
	Closed-loop	30	93.3%	0.0%	6.7%	0.0%	1.83 m
	- with memory	30	90.0%	10.0%	0.0%	0.0%	1.96 m

The replay comparisons in Fig. 7.3 and Fig. 7.4 reveal a pattern that is less visible in Section 7.2. In simulation, the updraft produces only a modest trajectory change when the glider passes through the modelled lift region along its centreline. In flight, the same passage is stretched and deflected more strongly, producing faster attitude and body rate changes than the model prediction. However, the real flights still succeed, and in some cases finish with more specific energy than the simulation replay.

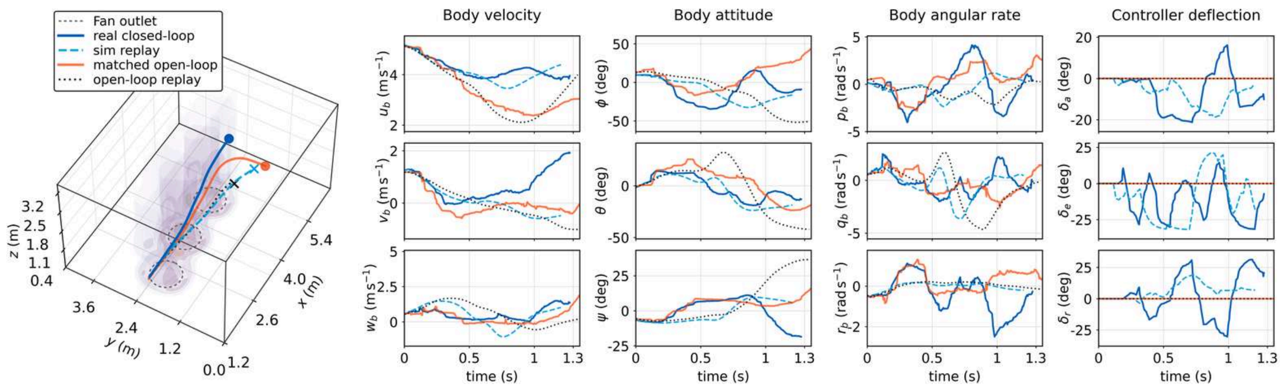


Fig. 7.3. Randomised three-fan challenge replay. Closed-loop transfer flight with memory, throw 18, and the open-loop failure, throw 8, are selected with similar launch states.

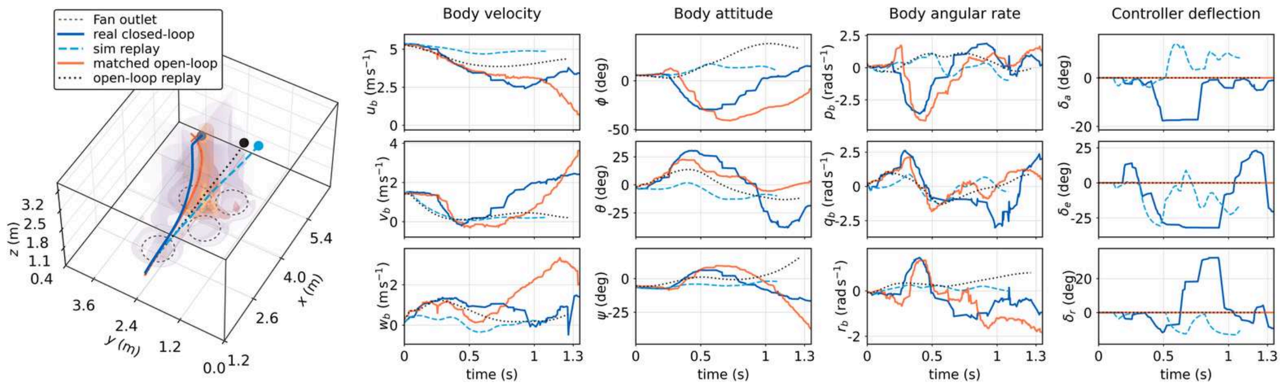


Fig. 7.4. Randomised four-fan challenge replay. Closed-loop transfer flight with memory, throw 26, and the open-loop failure, throw 7, are selected with similar launch states.

Therefore, at the final stage of the thesis, these random layout tests provide the strongest sim-to-real evidence. The controller survives layouts outside the validation ladder, remains viable when the real flow departs visibly from the replay. In that sense, the reality gap is crossed in flight.

The full flight test records are provided in Appendix G.2, with videos available upon request.

8 Conclusions and Future Work

This thesis began with a flight-control question: whether a small fixed-wing aircraft can safely exploit uncertain local lift inside a confined space, when the model used for controller development differs from reality. The answer developed here is conditional but positive. Sim-to-real transfer is possible in this setting, but only when the vehicle, flow field, command path, controller, validation ladder, and real-flight replay are treated as one traceable evidence chain.

8.1 Summary of Contributions and Main Findings

The contribution of this thesis is not limited to a single successful flight. It extends across three connected parts: an indoor workflow that makes the reality gap measurable, a viability-guided manoeuvre selection controller that exposes its assumptions, and real flight tests that show whether transfer succeeds.

8.1.1 Indoor Fixed-Wing Sim-to-Real Workflow

The first contribution is a reusable indoor workflow for fixed-wing sim-to-real testing under controlled aerodynamic forcing. The Flight Arena, Vicon-based state measurement and processing, command interface, measured latency model, safety-bounded flight volume, and manufactured glider were integrated into a repeatable experimental system. This workflow separates the main sources of difficulty: sensing, command delay, vehicle design limitations, updraft modelling, controller development, and real flight replay. It turns sim-to-real transfer from a binary success claim into a diagnosable engineering process.

A second part of this workflow is the updraft surrogate. The residual annular Gaussian process model gave the best reconstruction of the measured fan-generated updraft, reducing single-fan error significantly compared with simpler plume models. The four-fan case remained harder since plume interaction, fan-to-fan asymmetry, height-dependent merging, and time variation produced a less coherent residual field. This finding became important later: the controller could be transferred through four-fan updrafts, but the multi-source flow model is one of the main sources of the reality gap.

8.1.2 Viability-Guided Manoeuvre Selection

The second contribution is a viability-guided controller that replaces long-horizon trajectory tracking with short, locally stabilised manoeuvre primitives. Each primitive carries explicit evidence about its entry class, likely exit class, hard-failure rate, safety margin, lift exposure, and energy change. This makes online selection conservative by construction: a primitive is scored only after passing admissibility checks based on validation evidence, timing compatibility, and safety margin.

The simulation results show that this structure is computationally practical. The dense construction process generated 1605 speed-bin variants from 102400 construction replays, and held-out primitive validation summarised 256800 replay labels before library compression. Compression then exposed the central trade-off. The no-cluster library preserved the largest action set but was too slow for reliable real-time deployment. The balanced library retained near no-cluster mission performance while reducing the mean evaluated set from 110.1 to 7.0 candidates and completing 85.2% of selector decisions within the 0.10 s primitive horizon. In contrast, the heavy library was even faster but lost manoeuvre coverage, reducing mission performance. This shows that expanding the library beyond a moderate size gives only marginal benefit, so the library can be made much smaller, but also not arbitrarily small.

The spatial lift-memory component was less successful. In simulation, fewer than 10% of feasible launches changed the selected primitive due to memory, and mission performance was almost the same. In real flight, memory did not create a separate success mode and sometimes reduced the mission success. This is likely because, in the confined arena, most control authority is already spent maintaining a viable route through the updraft field, so the glider has limited freedom to take longer or more aggressive paths to exploit remembered lift. Therefore, bounded memory can only be concluded as a useful architectural idea, but in this experiment, its benefit does not justify the added tuning and validation burden.

8.1.3 Simulation and real flight Transfer Evidence

The third contribution is the transfer evidence itself. Mission simulation showed that launch speed is the main physical gate: below $5.0\,\mathrm{m\,s^{-1}}$, failures are dominated by insufficient initial kinetic energy, while at $6.0-8.0\,\mathrm{m\,s^{-1}}$ the controller reached the exit face in 83.7% of runs with a mean mission score of 99.2. Across the held-out environment ladder, performance degraded as uncertainty increased, but the controller still retained useful mission performance under full-domain randomisation.

The real flights confirmed the main simulation trend but also exposed the remaining model limitations under a more conservative launch speed range. Above $6.0\,\mathrm{m\,s^{-1}}$, still-air and single-fan flights reached the exit face reliably. The four-fan case remained harder, with floor impacts still present, consistent with the larger four-fan surrogate error and weaker lateral identification confidence. Matched simulation replay showed that longitudinal behaviour was reproduced reasonably, while lateral trajectories diverged, confirming that transfer did not occur because the simulator was perfect.

The strongest evidence came from the randomised fan layouts that were not included in the held-out validation ladder. In the random three-fan layout, open-loop mission success was 30.0%, while the closed-loop controller achieved 86.7%. In the random four-fan layout, open-loop success was 20.0%, while closed-loop success reached 93.3%. These tests show that the controller was not merely reproducing one validated updraft case. It survived new structured lift layouts under a reality gap that remained visible in replay, which is the central sim-to-real claim of the thesis.

8.2 Recommendations for Future Work

The first direction is to reduce the remaining reality gap. The present thesis used a quasi-steady glider model and a measured but fixed updraft surrogate for simplicity. The flight test discrepancies are consistent with known modelling difficulties in high angle of attack, sideslip-coupled and post-stall fixed-wing UAV dynamics [89], [185], [186], and with the need to model thermal updrafts as uncertain, drifting, time-varying flow fields [32], [116], [187]. Instead of immediately replacing the present quasi-steady model with a full CFD pipeline, future work should target control-oriented reduced-order modelling (ROM): unsteady aerodynamic ROMs, parametric ROMs, and data-driven operator-inference methods provide a practical middle layer between rigid-body simplicity and high-fidelity fluid structure [188], [189], [190].

The second direction is to move from sequential design toward co-design. In the present workflow, the glider is first designed, then identified, and finally controlled. This separation made the experiment tractable, but the results also show that vehicle design, control authority, latency, and mission success are tightly coupled. Future work could optimise geometry, structure, control surfaces, and guidance objectives together, following MDO [73], [74], [170] and control co-design methods [191], [192]. For aircraft, this direction is supported by high-fidelity aerostructural optimisation [193], [194], mission-aware wing optimisation [195] and simultaneous design-trajectory optimisation [196]. Morphing wing mechanisms provide another extension, because they allow the vehicle to change its aerodynamic response in flight and can themselves be integrated in concurrent aerostructural and mission optimisation [80], [197], [198].

The third direction is to extend the workflow toward greater autonomy. The present controller still assumes a known arena, a fixed updraft surrogate, motion-capture feedback, and offline validation. Future systems should reason from flight-level objectives instead of pre-specified manoeuvres [199], while estimating local flow online and updating uncertainty across repeated flights [200], [201], [202]. In this sense, the Flight Arena workflow is valuable beyond the specific controller tested here: it provides a reusable, reproducible, and extensible setting where flight experience can be accumulated and transferred while keeping the vehicle, model, controller, and validation evidence traceable.

References

[1] Colomina I, Molina P. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2014;92: 79–97. https://doi.org/10.1016/j.isprsjprs.2014.02.013.

[2] Nex F, Remondino F. UAV for 3D mapping applications: a review. *Applied Geomatics*. 2013;6(1): 1–15. https://doi.org/10.1007/s12518-013-0120-x.

[3] Manfreda S, McCabe M, Miller P, Lucas R, Pajuelo Madrigal V, Mallinis G, *et al.* On the Use of Unmanned Aerial Systems for Environmental Monitoring. *Remote Sensing*. 2018;10(4): 641. https://doi.org/10.3390/rs10040641.

[4] Erdelj M, Natalizio E, Chowdhury KR, Akyildiz IF. Help from the Sky: Leveraging UAVs for Disaster Management. *IEEE Pervasive Computing*. 2017;16(1): 24–32. https://doi.org/10.1109/mprv.2017.11.

[5] Neubauer K, Bullard E, Blunt R. *Collection of Data with Unmanned Aerial Systems (UAS) for Bridge Inspection and Construction Inspection*. Federal Highway Administration. 2021. https://doi.org/10.21949/1521674.

[6] Adkins KA. Urban flow and small unmanned aerial system operations in the built environment. *International Journal of Aviation, Aeronautics, and Aerospace*. 2019;6(1): 10. https://doi.org/10.15394/ijaaa.2019.1312.

[7] Nithya DS, Quaranta G, Muscarello V, Liang M. Review of Wind Flow Modelling in Urban Environments to Support the Development of Urban Air Mobility. *Drones*. 2024;8(4): 147–147. https://doi.org/10.3390/drones8040147.

[8] Gavrinev VO, Volkova MA, Romanov AM. Autonomous Thermal Soaring Methods: A Systematic Review. *Unmanned Systems*. 2025; 1–25. https://doi.org/10.1142/s2301385027300010.

[9] Allen MJ, Lin V. Guidance and Control of an Autonomous Soaring Vehicle with Flight Test Results. *45th AIAA Aerospace Sciences Meeting and Exhibit*. 2007. https://doi.org/10.2514/6.2007-867.

[10] Muratore F, Ramos F, Turk G, Yu W, Gienger M, Peters J. Robot Learning From Randomized Simulations: A Review. *Frontiers in Robotics and AI*. 2022;9. https://doi.org/10.3389/frobt.2022.799893.

[11] Jakobi N, Husbands P, Harvey I. Noise and the reality gap: The use of simulation in evolutionary robotics. *European Conference on Artificial Life*. 1995. p. 704–720. https://doi.org/10.1007/3-540-59496-5_337.

[12] Aljalbout E, Xing J, Romero A, Akinola I, Garrett CR, Heiden E, *et al.* The Reality Gap in Robotics: Challenges, Solutions, and Best Practices. *Annual Review of Control Robotics and Autonomous Systems*. 2025; https://doi.org/10.1146/annurev-control-031924-100130.

[13] Allevato AD, Schaertl Short E, Pryor M, Thomaz AL. Iterative residual tuning for system identification and sim-to-real robot learning. *Autonomous Robots*. 2020;44(7): 1167–1182. https://doi.org/10.1007/s10514-020-09925-w.

[14] Du Y, Watkins O, Darrell T, Abbeel P, Pathak D. Auto-Tuned Sim-to-Real Transfer. *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE; 2021. https://doi.org/10.1109/icra48506.2021.9562091.

[15] Tobin J, Fong R, Ray A, Schneider J, Zaremba W, Abbeel P. Domain randomization for transferring deep neural networks from simulation to the real world. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE; 2017. p. 23–30. https://doi.org/10.1109/IROS.2017.8202133.

[16] Schaff C, Sedal A, Ni S, Walter MR. Sim-to-real transfer of co-optimized soft robot crawlers. *Autonomous Robots*. 2023;47(8): 1195–1211. https://doi.org/10.1007/s10514-023-10130-8.

[17] Peng XB, Andrychowicz M, Zaremba W, Abbeel P. Sim-to-Real Transfer of Robotic Control with Dynamics Randomization. *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE; 2018. p. 3803–3810. https://doi.org/10.1109/ICRA.2018.8460528.

[18] Tiboni G, Arndt K, Ville Kyrki. DROPO: Sim-to-real transfer with offline domain randomization. *Robotics and Autonomous Systems*. 2023;166: 104432. https://doi.org/10.1016/j.robot.2023.104432.

[19] Belkhale S, Li R, Kahn G, McAllister R, Calandra R, Levine S. Model-Based Meta-Reinforcement Learning for Flight With Suspended Payloads. *IEEE Robotics and Automation Letters*. 2021;6(2): 1471–1478. https://doi.org/10.1109/lra.2021.3057046.

[20] Zhu S, Mou L, Li D, Ye B, Huang R, Zhao H. VR-Robo: A Real-to-Sim-to-Real Framework for Visual Robot Navigation and Locomotion. *IEEE Robotics and Automation Letters*. 2025;10(8): 7875–7882. https://doi.org/10.1109/lra.2025.3575648.

[21] Chebotar Y, Handa A, Makoviychuk V, Macklin M, Issac J, Ratliff N, *et al.* Closing the Sim-to-Real Loop: Adapting Simulation Randomization with Real World Experience. *2019 International Conference on Robotics and Automation (ICRA)*. IEEE; 2019. p. 8973–8979. https://doi.org/10.1109/ICRA.2019.8793789.

[22] Bousmalis K, Irpan A, Wohlhart P, Bai Y, Kelcey M, Kalakrishnan M, *et al.* Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping. *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE; 2018. p. 4243–4250. https://doi.org/10.1109/ICRA.2018.8460875.

[23] Kaufmann E, Bauersfeld L, Loquercio A, Müller M, Koltun V, Scaramuzza D. Champion-level drone racing using deep reinforcement learning. *Nature*. 2023;620(7976): 982–987. https://doi.org/10.1038/s41586-023-06419-4.

[24] Loquercio A, Kaufmann E, Ranftl R, Dosovitskiy A, Koltun V, Scaramuzza D. Deep Drone Racing: From Simulation to Reality With Domain Randomization. *IEEE Transactions on Robotics*. 2020;36(1): 1–14. https://doi.org/10.1109/tro.2019.2942989.

[25] Wada D, Araujo-Estrada S, Windsor S. Sim-to-Real Transfer for Fixed-Wing Uncrewed Aerial Vehicle: Pitch Control by High-Fidelity Modelling and Domain Randomization. *IEEE Robotics and Automation Letters*. 2022;7(4): 11735–11742. https://doi.org/10.1109/lra.2022.3205442.

[26] Bøhn E, Coates EM, Reinhardt D, Johansen TA. Data-Efficient Deep Reinforcement Learning for Attitude Control of Fixed-Wing UAVs: Field Experiments. *IEEE Transactions on Neural Networks and Learning Systems*. 2023;35(3): 3168–3180. https://doi.org/10.1109/tnnls.2023.3263430.

[27] Beard RW, Ferrin J, Humpherys J. Fixed Wing UAV Path Following in Wind With Input Constraints. *IEEE Transactions on Control Systems Technology*. 2014;22(6): 2103–2117. https://doi.org/10.1109/tcst.2014.2303787.

[28] Mohamed A, Massey K, Watkins S, Clothier R. The attitude control of fixed-wing MAVS in turbulent environments. *Progress in Aerospace Sciences*. 2014;66: 37–48. https://doi.org/10.1016/j.paerosci.2013.12.003.

[29] Britter R, Hanna SR. Flow and Dispersion in Urban Areas. *Annual Review of Fluid Mechanics*. 2003;35(1): 469–496. https://doi.org/10.1146/annurev.fluid.35.101101.161147.

[30] Tominaga Y, Mochida A, Yoshie R, Kataoka H, Nozu T, Yoshikawa M, *et al.* AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings. *Journal of Wind Engineering and Industrial Aerodynamics*. 2008;96(10-11): 1749–1761. https://doi.org/10.1016/j.jweia.2008.02.058.

[31] Blocken B. Computational Fluid Dynamics for urban physics: Importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations. *Building and Environment*. 2015;91: 219–245. https://doi.org/10.1016/j.buildenv.2015.02.015.

[32] Achermann F, Stastny T, Danciu B, Kolobov A, Chung JJ, Siegwart R, *et al.* WindSeer: real-time volumetric wind prediction over complex terrain aboard a small uncrewed aerial vehicle. *Nature Communications*. 2024;15(1): 3507. https://doi.org/10.1038/s41467-024-47778-4.

[33] McCormick BW. *Aerodynamics, Aeronautics, and Flight Mechanics*. Hoboken, NJ: John Wiley & Sons; 1995.

[34] Allen MJ. Autonomous Soaring for Improved Endurance of a Small Uninhabitated Air Vehicle. *43rd AIAA Aerospace Sciences Meeting and Exhibit*. 2005. https://doi.org/10.2514/6.2005-1025.

[35] Bencatel R, Tasso de Sousa J, Girard A. Atmospheric flow field models applicable for aircraft endurance extension. *Progress in Aerospace Sciences*. 2013;61: 1–25. https://doi.org/10.1016/j.paerosci.2013.03.001.

[36] Ákos Z, Nagy M, Leven S, Vicsek T. Thermal soaring flight of birds and unmanned aerial vehicles. *Bioinspiration & Biomimetics*. 2010;5(4): 045003. https://doi.org/10.1088/1748-3182/5/4/045003.

[37] Reddy G, Celani A, Sejnowski TJ, Massimo Vergassola. Learning to soar in turbulent environments. *Proceedings of the National Academy of Sciences*. 2016;113(33): E4877–E4884. https://doi.org/10.1073/pnas.1606075113.

[38] Suys T, Hwang S, De Croon GCHE, Remes BDW. Autonomous Control for Orographic Soaring of Fixed-Wing UAVs. *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE; 2023. p. 5338–5344. https://doi.org/10.1109/icra48891.2023.10161578.

[39] Hwang S, Remes BDW, De Croon GC. AOSoar: Autonomous Orographic Soaring of a Micro Air Vehicle. *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE; 2023. p. 5003–5010. https://doi.org/10.1109/iros55552.2023.10341699.

[40] Thomas C. *NASA Dryden aerospace engineer Michael Allen hand-launches a model motorized sailplane during a study validating the use of heat thermals to extend flight time*. NASA Image and Video Library. 2005. https://images.nasa.gov/details/EC05-0198-11 [Accessed 7th March 2026].

[41] Wharington J, Israel H. Control of a High Endurance Unmanned Air Vehicle. *21st Congress of International Council of the Aeronautical Sciences*. 1998. https://www.icas.org/icas_archive/ICAS1998/PAPERS/371.

[42] Lenschow DH, Stephens PL. The role of thermals in the convective boundary layer. *Boundary-Layer Meteorology*. 1980;19(4): 509–532. https://doi.org/10.1007/bf00122351.

[43] Turner JS. *Buoyancy Effects in Fluids*. Cambridge: Cambridge University Press; 2009.

[44] Morto BR, Taylor GI, John Stewart T. Turbulent gravitational convection from maintained and instantaneous sources. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*. 1956;234(1196): 1–23. https://doi.org/10.1098/rspa.1956.0011.

[45] Bower G, Flanzer T, Naiman A, Saripalli S. Dynamic Environment Mapping for Autonomous Thermal Soaring. *AIAA Guidance, Navigation, and Control Conference*. 2010. https://doi.org/10.2514/6.2010-8031.

[46] Raffel M, Willert CE, Scarano F, Kähler CJ, Wereley ST, Kompenhans J. *Particle Image Velocimetry: A Practical Guide*. Cham: Springer; 2018.

[47] Zagaglia D, Giuni M, Green RB. Investigation of the Rotor–Obstacle Aerodynamic Interaction in Hovering Flight. *Journal of the American Helicopter Society*. 2018;63(3): 1–12. https://doi.org/10.4050/jahs.63.032007.

[48] Ramasamy M, Leishman JG. Benchmarking Particle Image Velocimetry with Laser Doppler Velocimetry for Rotor Wake Measurements. *AIAA Journal*. 2007;45(11): 2622–2633. https://doi.org/10.2514/1.28130.

[49] Jørgensen FE. *How to Measure Turbulence with Hot-wire Anemometers*. Dantec Dynamics; 2005.

[50] Barlow JB, Rae WH, Pope A. *Low-speed wind tunnel testing*. 3rd ed. Hoboken, NJ: John Wiley & Sons; 1999.

[51] Bendat JS, Piersol AG. *Random Data: Analysis and Measurement Procedure*. 4th ed. Hoboken, NJ: John Wiley & Sons; 2011.

[52] Coleman HW, Steele WG. *Experimentation, Validation, and Uncertainty Analysis for Engineers*. Hoboken, NJ: John Wiley & Sons; 2009.

[53] Allen MJ. Updraft Model for Development of Autonomous Soaring Uninhabited Air Vehicles. *44th AIAA Aerospace Sciences Meeting and Exhibit*. 2006. https://doi.org/10.2514/6.2006-1510.

[54] Gedeon J. Dynamic analysis of dolphin-style thermal cross-country flight. *Technical Soaring*. 1974;3(1): 9–19. https://journals.sfu.ca/ts/index.php/ts/article/view/1114.

[55] Coleman RP, Feingold AM, Stempin CW. *Evaluation of the induced-velocity field of an idealized helicopter rotor*. National Advisory Committee for Aeronautics. 1945. https://ntrs.nasa.gov/citations/19930093086.

[56] Leishman JG. *Principles of Helicopter Aerodynamics*. Cambridge: Cambridge University Press; 2008.

[57] Tanner PE, Overmeyer AD, Jenkins LN, Yao CS, Bartram SM. Experimental Investigation of Rotorcraft Outwash in Ground Effect. *AHS International Annual Forum & Technology Display*. No. NF1676L-21054; 2015. https://ntrs.nasa.gov/citations/20160006428.

[58] Seeger M. Gaussian Processes for Machine Learning. *International Journal of Neural Systems*. 2004;14(02): 69–106. https://doi.org/10.1142/s0129065704001899.

[59] Buhmann MD. *Radial Basis Functions*. Cambridge: Cambridge University Press; 2003.

[60] Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and Analysis of Computer Experiments. *Statistical Science*. 1989;4(4): 409–423. https://doi.org/10.1214/ss/1177012413.

[61] Simons M. *Model Aircraft Aerodynamics*. Fox Chapel Publishing; 2015.

[62] Coussot JL. *On ne vas pas faire de chichis... Mais on peut faire un "CHICHI"!* JLC Aviation. https://www.jlc-aviation.fr/a%C3%A9romod%C3%A9lisme/essais/silence-model-chichi [Accessed 3rd March 2026].

[63] Silence Model. *CeePee 750 mm*. https://silencemodel.fr/kit-de-0-a-1990-mm/549-ceepee-750-mm.html [Accessed 3rd March 2026].

[64] MicroBirds. *Feather^2 Squared*. https://microbirds.com/rc-plane-radio-control-kit-glider-feather [Accessed 22nd January 2026].

[65] MicroBirds. *FireFly 2.0*. https://microbirds.com/radio-control-rc-gliders-firefly2 [Accessed 9th March 2026].

[66] Cory RE. *Supermaneuverable perching*. [Doctoral Thesis] [Massachusetts Institute of Technology]; 2010. http://hdl.handle.net/1721.1/60142.

[67] Depron. *Depron Insulation board: Technical Data Sheet*. 2013. https://www.depron-daemmplatte.de/site/assets/files/1044/pdb-depron-d_mmplatte_gb_20160301.pdfb_20160301.pdf [Accessed 23rd February 2026].

[68] Lissaman PBS. Low-Reynolds-Number Airfoils. Annual Review of Fluid Mechanics. 1983;15(1): 223–239. https://doi.org/10.1146/annurev.fl.15.010183.001255.

[69] Mateescu D, Abdo M. Analysis of flows past airfoils at very low Reynolds numbers. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*. 2010;224(7): 757–775. https://doi.org/10.1243/09544100jaero715.

[70] Mueller TJ, DeLaurier JD. Aerodynamics of small vehicles. *Annual Review of Fluid Mechanics*. 2003;35(1): 89–111. https://doi.org/10.1146/annurev.fluid.35.101101.161102.

[71] Anderson JD. *Aircraft Performance and Design*. Boston, MA: Mcgraw-Hill Higher Education; 2012.

[72] Raymer DP. *Aircraft Design: A Conceptual Approach*. Reston, VA: American Institute of Aeronautics And Astronautics; 2012.

[73] Martins JRRA, Lambe AB. Multidisciplinary Design Optimization: A Survey of Architectures. *AIAA Journal*. 2013;51(9): 2049–2075. https://doi.org/10.2514/1.j051895.

[74] Gray JS, Hwang JT, Martins JRRA, Moore KT, Naylor BA. OpenMDAO: an open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*. 2019;59(4): 1075–1104. https://doi.org/10.1007/s00158-019-02211-z.

[75] Lukaczyk TW, Wendorff AD, Colonno M, Economon TD, Alonso JJ, Orra TH, *et al.* SUAVE: An Open-Source Environment for Multi-Fidelity Conceptual Vehicle Design. *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. 2015. https://doi.org/10.2514/6.2015-3087.

[76] Burnell E, Damen NB, Hoburg W. GPkit: A Human-Centered Approach to Convex Optimization in Engineering Design. *CHI '20: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020. https://doi.org/10.1145/3313831.3376412.

[77] Sharpe P. *AeroSandbox: A Differentiable Framework for Aircraft Design Optimization*. [Master Thesis] [Massachusetts Institute of Technology]; 2021. https://hdl.handle.net/1721.1/140023.

[78] Drela M. XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils. In: Mueller TJ (ed.) *Low Reynolds Number Aerodynamics*. Berlin, Heidelberg: Springer; 1989. p. 1–12. https://link.springer.com/chapter/10.1007/978-3-642-84010-4_1.

[79] Khan W. *Dynamics Modeling of Agile Fixed-Wing Unmanned Aerial Vehicles*. [Doctoral Thesis] [McGill University]; 2016. https://escholarship.mcgill.ca/concern/theses/4m90dz005.

[80] Wanner J, Phan HV, Toumieh C, Floreano D. Flight through Narrow Gaps with Morphing-Wing Drones. ArXiv [Preprint]. Version 1. 2026. https://doi.org/10.48550/arXiv.2603.12059.

[81] Buda AA, Chen X, Botteghi N, Fasel U. Robust Co-design Optimisation for Agile Fixed-Wing UAVs. ArXiv [Preprint]. Version 1. 2025. https://doi.org/10.48550/arXiv.2603.11130.

[82] Moore J. *Robust Post-Stall Perching with a Fixed-Wing UAV*. [Doctoral Thesis] [Massachusetts Institute of Technology]; 2014. http://hdl.handle.net/1721.1/93861.

[83] Dorobantu A, Murch A, Mettler B, Balas G. System Identification for Small, Low-Cost, Fixed-Wing Unmanned Aircraft. *Journal of Aircraft*. 2013;50(4): 1117–1130. https://doi.org/10.2514/1.c032065.

[84] Simmons BM, Gresham JL, Woolsey CA. Flight-Test System Identification Techniques and Applications for Small, Low-Cost, Fixed-Wing Aircraft. *Journal of Aircraft*. 2023;60(5): 1503–1521. https://doi.org/10.2514/1.c037260.

[85] Grymin DJ, Farhood M. Two-Step System Identification and Trajectory Tracking Control of a Small Fixed-Wing UAV. *Journal of Intelligent & Robotic Systems*. 2015;83(1): 105–131. https://doi.org/10.1007/s10846-015-0298-8.

[86] Stevens BL, Lewis FL, Johnson EN. *Aircraft Control and Simulation: dynamics, Controls design, and Autonomous Systems*. Hoboken, NJ: John Wiley & Sons; 2016.

[87] Anderson BDO, Moore JB. *Optimal control: linear quadratic methods*. Mineola, NY: Dover Publications; 2007.

[88] López J, Dormido R, Dormido S, Gómez JP. A Robust H_∞ Controller for an UAV Flight Control System. *The Scientific World Journal*. 2015;2015(1). https://doi.org/10.1155/2015/403236.

[89] Mathisen S, Gryte K, Gros S, Johansen TA. Precision Deep-Stall Landing of Fixed-Wing UAVs Using Nonlinear Model Predictive Control. *Journal of Intelligent & Robotic Systems*. 2020;101(1). https://doi.org/10.1007/s10846-020-01264-3.

[90] Vachtsevanos G, Tang L, Drozeski G, Gutierrez L. From mission planning to flight control of unmanned aerial vehicles: Strategies and implementation tools. *Annual Reviews in Control*. 2005;29(1): 101–115. https://doi.org/10.1016/j.arcontrol.2004.11.002.

[91] Sujit PB, Saripalli S, Sousa JB. Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicles. *IEEE Control Systems Magazine*. 2014;34(1): 42–59. https://doi.org/10.1109/mcs.2013.2287568.

[92] Nelson DR, Barber DB, McLain TW, Beard RW. Vector Field Path Following for Miniature Air Vehicles. *IEEE Transactions on Robotics*. 2007;23(3): 519–529. https://doi.org/10.1109/tro.2007.898976.

[93] Frazzoli E, Dahleh MA, Feron E. Maneuver-based motion planning for nonlinear systems with symmetries. *IEEE Transactions on Robotics*. 2005;21(6): 1077–1091. https://doi.org/10.1109/tro.2005.852260.

[94] Burridge RR, Rizzi AA, Koditschek DE. Sequential Composition of Dynamically Dexterous Robot Behaviors. *The International Journal of Robotics Research*. 1999;18(6): 534–555. https://doi.org/10.1177/02783649922066385.

[95] Frazzoli E, Dahleh MA, Feron E. Real-Time Motion Planning for Agile Autonomous Vehicles. *Journal of Guidance, Control, and Dynamics*. 2002;25(1): 116–129. https://doi.org/10.2514/2.4856.

[96] Tedrake R, Manchester IR, Tobenkin M, Roberts JW. LQR-trees: Feedback Motion Planning via Sums-of-Squares Verification. *The International Journal of Robotics Research*. 2010;29(8): 1038–1052. https://doi.org/10.1177/0278364910369189.

[97] Tobenkin M, Manchester IR, Tedrake R. Invariant Funnels around Trajectories using Sum-of-Squares Programming. *IFAC Proceedings Volumes*. 2011;44(1): 9218–9223. https://doi.org/10.3182/20110828-6-it-1002.03098.

[98] Garone E, Di Cairano S, Kolmanovsky I. Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica*. 2017;75: 306–328. https://doi.org/10.1016/j.automatica.2016.08.013.

[99] Bemporad A. Reference governor for constrained nonlinear systems. *IEEE Transactions on Automatic Control*. 1998;43(3): 415–419. https://doi.org/10.1109/9.661611.

[100] Aubin JP, Bayen AM, Saint-Pierre P. *Viability theory: new directions*. Springer; 2013.

[101] Blanchini F. *Set invariance in control*. *Automatica*. 1999;35(11): 1747–1767. https://doi.org/10.1016/s0005-1098(99)00113-2.

[102] Botros A, Smith SL. Spatio-Temporal Lattice Planning Using Optimal Motion Primitives. *IEEE Transactions on Intelligent Transportation Systems*. 2023;24(11): 11950–11962. https://doi.org/10.1109/tits.2023.3297068.

[103] Monahan GE. State of the Art—A Survey of Partially Observable Markov Decision Processes: Theory, Models, and Algorithms. *Management Science*. 1982;28(1): 1–16. https://doi.org/10.1287/mnsc.28.1.1.

[104] Kaelbling LP, Littman ML, Cassandra AR. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*. 1998;101(1-2): 99-134. https://doi.org/10.1016/s0004-3702(98)00023-x.

[105] Prentice S, Roy N. The Belief Roadmap: Efficient Planning in Belief Space by Factoring the Covariance. *The International Journal of Robotics Research*. 2009;28(11-12): 1448–1465. https://doi.org/10.1177/0278364909341659.

[106] van den Berg J, Patil S, Alterovitz R. Motion planning under uncertainty using iterative local optimization in belief space. *The International Journal of Robotics Research*. 2012;31(11): 1263–1278. https://doi.org/10.1177/0278364912456319.

[107] Indelman V, Carlone L, Dellaert F. Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments. *The International Journal of Robotics Research*. 2015;34(7): 849–882. https://doi.org/10.1177/0278364914561102.

[108] Kondo K, Tordesillas J, How JP. SANDO: Safe Autonomous Trajectory Planning for Dynamic Unknown Environments. ArXiv [Preprint]. Version 1. 2026. https://arxiv.org/abs/2604.07599.

[109] Brian Y. A Frontier-Based Approach for Autonomous Exploration. *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation'*. IEEE; 1997. https://doi.org/10.1109/CIRA.1997.613851

[110] Placed JA, Strader J, Carrillo H, Atanasov N, Indelman V, Carlone L, *et al.* A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers. *IEEE Transactions on Robotics*. 2023;39(3): 1686–1705. https://doi.org/10.1109/tro.2023.3248510.

[111] Kober J, Bagnell JA, Peters J. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*. 2013;32(11): 1238–1274. https://doi.org/10.1177/0278364913495721.

[112] Moerland TM, Broekens J, Plaat A, Jonker CM. Model-based Reinforcement Learning: A Survey. *Foundations and Trends in Machine Learning*. 2023;16(1): 1–118. https://doi.org/10.1561/2200000086.

[113] Brunke L, Greeff M, Hall AW, Yuan Z, Zhou S, Panerati J, *et al.* Safe Learning in Robotics: From Learning-Based Control to Safe Reinforcement Learning. *Annual Review of Control, Robotics, and Autonomous Systems*. 2022;5(1). https://doi.org/10.1146/annurev-control-042920-020211.

[114] Tabor S, Guilliard I, Kolobov A. ArduSoar: An Open-Source Thermalling Controller for Resource-Constrained Autopilots. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE; 2018. p. 6255–6262. https://doi.org/10.1109/iros.2018.8593510.

[115] Guilliard I, Rogahn R, Piavi J, Kolobov A. Autonomous Thermalling as a Partially Observable Markov Decision Process. *14th Robotics: Science and Systems, RSS 2018*. MIT Press Journals; 2018. https://doi.org/10.15607/rss.2018.xiv.068.

[116] Reddy G, Wong-Ng J, Celani A, Sejnowski TJ, Vergassola M. Glider soaring via reinforcement learning in the field. *Nature*. 2018;562(7726): 236–239. https://doi.org/10.1038/s41586-018-0533-0.

[117] Cui Y, Yan D, Wan Z. Study on the Glider Soaring Strategy in Random Location Thermal Updraft via Reinforcement Learning. *Aerospace*. 2023;10(10): 834. https://doi.org/10.3390/aerospace10100834.

[118] Zhao J, Li J, Zeng L. Energy-Harvesting Strategy Investigation for Glider Autonomous Soaring Using Reinforcement Learning. *Aerospace*. 2023;10(10): 895–895. https://doi.org/10.3390/aerospace10100895.

[119] Kumar V, Michael N. Opportunities and challenges with autonomous micro aerial vehicles. *The International Journal of Robotics Research*. 2012;31(11): 1279–1291. https://doi.org/10.1177/0278364912455954.

[120] Coletti C, Williams K. Real-Time Control Interface for Research and Development Using Commercial-Off-The-Shelf (COTS) Mobile Robotics and UAVs. *AIAA SCITECH 2024 Forum*. Reston, Virginia: American Institute of Aeronautics and Astronautics; 2024. https://doi.org/10.2514/6.2024-0235.

[121] Cheng Z, West R, Einstein C. End-to-End Analysis and Design of a Drone Flight Controller. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2018;37(11): 2404–2415. https://doi.org/10.1109/tcad.2018.2857399.

[122] Windolf M, Götzen N, Morlock M. Systematic accuracy and precision analysis of video motion capturing systems—exemplified on the Vicon-460 system. *Journal of Biomechanics*. 2008;41(12): 2776–2780. https://doi.org/10.1016/j.jbiomech.2008.06.024.

[123] Merriaux P, Dupuis Y, Boutteau R, Vasseur P, Savatier X. A Study of Vicon System Positioning Performance. *Sensors*. 2017;17(7): 1591. https://doi.org/10.3390/s17071591.

[124] Mellinger D, Kumar V. Minimum snap trajectory generation and control for quadrotors. *2011 IEEE International Conference on Robotics and Automation*. IEEE; 2011. https://doi.org/10.1109/icra.2011.5980409.

[125] Schoellig AP, Mueller FL, D'Andrea R. Optimization-based iterative learning for precise quadrocopter trajectory tracking. *Autonomous Robots*. 2012;33(1-2): 103–127. https://doi.org/10.1007/s10514-012-9283-2.

[126] Zhang W, Branicky MS, Phillips SM. Stability of networked control systems. *IEEE Control Systems Magazine*. 2001;21(1): 84–99. https://doi.org/10.1109/37.898794.

[127] Miao Y, Shen W, Cui H, Mitra S. FalconWing: An Ultra-Light Indoor Fixed-Wing UAV Platform for Vision-Based Autonomy. ArXiv [Preprint]. Version 3. 2025. https://doi.org/10.48550/arXiv.2505.01383.

[128] Reid I, Ritchie J, Moore J, Sutherland B, Snow G, Tokumaru P, *et al.* ROSplane 2.0: A Fixed-Wing Autopilot for Research. ArXiv [Preprint]. Version 2. 2026; https://doi.org/10.48550/arXiv.2510.01041.

[129] Pickem D, Glotfelter P, Wang L, Mote M, Ames A, Feron E, *et al.* The Robotarium: A remotely accessible swarm robotics research testbed. *2017 IEEE International Conference on Robotics and Automation (ICRA).* IEEE; 2017. p. 1699–1706. https://doi.org/10.1109/ICRA.2017.7989200.

[130] Tani J, Daniele AF, Bernasconi G, Camus A, Petrov A, Courchesne A, *et al.* Integrated Benchmarking and Design for Reproducible and Accessible Evaluation of Robotic Agents. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE; 2020. p. 6229–6236. https://doi.org/10.1109/iros45743.2020.9341677.

[131] Dasari S, Wang J, Hong J, Bahl S, Lin Y, Wang A, *et al.* RB2: Robotic Manipulation Benchmarking with a Twist. ArXiv [Preprint]. Version 2. 2022. https://doi.org/10.48550/arXiv.2203.08098.

[132] Lupashin S, Schöllig A, Hehn M, D'Andrea R. The Flying Machine Arena as of 2010. *2011 IEEE International Conference on Robotics and Automation.* IEEE; 2011. https://doi.org/10.1109/icra.2011.5980308.

[133] Srivastava K, Kulkarni R, Velmurugan M, Sanker NJ. VizFlyt: Perception-centric Pedagogical Framework For Autonomous Aerial Robots. *2025 IEEE International Conference on Robotics and Automation (ICRA).* IEEE; 2025. p. 9920–9927. https://doi.org/10.1109/icra55743.2025.11128463.

[134] Kaya YF, Häusermann D, Nguyen PH, Orr L, Kovač M. The DroneHub: a multi-environment testbed for sustainability robotics. *npj Robotics.* 2026;4(1). https://doi.org/10.1038/s44182-025-00054-z.

[135] Vicon Motion Systems Limited. *Vicon Tracker User Guide.* https://help.vicon.com/space/Tracker44/376309877/PDF+downloads+for+Vicon+Tracker [Accessed 29th November 2025].

[136] Clarke International. *Clarke CAMAX30 30" Extra High Output Drum Fan.* https://www.clarkeinternational.com/p/clarke-camax30-extra-high-output-drum-fan [Accessed 1st December 2025].

[137] BENETECH. *Hot Wire Anemometer GM8903.* http://www.benetechco.net/en/products/gm8903.html [Accessed 20th January 2026].

[138] Hultmark M, Smits AJ. Temperature corrections for constant temperature and constant current hot-wire anemometers. *Measurement Science and Technology.* 2010;21(10): 105404. https://doi.org/10.1088/0957-0233/21/10/105404.

[139] FrSky. *Instruction Manual for FrSky 2.4GHz Taranis Q X7/X7S ACCESS.* 2019. https://www.frsky-rc.com/wp-content/uploads/Downloads/Amanual/X7%20X7S%20ACCESS%20Manual.pdf [Accessed 19th March 2026].

[140] FrSky. *Instruction Manual for FrSky ARCHER PLUS R6/GR6 Receiver.* Version 1. 2023. https://www.frsky-rc.com/wp-content/uploads/Downloads/Amanual/ARCHER%20PLUS%20R6%20GR6%20Manual.pdf [Accessed 19th March 2026].

[141] Arduino. *Arduino Nano 33 IoT datasheet.* 2026. https://docs.arduino.cc/resources/datasheets/ABX00027-datasheet.pdf [Accessed 11th June 2026].

[142] UCTRONICS. *24MHz 8CH USB Logic Analyzer.* Revision 1. 2018. https://www.uctronics.com/download/Amazon/U6041.pdf [Accessed 27th March 2026].

[143] Stemedu. *Micro 2.1g Servo DM-S0020 super mini digital servo.* https://www.amazon.com/dp/B0DXPYPV9Z [Accessed 6th March 2025].

[144] Jongeneel M, Saccon A. Geometric Savitzky-Golay Filtering of Noisy Rotations on SO(3) with Simultaneous Angular Velocity and Acceleration Estimation. *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE; 2022. https://doi.org/10.1109/iros47612.2022.9981409.

[145] Åström KJ, Wittenmark B. *Computer-controlled systems: theory and design.* Mineola, NY: Dover Publications; 2011.

[146] Yang Y, Sciacchitano A, Veldhuis LLM, Eitelberg G. Analysis of propeller-induced ground vortices by particle image velocimetry. *Journal of Visualization*. 2017;21(1): 39–55. https://doi.org/10.1007/s12650-017-0439-1.

[147] Hodson TO. Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development*. 2022;15(14): 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022.

[148] Huber PJ, Ronchetti EM. *Robust Statistics*. Hoboken, NJ: John Wiley & Sons; 2011.

[149] Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*. 2020;17(3): 261–272. https://www.nature.com/articles/s41592-019-0686-2.

[150] Ugray Z, Lasdon L, Plummer J, Glover F, Kelly J, Martí R. Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization. *INFORMS Journal on Computing*. 2007;19(3): 328–340. https://doi.org/10.1287/ijoc.1060.0175.

[151] Hoerl AE, Kennard RW. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*. 1970;12(1): 55–67. https://doi.org/10.1080/00401706.1970.10488634.

[152] Bertsekas DP. On Penalty and Multiplier Methods for Constrained Minimization. *SIAM Journal on Control and Optimization*. 1976;14(2): 216–235. https://doi.org/10.1137/0314017.

[153] Vandeven H. Family of spectral filters for discontinuous problems. *Journal of Scientific Computing*. 1991;6(2): 159–192. https://doi.org/10.1007/bf01062118.

[154] Charbonnier P, Blanc-Feraud L, Aubert G, Barlaud M. Two deterministic half-quadratic regularization algorithms for computed imaging. *Proceedings of 1st International Conference on Image Processing*. 1994. https://doi.org/10.1109/icip.1994.413553.

[155] Black MJ, Anandan P. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Computer Vision and Image Understanding*. 1996;63(1): 75–104. https://doi.org/10.1006/cviu.1996.0006.

[156] Fritsch FN, Carlson RW. Monotone Piecewise Cubic Interpolation. *SIAM Journal on Numerical Analysis*. 1980;17(2): 238–246. https://doi.org/10.1137/0717021.

[157] Transtrum MK, Machta BB, Sethna JP. Geometry of nonlinear least squares with applications to sloppy models and optimization. *Physical Review E*. 2011;83(3). https://doi.org/10.1103/physreve.83.036701.

[158] Forrester AIJ, Keane AJ. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*. 2009;45(1-3): 50–79. https://doi.org/10.1016/j.paerosci.2008.11.001.

[159] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 2011;12(85): 2825-2830. http://jmlr.org/papers/v12/pedregosa11a.html.

[160] Bousquet O, Von Luxburg U, Rätsch G. *Advanced Lectures on Machine Learning*. Berlin Heidelberg: Springer; 2004.

[161] Bishop CM. *Pattern Recognition and Machine Learning*. New York, NY: Springer; 2006.

[162] Roberts DR, Bahn V, Ciuti S, Boyce MS, Elith J, Guillera-Arroita G, *et al.* Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*. 2017;40(8): 913–929. https://doi.org/10.1111/ecog.02881.

[163] Arlot S, Celisse A. A survey of cross-validation procedures for model selection. Statistics Surveys. 2010;4(0): 40–79. https://doi.org/10.1214/09-ss054.

[164] Wipf D, Nagarajan S. A new view of automatic relevance determination. *Advances in Neural Information Processing Systems 20 (NIPS 2007)*. 2007. https://proceedings.neurips.cc/paper_files/paper/2007/file/9c01802ddb981e6bcfbec0f0516b8e35-Paper.pdf.

[165] Andrianakis I, Challenor PG. The effect of the nugget on Gaussian process emulators of computer models. *Computational Statistics & Data Analysis*. 2012;56(12): 4215–4228. https://doi.org/10.1016/j.csda.2012.04.020.

[166] Adrian L, Joseph L. Dynamic interaction of multiple buoyant jets. *Journal of Fluid Mechanics*. 2012;708: 539–575. https://doi.org/10.1017/jfm.2012.332.

[167] Easy Composites Ltd. *Carbon Fibre Pultrusions (CFP): Technical Datasheet*. Version 1. 2024. https://media.easycomposites.co.uk/datasheets/EC-TDS-Carbon-Fibre-Pultrusions.pdf [Accessed 24th February 2026].

[168] Bambu Lab. *Bambu Filament Technical Data Sheet V2.0: PLA-CF*. Version 2. https://store.bblcdn.com/842f399d4e274507953a5231126ec8c1.pdf [Accessed 23rd February 2026].

[169] Bambu Lab. *Bambu Filament Technical Data Sheet: PAHT-CF*. Version 2. https://store.bblcdn.com/7166b20bc68f4c87b32c972b856eeb4b.pdf [Accessed 23rd February 2026].

[170] Sharpe P. *Accelerating Practical Engineering Design Optimization with Computational Graph Transformations*. [Doctoral Thesis] [Massachusetts Institute of Technology]; 2024. https://hdl.handle.net/1721.1/157809.

[171] Andersson JAE, Gillis J, Horn G, Rawlings JB, Diehl M. CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*. 2018;11(1): 1–36. https://doi.org/10.1007/s12532-018-0139-4.

[172] Wächter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*. 2006;106(1): 25–57. https://doi.org/10.1007/s10107-004-0559-y.

[173] Kurtulus DF. Critical Angle and Fundamental Frequency of Symmetric Airfoils at Low Reynolds Numbers. *Journal of Applied Fluid Mechanics*. 2022;15(3): 723–735. https://doi.org/10.47176/jafm.15.03.33099.

[174] Sadraey MH. *Aircraft Design: A Systems Engineering Approach*. Chichester: John Wiley & Sons; 2013.

[175] Niu MC. *Airframe Structural Design: Practical Design Information and Data on Aircraft Structures*. Hong Kong: Hong Kong Conmilit Press limited; 2011.

[176] Beyer HG, Sendhoff B. Robust optimization – A comprehensive survey. *Computer Methods in Applied Mechanics and Engineering*. 2007;196(33-34): 3190–3218. https://doi.org/10.1016/j.cma.2007.03.003.

[177] Yao W, Chen X, Luo W, van Tooren M, Guo J. Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. *Progress in Aerospace Sciences*. 2011;47(6): 450–479. https://doi.org/10.1016/j.paerosci.2011.05.001.

[178] Miettinen K. *Nonlinear Multiobjective Optimization*. Springer Science & Business Media; 1999.

[179] Lewis K, Mistree F. Designing top-level aircraft specifications - A decision-based approach to a multiobjective, highly constrained problem. *36th Structures, Structural Dynamics and Materials Conference*. 1995. https://doi.org/10.2514/6.1995-1431.

[180] Martins J, Ning A. *Engineering Design Optimization*. Cambridge: Cambridge University Press; 2021.

[181] WEST SYSTEM. *105 Epoxy Resin with 205 Fast Hardener: Technical Data Sheet*. Revision 2. 2024. https://eu.westsystem.com/app/uploads/2024/08/WEST-SYSTEM-105-Epoxy-Resin-with-205-Fast-Hardener-Rev-2.pdf [Accessed 25th February 2026].

[182] 3M. *Tartan Filament Tape 8954: Product Data Sheet*. 2009. https://multimedia.3m.com/mws/media/860314O/3m-tartan-filament-tape-8954-product-data-sheet-english.pdf [Accessed 25th February 2026].

[183] Alloy Wire International. *Stainless Steel 304: Technical Datasheet*. AWS 161, Revision 2. 2021. https://www.alloywire.com/wp-content/uploads/2021/06/AW_Datasheet_STAINLESS-STEEL-304_Rev-2.pdf [Accessed 13th March 2026].

[184] Anderson JD. *Fundamentals of Aerodynamics*. London: Mcgraw-Hill Higher Education; 2007.

[185] Selig MS. Real-Time Flight Simulation of Highly Maneuverable Unmanned Aerial Vehicles. *Journal of Aircraft*. 2014;51(6): 1705–1725. https://doi.org/10.2514/1.c032370.

[186] Cunis T, Burlion L, Condomines JP. Piecewise Polynomial Modeling for Control and Analysis of Aircraft Dynamics Beyond Stall. *Journal of Guidance, Control, and Dynamics*. 2019;42(4): 949–957. https://doi.org/10.2514/1.g003618.

[187] Oettershagen P, Stastny T, Hinzmann T, Rudin K, Mantel T, Melzer A, *et al.* Robotic technologies for solar-powered UAVs: Fully autonomous updraft-aware aerial sensing for multiday search-and-rescue missions. *Journal of Field Robotics*. 2017;35(4): 612–640. https://doi.org/10.1002/rob.21765.

[188] Wang Y, Wynn A, Palacios R. Nonlinear Aeroelastic Control of Very Flexible Aircraft Using Model Updating. *Journal of Aircraft*. 2018;55(4): 1551–1563. https://doi.org/10.2514/1.c034684.

[189] Rowley CW, Dawson STM. Model Reduction for Flow Analysis and Control. *Annual Review of Fluid Mechanics*. 2017;49(1): 387–417. https://doi.org/10.1146/annurev-fluid-010816-060042.

[190] Lucia DJ, Beran PS, Silva WA. Reduced-order modeling: new approaches for computational physics. *Progress in Aerospace Sciences*. 2004;40(1-2): 51–117. https://doi.org/10.1016/j.paerosci.2003.12.001.

[191] Fathy HK, Reyer JA, Papalambros PY, Ulsov AG. On the coupling between the plant and controller optimization problems. *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*. IEEE; 2001. p. 1864–1869. https://doi.org/10.1109/ACC.2001.946008.

[192] Garcia-Sanz M. Control Co-Design: an engineering game changer. *Advanced Control for Applications: Engineering and Industrial Systems*. 2019. https://doi.org/10.1002/adc2.18.

[193] Fasel U, Keidel D, Molinari G, Ermanni P. Aerostructural optimization of a morphing wing for airborne wind energy applications. *Smart Materials and Structures*. 2017;26(9): 095043. https://doi.org/10.1088/1361-665x/aa7c87.

[194] Kenway GKW, Martins JRRA. Multipoint High-Fidelity Aerostructural Optimization of a Transport Aircraft Configuration. *Journal of Aircraft*. 2014;51(1): 144–160. https://doi.org/10.2514/1.c032150.

[195] Adler EJ, Martins JRRA. Efficient Aerostructural Wing Optimization Considering Mission Analysis. *Journal of Aircraft*. 2023;60(3): 800–816. https://doi.org/10.2514/1.c037096.

[196] Kaneko S, Martins JRRA. Simultaneous Design and Trajectory Optimization Strategies for Computationally Expensive Models. *AIAA Journal*. 2025;63(2): 420–438. https://doi.org/10.2514/1.J063976.

[197] Fasel U, Tiso P, Keidel D, Ermanni P. Concurrent Design and Flight Mission Optimization of Morphing Airborne Wind Energy Wings. *AIAA Journal*. 2021;59(4): 1254–1268. https://doi.org/10.2514/1.j059621.

[198] Li D, Zhao S, Da Ronch A, Xiang J, Drofelnik J, Li Y, et al. A review of modelling and analysis of morphing wings. *Progress in Aerospace Sciences*. 2018;100: 46–62. https://doi.org/10.1016/j.paerosci.2018.06.002.

[199] Chen WH. Goal-Oriented Control Systems (GOCS): From HOW to WHAT. *IEEE/CAA Journal of Automatica Sinica*. 2024;11(4): 816–819. https://doi.org/10.1109/jas.2024.124323.

[200] Notter S, Groß P, Schrapel P, Fichter W. Multiple Thermal Updraft Estimation and Observability Analysis. *Journal of Guidance Control and Dynamics*. 2019;43(3): 490–503. https://doi.org/10.2514/1.g004205.

[201] Liem RP, Mader CA, Martins JRRA. Surrogate models and mixtures of experts in aerodynamic performance prediction for aircraft mission analysis. *Aerospace Science and Technology*. 2015;43: 126–151. https://doi.org/10.1016/j.ast.2015.02.019.

[202] Liu C, Chen WH. Disturbance Rejection Flight Control for Small Fixed-Wing Unmanned Aerial Vehicles. *Journal of Guidance, Control, and Dynamics*. 2016;39(12): 2810–2819. https://doi.org/10.2514/1.g001958.

[203] Hewing L, Wabersich KP, Menner M, Zeilinger MN. Learning-Based Model Predictive Control: Toward Safe Learning in Control. *Annual Review of Control, Robotics, and Autonomous Systems*. 2020;3(1): 269–296. https://doi.org/10.1146/annurev-control-090419-075625.

[204] GAONENG. *GAONENG GNB LiHV 1S 3.8V 300mAh 80C PH2.0 Cabled LiPo Battery*. https://www.gaoneng.shop/products/gaoneng-gnb-lihv-1s-3.8v-300mah-80c-ph2.0-cabled-lipo-battery [Accessed 9th December 2025].

[205] Pololu. *Pololu 5V Step-Up/Step-Down Voltage Regulator S7V7F5*. https://www.pololu.com/product/2119 [Accessed 9th December 2025].

A Reproducibility, Research Integrity, and Version Record

A.1 Reproducibility Statement

This thesis is accompanied by a reproducibility package containing source code, configurations, processed datasets, calibration artefacts, frozen controller inputs, plotting scripts, and editable source files required to inspect and reproduce the reported computational and experimental results. It is archived on Zenodo at https://doi.org/10.5281/zenodo.20927007 and browsable at https://github.com/GH-X-ST/Nausicaa.

The repository is organised by workflow instead of a single monolithic software package. Each major project folder provides a local `environment.toml` describing the relevant software requirements and an `implement_sequence.txt` summarising the execution order used to generate or analyse the results. The root `README.md` provides the repository-level entry point, while the folder-level files define the reproducibility procedure for the corresponding thermal modelling, glider design, control, flight test, and latency analysis components.

Computational results based on simulation, modelling, optimisation, system identification, and post-processing can be reproduced by executing the supplied scripts with the corresponding configuration files and fixed random seeds where stochastic methods are used. The software environment was tested on Windows 11 25H2 with Python 3.12.11, MATLAB R2026a, Arduino IDE 2.3.8 and Vicon Tracker 3.9. Minor numerical differences across platforms are expected because of standard floating point, solver, and library version differences.

Experimental results reported in Appendices B, C, F and G depend on physical measurements in the Flight Arena and therefore cannot be regenerated deterministically without the same hardware, calibration state, firmware, and experimental setup. To support verification, the archived materials include the measurement logs, preprocessing scripts, calibration records, frozen inputs, and plotting code used to produce the reported statistics, heat maps, validation figures, and latency summaries.

Figures designed by the author, including conceptual illustrations, workflow diagrams, and schematics, were prepared using standard graphical software such as Microsoft PowerPoint, Adobe Photoshop, Adobe Illustrator, and Union. Their preparation drew on the author's prior graphic design experience and did not involve external assistance or the use of generative AI tools.

A.2 Declaration of Generative AI Use

Generative AI tools were used in a limited and documented manner to support the preparation of this thesis, consistent with Imperial College London guidance on responsible use [12].

Specifically, generative AI was used for: (i) proofreading, polishing, and grammar refinement of draft text written by the author, (ii) restructuring of LaTeX formatting, and (iii) programming assistance for non-novel tasks (e.g. refactoring and debugging). All technical decisions, modelling assumptions, experimental procedures, results, and interpretations remain the author's work.

Accordingly, generative AI was not used to: (i) fabricate, alter, or otherwise misrepresent data, including experimental measurements, simulation outputs, and numerical results, (ii) write code to produce figures or tables without author verification and reproducibility from the project database, (iii) create citations, or (iv) produce assessment content in a manner that would replace the author's original analysis and understanding.

All AI-assisted content was reviewed and edited by the author. Code suggestions were adopted only after inspection, execution, and validation against expected behaviour. Textual suggestions were checked for technical accuracy and consistency with the project record. No confidential, personal, or restricted information (e.g. unpublished third-party materials) was entered into generative AI tools.

[12] Available at https://www.imperial.ac.uk/about/education/resources/ai-education-hub/generative-ai-principles.

B Supplementary State Processing and Timing Characterisation

B.1 Command Interface Hardware and Latency Characterisation

B.1.1 Hardware and Signal Path Definitions

Table B.1 Command interface components, specifications, and test settings.

Component	Use	Relevant specification or test setting	Source
Vicon Vantage cameras	Motion capture camera network	Gigabit-Ethernet camera connection carrying power, data, and synchronisation; provides high-speed optical tracking of reflective markers	
Vicon host PC, Vicon Tracker software, and DataStream SDK	Reflective marker reconstruction and state streaming	Vicon Tracker software processed the camera data, while the DataStream SDK provided programmatic access to the reconstructed Vicon data stream	[123]
Ground computer and Vicon DataStream link	State reception and host-time alignment	Vicon DataStream SDK connection to the Vicon host computer in ServerPush mode; segment data enabled with buffer size one; Vicon capture time recorded as host read time minus reported Vicon latency	
Arduino Nano 33 IoT	Direct-servo controller and COTS RC bridge	3.3 V logic with non-5 V-tolerant I/O; Wi-Fi and Micro USB interfaces used in different timing tests; D9 – D12 generated four 50 Hz direct servo PWM outputs; D3 generated the trainer port PPM signal, and D2 provided the reference pulse for clock alignment	[141]
Ground computer to Arduino Nano 33 IoT command links	Command dispatch and firmware telemetry logging	Direct servo tests sent UDP packets to the Arduino Nano 33 IoT over Wi-Fi; COTS RC tests used USB serial at 1 Mbaud; 15-byte command packets were sent every 0.02 s, with command, telemetry, and event logs saved for analysis	
GNB LiHV battery	Onboard and test power source	1S LiHV, 3.8 V nominal voltage, 300 mAh capacity, and 80C discharge rating	[204]
Pololu S7V7F5 regulator	Direct servo test power regulation	2.7 – 11.8 V input range and fixed 5 V output for the servo side supply rail	[205]
FrSky Taranis Q X7 transmitter	COTS RC transmitter and trainer port interface	ACCESS/ACCST transmitter; wired trainer port retained; 50 Hz PPM signal injected through the 3.5 mm TS trainer input in the timing test	[139]
FrSky Archer Plus R6 receiver	Onboard RC receiver and COTS timing endpoint	Six PWM channel ports, 3.5 – 10 V operating range, $<$ 65 mA current at 5 V, and ACCESS/ACCST D16 compatibility	[140]
AITRIP logic analyser	Electrical timing capture	Eight digital channels; 24 MHz maximum sampling specification, with 4 MHz used in the present setting	[142]
DOMAN S0020 micro servo	Control surface actuator	Digital 2.1 g servo; 4.8–6.0 V operating range; speed 0.10 s/60° and torque 0.20 kg cm at 4.8 V, or 0.08 s/60° and 0.25 kg cm at 6.0 V; 500–2500 μs PWM range	[143]
Direct servo command path PWM mapping	Direct command path timing signal	Arduino pins D9 – D12 generated 50 Hz PWM signals captured on analyser channels CH0 – CH3 for left aileron, right aileron, rudder, and elevator	-
COTS RC command path PPM/PWM mapping	Transmitter-receiver timing signal	Arduino D3 generated the 50 Hz trainer PPM signal; D2 provided clock alignment; receiver CH1 – CH4 provided the captured PWM outputs	-

B.1.2 Direct Servo Command Path Latency Test

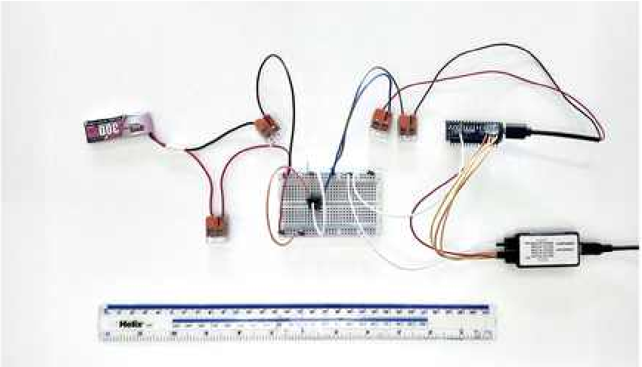


Fig. B.1. Hardware implementation of the direct servo command path latency test setup.

Algorithm B.1 Direct servo command path latency extraction.

Input: Direct servo test firmware F_D, command schedule $\mathcal{U}$, capture set $\mathcal{Y}_D$, matching tolerances τ_D, and run mode r.

Output: Accepted transition set $\mathcal{E}_D$, latency segment table Λ_D, and rejection audit A_D.

Upload F_D and verify communication with the direct servo controller
$\mathcal{E}_D \leftarrow \emptyset$, $A_D \leftarrow \emptyset$
if $r =$ new **then**
    $\mathcal{U} \leftarrow$ generate a repeatable surface transition schedule
    $(\mathcal{G}_D, \mathcal{B}_D) \leftarrow$ execute $\mathcal{U}$ through the direct servo command path
    $\mathcal{Y}_D \leftarrow$ capture reference and servo output pulse trains
else
    $(\mathcal{G}_D, \mathcal{B}_D, \mathcal{Y}_D, \mathcal{U}) \leftarrow$ load the selected recorded run
$\phi_D \leftarrow$ fit ground computer, controller, and capture clock maps from $\mathcal{G}_D$, $\mathcal{B}_D$, and $\mathcal{Y}_D$
$\mathcal{C}_D \leftarrow$ decode command transitions and servo output transitions under ϕ_D
for *each command transition* $c_i \in \mathcal{C}_D$ **do**
    $a_i \leftarrow$ unique controller apply event associated with c_i
    $y_i \leftarrow$ first admissible servo output transition after a_i within τ_D
    if a_i *and* y_i *satisfy uniqueness, ordering, and pulse consistency tests* **then**
        $\lambda_i \leftarrow \{t_{\text{rx}} - t_{\text{cmd}},\ t_{\text{apply}} - t_{\text{rx}},\ t_{\text{out}} - t_{\text{apply}},\ t_{\text{out}} - t_{\text{sched}}\}$
        Add (c_i, λ_i) to $\mathcal{E}_D$
    else
        Add (c_i, reason) to A_D
$\Lambda_D \leftarrow$ aggregate latency segments from $\mathcal{E}_D$ by surface and over all surfaces
return $\mathcal{E}_D$, Λ_D, and A_D

Table B.2 Per-run stage-by-stage electrical timing for the direct servo command path. The n_{match} column denotes accepted matched transition events.

Run	n_{match}	Path segment	Unit	median	P_{95}	P_{99}	σ
1	914	Ground command to dispatch	ms	0.26	1.14	4.76	0.84
		Ground dispatch to controller receive	ms	21.6	24.0	24.4	1.33
		Controller receive to PWM update	ms	0.20	0.21	0.22	0.01
		PWM update to next servo pulse	ms	19.9	29.3	30.4	5.93
		Ground command to servo pulse	ms	**42.3**	**51.7**	**52.7**	**5.90**
2	940	Ground command to dispatch	ms	0.20	0.41	0.94	0.15
		Ground dispatch to controller receive	ms	21.7	34.1	35.6	8.14
		Controller receive to PWM update	ms	0.20	0.21	0.22	0.01
		PWM update to next servo pulse	ms	12.2	21.8	23.1	6.02
		Ground command to servo pulse	ms	**34.2**	**51.7**	**56.8**	**10.3**
3	910	Ground command to dispatch	ms	0.27	0.90	3.36	0.97
		Ground dispatch to controller receive	ms	22.6	32.3	33.2	5.32
		Controller receive to PWM update	ms	0.21	0.30	0.31	0.04
		PWM update to next servo pulse	ms	12.5	21.8	23.0	5.97
		Ground command to servo pulse	ms	**36.5**	**50.7**	**54.3**	**8.14**
4	956	Ground command to dispatch	ms	0.20	0.64	1.96	0.85
		Ground dispatch to controller receive	ms	27.5	28.8	29.1	1.15
		Controller receive to PWM update	ms	0.20	0.21	0.22	0.01
		PWM update to next servo pulse	ms	20.5	30.2	31.5	5.92
		Ground command to servo pulse	ms	**48.6**	**58.1**	**58.9**	**5.86**
5	932	Ground command to dispatch	ms	0.34	2.24	7.38	1.22
		Ground dispatch to controller receive	ms	21.5	33.1	33.5	6.20
		Controller receive to PWM update	ms	0.20	0.21	0.22	0.01
		PWM update to next servo pulse	ms	14.7	24.5	25.3	6.03
		Ground command to servo pulse	ms	**38.1**	**52.0**	**57.3**	**8.25**

B.1.3 COTS RC Transmitter-Receiver Command Chain Latency Test

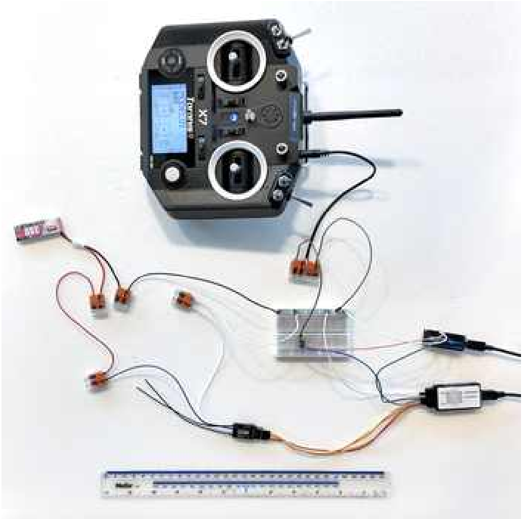


Fig. B.2. Hardware implementation of the COTS RC command chain latency test setup [13].

Algorithm B.2 COTS RC transmitter-receiver command chain latency extraction.

Input: COTS RC bridge firmware F_C, command schedule $\mathcal{U}$, trainer and receiver captures $\mathcal{Y}_C$, clock repair model ϕ_C, matching tolerances τ_C, and run mode r.

Output: Accepted COTS RC transition set $\mathcal{E}_C$, command chain latency table Λ_C, and audit A_C.

Upload F_C and verify the ground computer command link, trainer output, and receiver output
$\mathcal{E}_C \leftarrow \emptyset$, $A_C \leftarrow \emptyset$
if $r =$ new **then**
  $\mathcal{U} \leftarrow$ generate a repeatable COTS RC transition schedule
  $(\mathcal{G}_C, \mathcal{B}_C) \leftarrow$ execute $\mathcal{U}$ through the RC command path
  $\mathcal{Y}_C \leftarrow$ capture reference, trainer, and receiver pulse trains
else
  $(\mathcal{G}_C, \mathcal{B}_C, \mathcal{Y}_C, \mathcal{U}) \leftarrow$ load the selected recorded run
$\phi_C \leftarrow$ repair the controller clock from ground computer dispatch, board receive, and board commit events
$\mathcal{C}_C \leftarrow$ decode scheduled, trainer output, and receiver output transitions under ϕ_C
for *each scheduled transition* $c_i \in \mathcal{U}$ **do**
  $b_i \leftarrow$ associated board receive and commit pair
  $p_i \leftarrow$ associated trainer output transition
  $q_i \leftarrow$ associated receiver output transition
  if b_i, p_i, *and* q_i *satisfy uniqueness and* τ_C **then**
    $\lambda_i \leftarrow \{t_{\text{rx}} - t_{\text{cmd}}, t_{\text{commit}} - t_{\text{rx}}, t_{\text{trainer}} - t_{\text{commit}}, t_{\text{receiver}} - t_{\text{trainer}}\}$
    Add (c_i, λ_i) to $\mathcal{E}_C$
  else
    Add (c_i, reason) to A_C
$\Lambda_C \leftarrow$ aggregate command chain latency segments from $\mathcal{E}_C$ by surface and over all surfaces
Add clock repair residuals and rejected event counts to A_C
return $\mathcal{E}_C$, Λ_C, and A_C

---

[13] The transmitter was configured using the factory trainer port settings. For ACCESS-based FrSky transmitters running OpenTX, firmware version 2.3.15 or later is recommended for this setup, since the 2.3 branch includes a fix for trainer port detection on selected FrSky ACCESS radios, including the X7 Access and X9 Lite. The related OpenTX issue and fix are documented at https://github.com/opentx/opentx/issues/8670 and https://github.com/opentx/opentx/pull/8811.

Table B.3 Per-run stage-by-stage electrical timing for the COTS RC transmitter-receiver command chain. The n_{match} column denotes accepted matched transition events.

Run	n_{match}	Path segment	Unit	median	P_{95}	P_{99}	σ
1	919	Ground command to dispatch	ms	0.16	0.44	0.83	0.17
		Ground dispatch to Nano receive	ms	5.28	5.74	6.07	0.36
		Nano receive to PPM update	ms	16.5	17.1	17.2	0.43
		PPM update to next trainer frame	ms	4.39	20.0	20.1	8.10
		Trainer PPM to receiver PWM	ms	12.9	34.2	34.4	11.1
		Ground command to receiver PWM	ms	**37.8**	**57.8**	**57.8**	**7.17**
2	947	Ground command to dispatch	ms	0.16	0.47	1.15	0.17
		Ground dispatch to Nano receive	ms	6.64	7.07	7.26	0.26
		Nano receive to PPM update	ms	3.55	3.97	4.07	0.34
		PPM update to next trainer frame	ms	4.44	20.1	20.1	8.25
		Trainer PPM to receiver PWM	ms	6.98	28.5	48.6	12.9
		Ground command to receiver PWM	ms	**20.5**	**40.6**	**60.6**	**9.28**
3	912	Ground command to dispatch	ms	0.17	0.57	1.43	0.41
		Ground dispatch to Nano receive	ms	6.96	7.53	7.71	0.26
		Nano receive to PPM update	ms	11.6	11.9	12.0	0.52
		PPM update to next trainer frame	ms	4.44	20.1	20.2	8.23
		Trainer PPM to receiver PWM	ms	12.1	33.6	33.7	11.0
		Ground command to receiver PWM	ms	**34.0**	**54.0**	**54.0**	**6.97**
4	953	Ground command to dispatch	ms	0.17	0.72	1.57	0.58
		Ground dispatch to Nano receive	ms	7.07	7.49	7.68	0.22
		Nano receive to PPM update	ms	5.50	5.83	5.90	0.69
		PPM update to next trainer frame	ms	4.38	20.0	20.1	8.26
		Trainer PPM to receiver PWM	ms	5.39	43.8	46.9	15.7
		Ground command to receiver PWM	ms	**21.1**	**61.0**	**61.2**	**12.8**
5	941	Ground command to dispatch	ms	0.16	0.37	0.80	0.11
		Ground dispatch to Nano receive	ms	6.91	7.31	7.52	0.26
		Nano receive to PPM update	ms	14.6	15.0	15.1	0.30
		PPM update to next trainer frame	ms	1.48	4.48	4.52	1.63
		Trainer PPM to receiver PWM	ms	2.41	40.3	42.7	11.9
		Ground command to receiver PWM	ms	**25.2**	**64.7**	**65.4**	**11.8**

B.2 Implementation Details for Vicon-Based State Estimation

This appendix gives the state-processing details behind Section 3.3.1. Vicon motion capture system provides rigid-body position and attitude. The controller developed later in Chapter 6, however, requires the 15-state vector in (3.2). Position and attitude are first mapped into the arena frame defined in Fig. 3.1, translational velocity, angular rate, and actuator state are then estimated online.

B.2.1 Pose Calibration Convention

Let $\mathbf{p}_{V,k} = [x_{V,k},\, y_{V,k},\, z_{V,k}]^\top$ denote the raw Vicon position at frame k, and let $\boldsymbol{\eta}_{V,k} = [\phi_{V,k}, \theta_{V,k}, \psi_{V,k}]^\top$ denote the corresponding raw roll, pitch, and yaw angles.

During calibration, the airframe is held at a known arena-frame point $\mathbf{p}^\star$. If the valid Vicon samples collected during this calibration are indexed by $\mathcal{K}_c$, their mean is

$$\bar{\mathbf{p}}_V = \frac{1}{|\mathcal{K}_c|} \sum_{j \in \mathcal{K}_c} \mathbf{p}_{V,j}. \tag{B.1}$$

The session position offset is then

$$\Delta\mathbf{p} = \mathbf{p}^\star - \bar{\mathbf{p}}_V, \qquad \mathbf{p}_k = \mathbf{p}_{V,k} + \Delta\mathbf{p}. \tag{B.2}$$

The attitude correction accounts for the Vicon system to the controller sign convention and the calibrated roll, pitch, and yaw offsets. The processed attitude is

$$\boldsymbol{\eta}_k = \mathrm{wrap}_\pi\left(\mathbf{S}\boldsymbol{\eta}_{V,k} + \boldsymbol{\epsilon}_a\right), \tag{B.3}$$

or, equivalently,

$$\begin{bmatrix} \phi_k \\ \theta_k \\ \psi_k \end{bmatrix} = \mathrm{wrap}_\pi\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \phi_{V,k} \\ \theta_{V,k} \\ \psi_{V,k} \end{bmatrix} + \begin{bmatrix} \epsilon_\phi \\ \epsilon_\theta \\ \epsilon_\psi \end{bmatrix} \right), \tag{B.4}$$

where $\mathrm{wrap}_\pi(\cdot)$ is applied elementwise and maps each angle to $[-\pi, \pi)$:

$$\mathrm{wrap}_\pi(\alpha) = \alpha - 2\pi \left\lfloor \frac{\alpha + \pi}{2\pi} \right\rfloor. \tag{B.5}$$

The active calibration profile stores the position offset $\Delta\mathbf{p}$, the attitude sign matrix $\mathbf{S}$, the attitude offset $\boldsymbol{\epsilon}_a$, and a calibration hash recorded with each flight log.

B.2.2 Frame Timing and Translational Velocity

The Vicon stream provides both a host timestamp and, when available, a Vicon frame number [135]. The frame number time step is preferred because it follows the motion capture sampling clock:

$$\Delta t_k = \frac{n_k - n_{k-1}}{f_V}, \qquad f_V = 200\ \mathrm{Hz}, \tag{B.6}$$

where n_k is the Vicon frame number. If frame numbers are missing or not strictly increasing, the host timestamp difference is used instead. Duplicate or reordered frames are not used for derivative updates.

The raw arena frame velocity estimate is the finite difference of the calibrated position,

$$\tilde{\mathbf{v}}_k = \frac{\mathbf{p}_k - \mathbf{p}_{k-1}}{\Delta t_k}. \tag{B.7}$$

This finite-difference estimate is noisy because it is computed directly from successive measured positions.

Hence, a one-pole low-pass filter with cut-off frequency $f_c = 8\,\mathrm{Hz}$ is applied:

$$\mathbf{v}_k = \mathbf{v}_{k-1} + \alpha_k\left(\tilde{\mathbf{v}}_k - \mathbf{v}_{k-1}\right), \qquad \alpha_k = 1 - \exp(-2\pi f_c\,\Delta t_k)\,. \tag{B.8}$$

The controller uses translational velocity in the aerodynamic body frame. The arena frame is z-up, whereas the aerodynamic body-axis convention takes w as positive downward. The vertical component is therefore sign-reversed before the velocity is rotated into body axes:

$$\begin{bmatrix} u_{b,k} \\ v_{b,k} \\ w_{b,k} \end{bmatrix} = \mathbf{C}_{wb,k}^{\top}\,\mathbf{D}_z\mathbf{v}_k\,, \qquad \mathbf{D}_z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{B.9}$$

The body-to-world (arena) direction cosine matrix used in this rotation is formed from the processed attitude $\boldsymbol{\eta}_k = [\phi_k,\, \theta_k,\, \psi_k]^{\top}$:

$$\mathbf{C}_{wb,k} = \begin{bmatrix} c_{\theta,k}\,c_{\psi,k} & s_{\phi,k}\,s_{\theta,k}\,c_{\psi,k} - c_{\phi,k}\,s_{\psi,k} & c_{\phi,k}\,s_{\theta,k}\,c_{\psi,k} + s_{\phi,k}\,s_{\psi,k} \\ c_{\theta,k}\,s_{\psi,k} & s_{\phi,k}\,s_{\theta,k}\,s_{\psi,k} + c_{\phi,k}\,c_{\psi,k} & c_{\phi,k}\,s_{\theta,k}\,s_{\psi,k} - s_{\phi,k}\,c_{\psi,k} \\ -s_{\theta,k} & s_{\phi,k}\,c_{\theta,k} & c_{\phi,k}\,c_{\theta,k} \end{bmatrix}, \tag{B.10}$$

where $c_{\phi,k} = \cos\phi_k$, $s_{\phi,k} = \sin\phi_k$, and similarly for θ_k and ψ_k.

B.2.3 Angular Rate Estimation

The body angular rate is estimated on the rotation group $SO(3)$, the space of valid 3D rotation matrices, instead of by directly differentiating Euler angles. Motivated by the geometric treatment of noisy rotations by Jongeneel and Saccon [144], the implementation uses a logarithm map estimate of the frame-to-frame rotation, followed by clipping and one-pole filtering.

For two successive body-to-world (arena) direction cosine matrices, the incremental body-axis rotation is expressed by

$$\tilde{\boldsymbol{\omega}}_{b,k} = \frac{1}{\Delta t_k}\,\mathrm{Log}(Q_k)^{\vee}, \qquad Q_k = \mathbf{C}_{wb,k-1}^{\top}\,\mathbf{C}_{wb,k}, \tag{B.11}$$

where $\mathrm{Log}(Q_k)^{\vee}$ is the rotation vector associated with the incremental rotation from sample $k-1$ to sample k. The resulting body-rate estimate is ordered as $\tilde{\boldsymbol{\omega}}_{b,k} = [\tilde{p}_{b,k},\, \tilde{q}_{b,k},\, \tilde{r}_{b,k}]^{\top}$.

Occasional frame-to-frame tracking errors can produce unrealistically large angular rate estimates. To prevent these spikes from entering the controller state, the raw estimate is first limited component-wise:

$$\tilde{\boldsymbol{\omega}}_{b,k}^{c} = \mathrm{clip}\left(\tilde{\boldsymbol{\omega}}_{b,k},\, -\boldsymbol{\omega}_{\max},\, \boldsymbol{\omega}_{\max}\right). \tag{B.12}$$

where each angular-rate component is bounded to $\pm 6.0\,\mathrm{rad\,s^{-1}}$. This clipped estimate is smoothed with the same one-pole filter used for translational velocity in (B.8):

$$\boldsymbol{\omega}_{b,k} = \boldsymbol{\omega}_{b,k-1} + \alpha_k\left(\tilde{\boldsymbol{\omega}}_{b,k}^{c} - \boldsymbol{\omega}_{b,k-1}\right). \tag{B.13}$$

The filtered value is then stored in the controller state as $[p_{b,k},\, q_{b,k},\, r_{b,k}]^{\top} = \boldsymbol{\omega}_{b,k}$.

B.2.4 Controller State Assembly

At each valid Vicon frame, the processed position, attitude, velocity, angular-rate, and actuator-state estimates are packed into the 15-state vector required by the controller:

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{p}_k^{\top} & \boldsymbol{\eta}_k^{\top} & \mathbf{v}_{b,k}^{\top} & \boldsymbol{\omega}_{b,k}^{\top} & \boldsymbol{\delta}_k^{\top} \end{bmatrix}^{\top}, \tag{B.14}$$

where

$$\mathbf{v}_{b,k} = [u_{b,k},\, v_{b,k},\, w_{b,k}]^{\top}, \qquad \boldsymbol{\omega}_{b,k} = [p_{b,k},\, q_{b,k},\, r_{b,k}]^{\top}, \qquad \boldsymbol{\delta}_k = [\delta_{a,k},\, \delta_{e,k},\, \delta_{r,k}]^{\top}. \tag{B.15}$$

The control surface state $\boldsymbol{\delta}_k$ is not measured by Vicon during flight, it is propagated from the command history using the command to control surface timing model introduced in Appendix B.3.

The runtime constants used by the state processing routine are listed in Table B.4.

Table B.4 Runtime constants for Vicon-based state processing.

Quantity	Symbol	Unit	Value
Vicon tracking rate	f_V	Hz	200
Velocity and angular rate filter cut-off frequency	f_c	Hz	8
Angular rate clipping bound	$\omega_{\max}$	$\mathrm{rad\,s^{-1}}$	6.0
Position offset, x	Δp_x	m	3.9965
Position offset, y	Δp_y	m	2.5229
Position offset, z	Δp_z	m	0.0362
Attitude sign, roll	$S_{\phi\phi}$	–	1
Attitude sign, pitch	$S_{\theta\theta}$	–	-1
Attitude sign, yaw	$S_{\psi\psi}$	–	-1
Attitude offset, roll	ϵ_ϕ	rad	7.44×10^{-4}
Attitude offset, pitch	ϵ_θ	rad	7.20×10^{-5}
Attitude offset, yaw	ϵ_ψ	rad	6.25×10^{-4}

The complete online state processing procedure is summarised in Algorithm B.3.

Algorithm B.3 Vicon state processing and controller state assembly

Input: Rigid-body Vicon stream, calibration point $\mathbf{p}^\star$, calibration profile, cutoff frequency f_c, and command-history actuator model.

Output: Processed controller state $\mathbf{x}_k$ and state-processing audit.

Average valid calibration samples to obtain $\bar{\mathbf{p}}_V$
Compute $\Delta\mathbf{p} = \mathbf{p}^\star - \bar{\mathbf{p}}_V$
Load attitude signs $\mathbf{S}$ and attitude offset $\boldsymbol{\epsilon}_a$
Initialise filtered velocity, filtered angular rate, actuator state, and audit flags
for *each Vicon frame k* **do**
    Read raw position $\mathbf{p}_{V,k}$, raw attitude $\boldsymbol{\eta}_{V,k}$, frame number, and timestamp
    if *the pose is invalid or occluded* **then**
        log the invalid frame, skip derivative updates, and continue
    Compute $\mathbf{p}_k$ using (B.2)
    Compute $\boldsymbol{\eta}_k$ using (B.3)
    Resolve Δt_k from Vicon frame timing or host timestamp fallback
    Estimate and filter translational velocity using (B.7)–(B.9)
    Estimate, clip, and filter angular rate using (B.11)–(B.13)
    Propagate actuator state from the command history model in Appendix B.3
    Assemble $\mathbf{x}_k$ using (B.14)
    Log timing source, validity, filtering status, and rate-limiting flags
return processed states and audit log

B.3 Control Surface Command Mapping and Actuator Timing

This appendix gives the command-side details behind Section 3.3.2. It defines how the controller's three aggregate commands are quantised, mapped to the four physical control surfaces, and converted into the actuator state model used in simulation and real flight state estimation. The Vicon state feedback delay is treated separately from the actuator model in the controller simulations in Chapter 6.

B.3.1 Command Lattice and Hardware Mapping

The controller outputs the normalised aggregate command vector $\mathbf{u}_k^{\mathrm{cmd}} = [u_{a,k}^{\mathrm{cmd}},\, u_{e,k}^{\mathrm{cmd}},\, u_{r,k}^{\mathrm{cmd}}]^\top$, where the entries correspond to aileron, elevator, and rudder action. Before use, each component is clipped to $[-1, 1]$ and rounded to the executable command lattice:

$$\mathcal{Q}_{0.2}(u) = 0.2 \ \mathrm{round}\left(\frac{\mathrm{sat}_{[-1,1]}(u)}{0.2}\right), \qquad \boldsymbol{\nu}_k = \begin{bmatrix} \mathcal{Q}_{0.2}(u_{a,k}^{\mathrm{cmd}}) \\ \mathcal{Q}_{0.2}(u_{e,k}^{\mathrm{cmd}}) \\ \mathcal{Q}_{0.2}(u_{r,k}^{\mathrm{cmd}}) \end{bmatrix}. \tag{B.16}$$

The resulting executable aggregate command is $\boldsymbol{\nu}_k = [\nu_{a,k},\, \nu_{e,k},\, \nu_{r,k}]^\top$.

The physical surface vector and the servo command vector are written in the receiver channel order, $\mathbf{s}_k = [s_{a_R,k},\, s_{a_L,k},\, s_{r,k},\, s_{e,k}]^\top$, which is demonstrated in Fig. 3.4. The aggregate command is expanded to this order as

$$\mathbf{s}_k^{\mathrm{phys}} = \mathbf{M}_s\, \boldsymbol{\nu}_k\,, \qquad \mathbf{M}_s = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{B.17}$$

Servo installation signs are then applied:

$$\mathbf{s}_k^{\mathrm{servo}} = \mathbf{D}_s\, \mathbf{s}_k^{\mathrm{phys}}, \qquad \mathbf{D}_s = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}. \tag{B.18}$$

Each normalised servo command is then encoded as a 16-bit packet value:

$$c_{i,k} = \mathrm{round}\left[\frac{s_{i,k}^{\mathrm{servo}} + 1}{2}\left(2^{16} - 1\right)\right], \qquad i = 1, \ldots, 4\,. \tag{B.19}$$

B.3.2 Bench Response Measurement

As illustrated in Fig. 3.6, the actuator timing test tracked temporary Vicon markers attached to the individual control surfaces while repeated step commands were sent through the selected COTS RC command path. For event i, let t_i be the command write time and let $s_i(t_j)$ be the filtered measured surface position at sample time t_j.

A pre-command window $W_{0,i}$ defines the initial surface position,

$$s_{0,i} = \frac{1}{|W_{0,i}|} \sum_{t_j \in W_{0,i}} s_i(t_j)\,, \tag{B.20}$$

and a final settled window $W_{1,i}$ defines the achieved surface position,

$$s_{1,i} = \frac{1}{|W_{1,i}|} \sum_{t_j \in W_{1,i}} s_i(t_j)\,. \tag{B.21}$$

The measured response is then normalised as

$$\eta_i(t_j) = \frac{s_i(t_j) - s_{0,i}}{s_{1,i} - s_{0,i}}. \tag{B.22}$$

This normalisation handles both positive and negative steps, because the denominator keeps the measured step direction. Events were rejected when the averaging windows were invalid, the achieved step was too small, or the crossing time could not be identified reliably.

For a response fraction $\gamma \in \{0.1, 0.5, 0.9\}$, the crossing time is defined as

$$t_{\gamma,i} = \inf\{t_j - t_i : t_j \geq t_i,\ \eta_i(t_j) \geq \gamma\}. \tag{B.23}$$

When the crossing lies between two samples, $\eta_i(t_j) < \gamma \leq \eta_i(t_{j+1})$, linear interpolation is used:

$$t_{\gamma,i} = t_j - t_i + \frac{\gamma - \eta_i(t_j)}{\eta_i(t_{j+1}) - \eta_i(t_j)}\,(t_{j+1} - t_j). \tag{B.24}$$

B.3.3 Control Surface State Model

The controller and command interface use normalised aggregate commands, but the flight dynamics model uses control surface deflections in radians. Therefore, the executable command $\boldsymbol{\nu}_k$ is first converted into an aggregate surface target:

$$\boldsymbol{\delta}_k^{\text{cmd}} = \mathcal{R}_\delta(\boldsymbol{\nu}_k) = \begin{bmatrix} \mathcal{R}_a(\nu_{a,k}) \\ \mathcal{R}_e(\nu_{e,k}) \\ \mathcal{R}_r(\nu_{r,k}) \end{bmatrix}, \tag{B.25}$$

where $\mathcal{R}_a$, $\mathcal{R}_e$, and $\mathcal{R}_r$ apply the measured aggregate surface limits: $[19.3^\circ, -21.5^\circ]$ for aileron, $[23.7^\circ, -32.0^\circ]$ for elevator, and $[33.0^\circ, -33.0^\circ]$ for rudder.

The measured surface motion is then modelled as two effects: an initial-response delay before repeatable movement begins, followed by a first-order lag toward the delayed surface target. The initial-response delay is taken as the median 10% crossing time, $\tilde{t}_{10}$. The delayed target is denoted by $\boldsymbol{\delta}_{k-d}^{\text{cmd}}$, where d is the corresponding delay in samples.

According to Fig. 3.5, Vicon system tracking configuration during the flight test does not provide direct control surface deflection measurements. Accordingly, for simulation and real flight state estimation, the actuator state is updated by

$$\boldsymbol{\delta}_{k+1} = \boldsymbol{\delta}_k + \left[1 - \exp\left(-\frac{\Delta t_k}{\tau_\delta}\right)\right]\left(\boldsymbol{\delta}_{k-d}^{\text{cmd}} - \boldsymbol{\delta}_k\right). \tag{B.26}$$

The time constant τ_δ is selected from the measured half-response time. For a first-order response,

$$0.5 = 1 - \exp\left[-\frac{t_{50} - \tilde{t}_{10}}{\tau_\delta}\right], \qquad \tau_\delta = \frac{t_{50} - \tilde{t}_{10}}{\ln 2}. \tag{B.27}$$

B.3.4 Timing Cases Passed to Controller Development

The controller timing model developed from the preceding measurements keeps two effects separate: Vicon-based state feedback delay and the actuator response model. The median case uses the median state feedback delay, the median initial-response time $\tilde{t}_{10}$, and the median actuator time constant $\tilde{\tau}_\delta$. The conservative case uses the conservative state feedback delay, the same measured initial-response time, and the conservative actuator time constant $\tau_\delta^{\text{cons}}$. This separation allows the controller tests in Chapter 6 to evaluate timing mismatch in state estimate and in control surface response separately.

The per-run response statistics used to form these timing cases are listed in Table B.5, where the n column counts Vicon frames for Vicon latency rows and accepted events for the surface response rows.

Table B.5 Per-run Vicon latency and command to control surface response timing.

Run	n	Metric	Unit	median	P_{95}	P_{99}	max	σ
1	20728	Vicon capture to report latency, $T_{V,\ell}$	ms	16.5	18.0	19.5	28.4	1.14
	160	Surface initial-response, t_{10}	ms	74.5	112.7	120.3	127.3	31.5
	160	Surface half-response, t_{50}	ms	111.2	132.9	140.2	150.5	15.1
	160	Surface near-final response, t_{90}	ms	131.1	166.7	242.9	278.9	25.0
2	27464	Vicon capture to report latency, $T_{V,\ell}$	ms	14.8	15.3	15.6	17.1	0.47
	160	Surface initial-response, t_{10}	ms	72.6	97.1	111.2	131.9	28.3
	160	Surface half-response, t_{50}	ms	105.6	123.4	133.1	149.7	13.5
	160	Surface near-final response, t_{90}	ms	129.2	170.6	270.2	274.5	29.2
3	23860	Vicon capture to report latency, $T_{V,\ell}$	ms	14.8	15.3	15.7	17.1	0.53
	160	Surface initial-response, t_{10}	ms	71.3	96.6	111.5	114.3	27.4
	160	Surface half-response, t_{50}	ms	104.2	126.0	139.1	149.1	14.0
	160	Surface near-final response, t_{90}	ms	127.5	161.2	283.7	318.5	28.9
4	25500	Vicon capture to report latency, $T_{V,\ell}$	ms	14.8	15.3	15.7	31.8	0.52
	160	Surface initial-response, t_{10}	ms	72.4	106.9	116.7	137.0	35.7
	160	Surface half-response, t_{50}	ms	109.2	128.3	140.8	147.3	18.6
	160	Surface near-final response, t_{90}	ms	131.5	151.7	166.1	175.3	15.9
5	25364	Vicon capture to report latency, $T_{V,\ell}$	ms	14.8	15.3	15.7	36.9	0.59
	160	Surface initial-response, t_{10}	ms	74.6	112.2	127.2	136.2	35.9
	160	Surface half-response, t_{50}	ms	109.0	134.6	141.7	150.8	19.1
	160	Surface near-final response, t_{90}	ms	130.1	152.6	165.9	174.0	16.6

The full response-extraction workflow is summarised in Algorithm B.4.

Algorithm B.4 Command to control surface response analysis

Input: Command schedule, selected RC command path, Vicon surface marker stream, surface filter, and crossing fractions $\Gamma = \{0.1, 0.5, 0.9\}$.

Output: Accepted response events, crossing-time summary, and rejection audit.

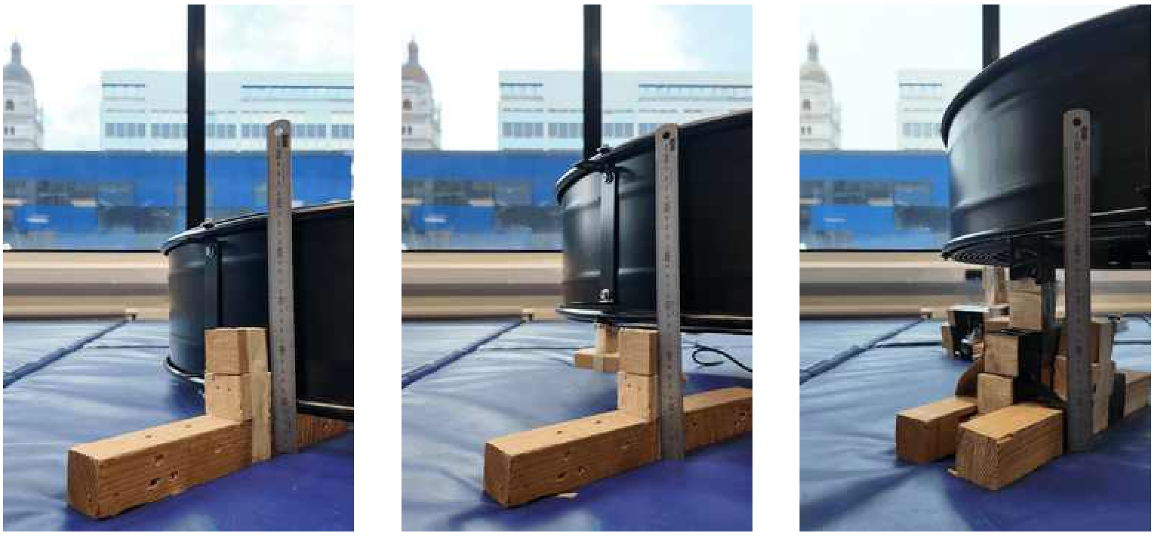
```
Initialise the command path, set surfaces to neutral, and start the Vicon stream
for each scheduled command event do
    Quantise and map the aggregate command using (B.16)–(B.19)
    Send the packet, record the command-write time t_i, and log the filtered surface measurement s_i(t_j)
for each recorded event do
    Compute s_{0,i}, s_{1,i}, and η_i(t_j) using (B.20)–(B.22)
    if the event has insufficient amplitude or ambiguous crossing then
        reject the event and record the reason
    else
        Compute t_{10,i}, t_{50,i}, and t_{90,i} using (B.23)–(B.24)
        add the event to the accepted table
Aggregate accepted events into median crossing times and the conservative half-response bound
return accepted events, timing summary, and rejection audit
```

C Experimental Characterisation of the Fan-Generated Updraft

C.1 Test Facility and Measurement Configurations

(a) with a single axial drum fan. (b) with four axial drum fans.

Fig. C.1. Experimental layout for indoor updraft measurements.

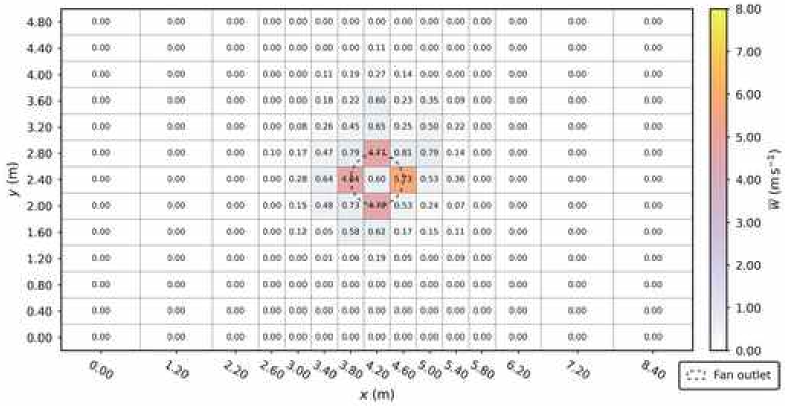


(a) outlet height: 0.245 m, support offset: 0.040 m. (b) outlet height: 0.330 m, support offset: 0.125 m. (c) outlet height: 0.425 m, support offset: 0.220 m.

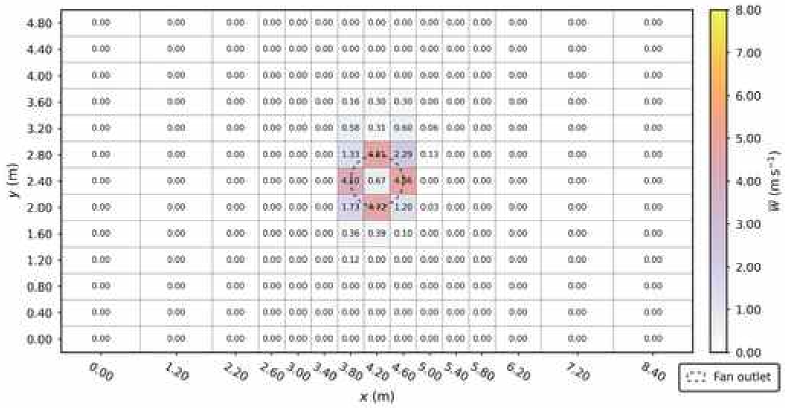
Fig. C.2. Single fan configuration with varying outlet heights for ground-effect assessment.

C.2 Spatial Field Measurements

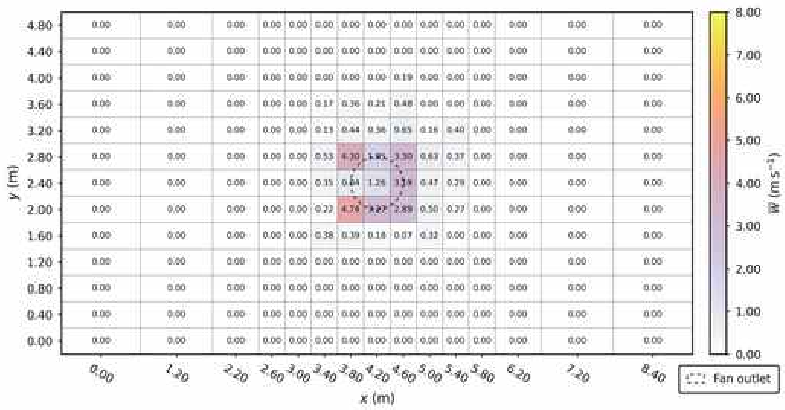
C.2.1 Single-Fan Case



(a) 0.20 m above fan outlet plane.

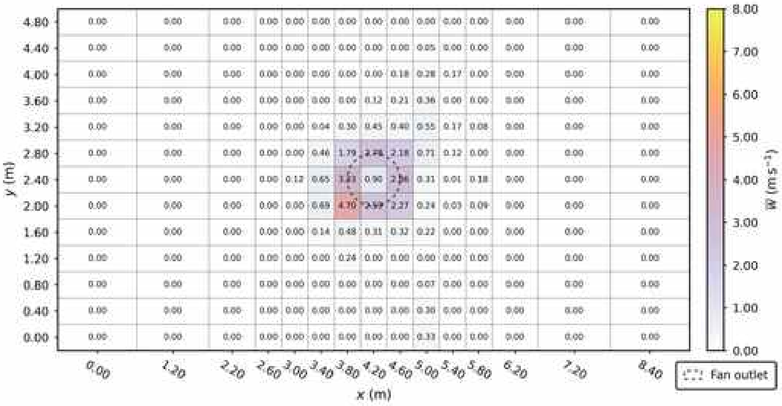


(b) 0.35 m above fan outlet plane.

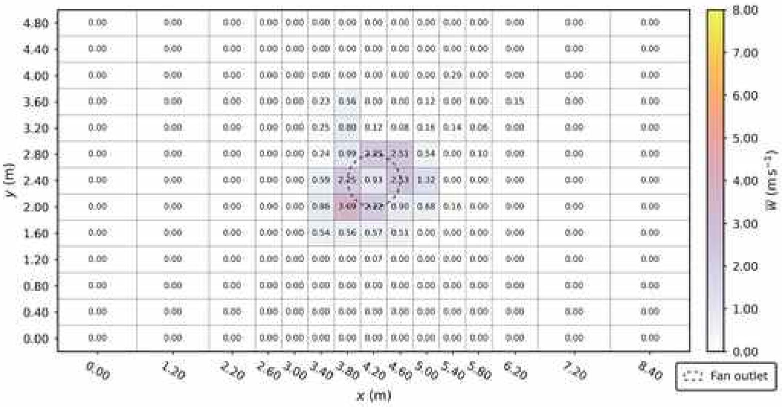


(c) 0.50 m above fan outlet plane.

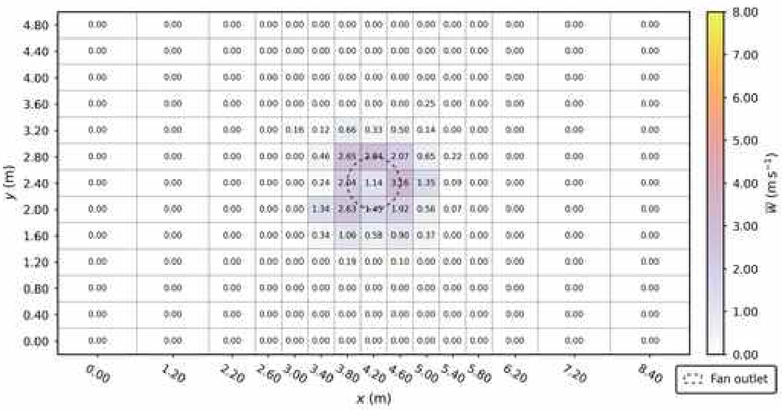
Fig. C.3. Spatial distribution of mean vertical velocity at lower measurement heights (single fan).



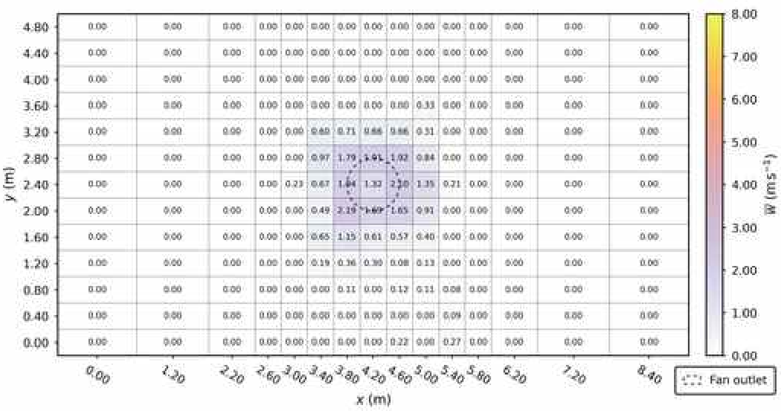
(a) 0.75 m above fan outlet plane.



(b) 1.10 m above fan outlet plane.



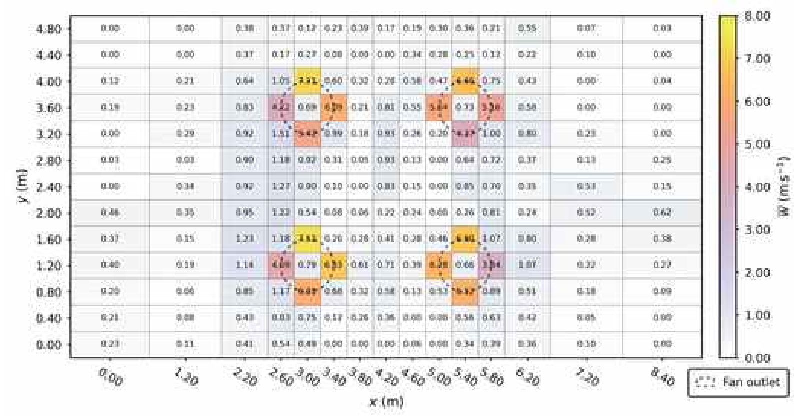
(c) 1.60 m above fan outlet plane.



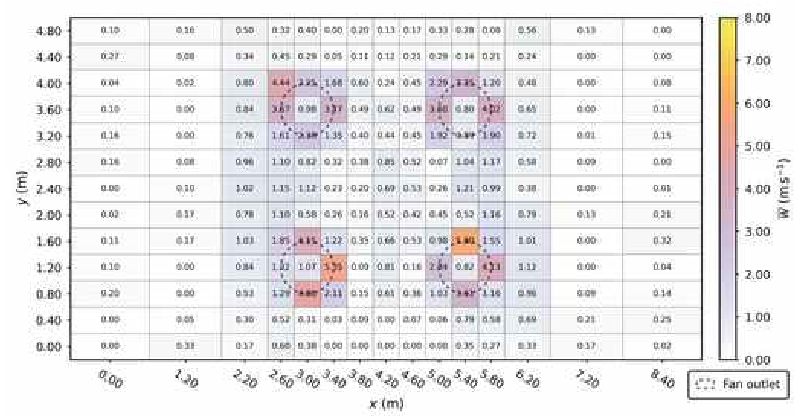
(d) 2.20 m above fan outlet plane.

Fig. C.4. Spatial distribution of mean vertical velocity at higher measurement heights (single fan).

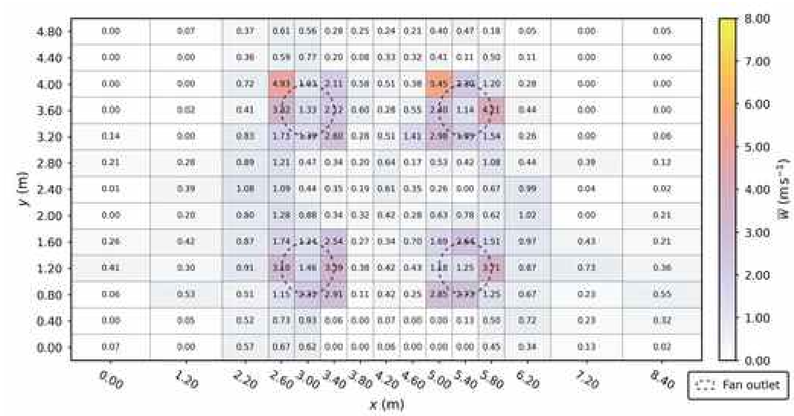
C.2.2 Four-Fan Case



(a) 0.20 m above fan outlet plane.

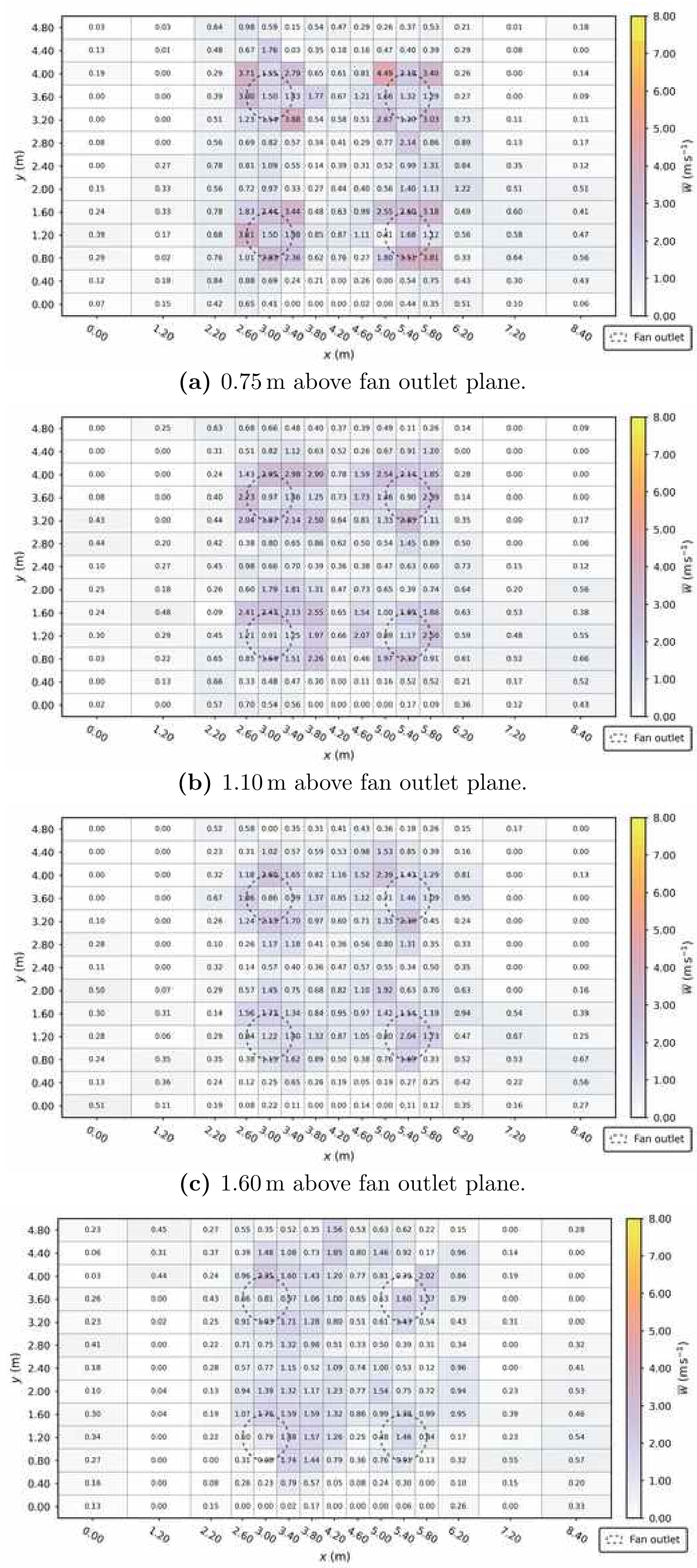


(b) 0.35 m above fan outlet plane.



(c) 0.50 m above fan outlet plane.

Fig. C.5. Spatial distribution of mean vertical velocity at lower measurement heights (four fans).



(a) 0.75 m above fan outlet plane.

(b) 1.10 m above fan outlet plane.

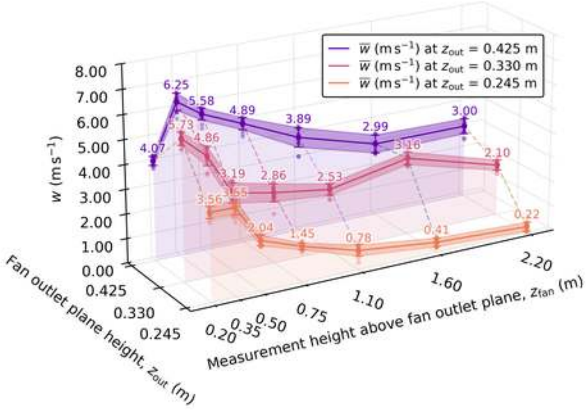
(c) 1.60 m above fan outlet plane.

(d) 2.20 m above fan outlet plane.

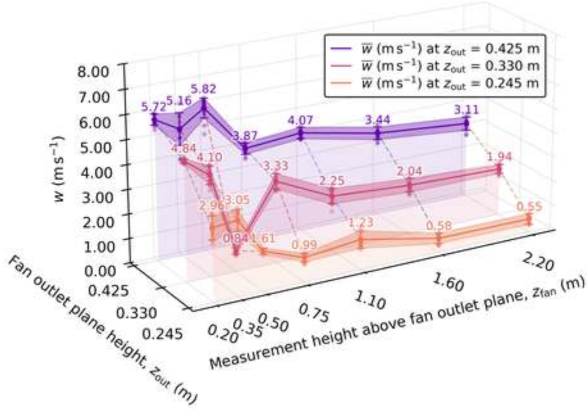
Fig. C.6. Spatial distribution of mean vertical velocity at higher measurement heights (four fans).

C.3 Supplementary Measurements at Sampling Points

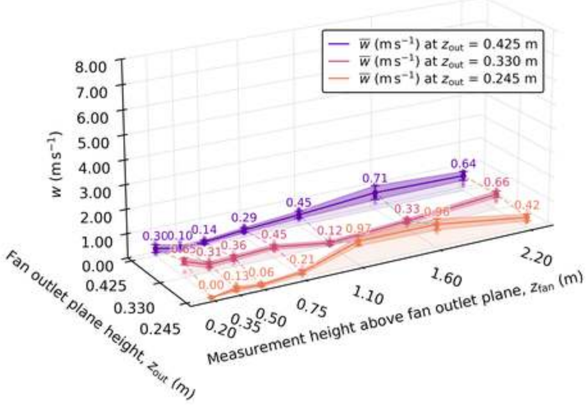
C.3.1 Single-Fan Case



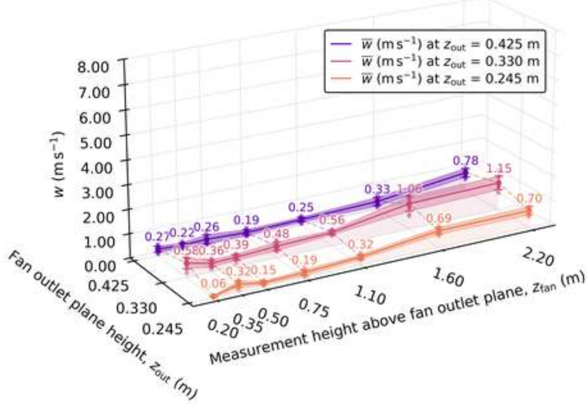
(a) Point 1 (near-core) at $(x, y) = (4.60, 2.40)$m.



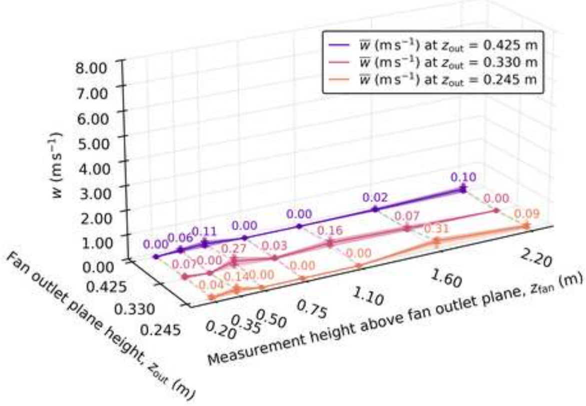
(b) Point 2 (near-core) at $(x, y) = (3.80, 2.40)$m.



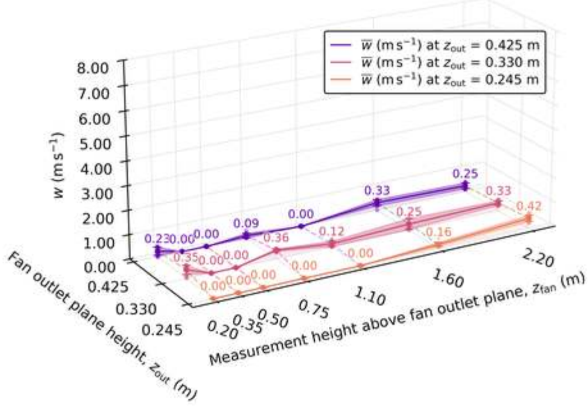
(c) Point 3 (off-core) at $(x, y) = (4.20, 3.20)$ m.



(d) Point 4 (off-core) at $(x, y) = (3.80, 1.60)$ m.

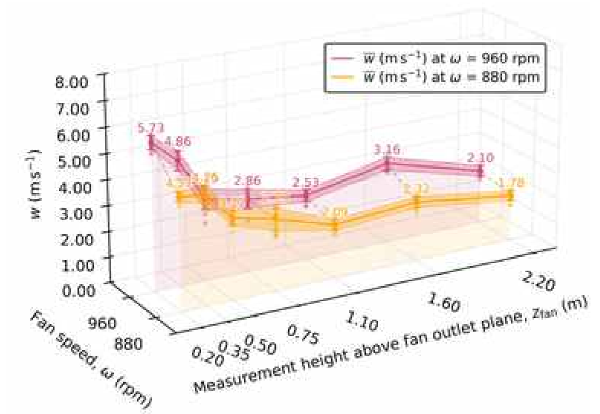


(e) Point 5 (far-field) at $(x, y) = (5.40, 2.00)$ m.

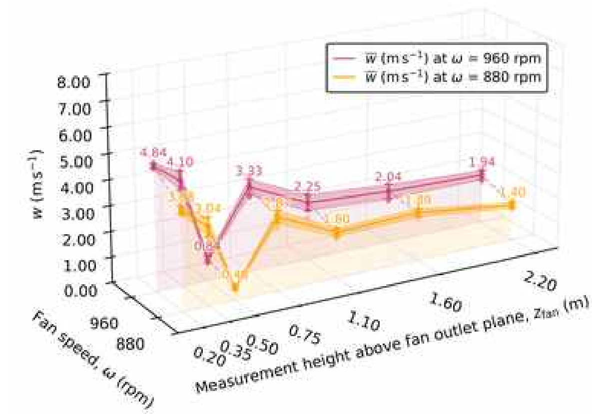


(f) Point 6 (far-field) at $(x, y) = (5.00, 3.60)$ m.

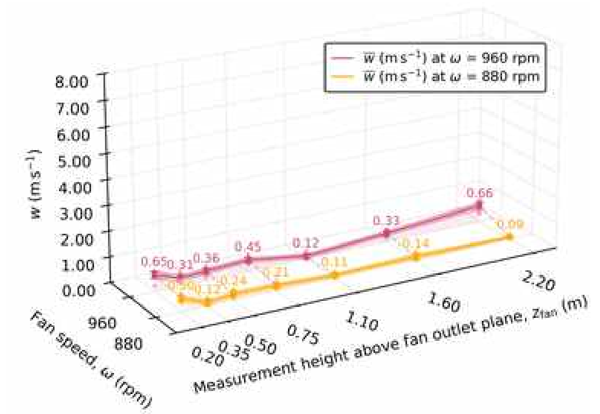
Fig. C.7. Vertical velocity profiles at six sampling points for three fan outlet plane heights (single fan). The solid line shows the mean, semi-transparent markers show the 15 samples (1 Hz over 15 s) at each measurement height, and shaded bands show $\pm 1\sigma$.



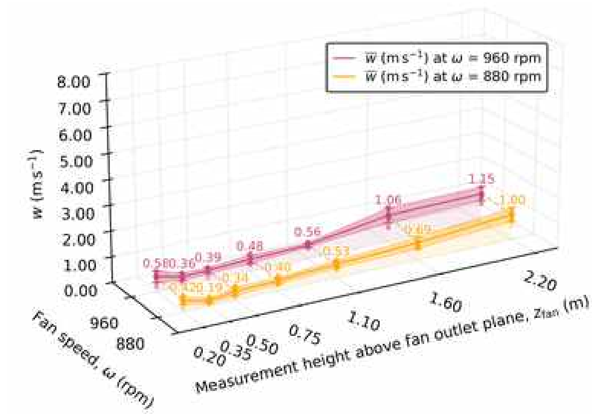
(a) Point 1 (near-core) at $(x,\ y) = (4.60,\ 2.40)$ m.



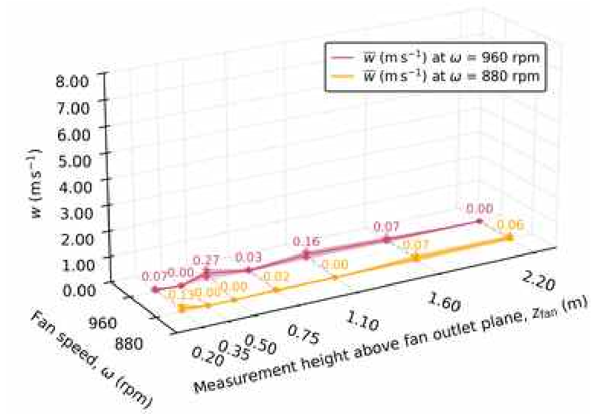
(b) Point 2 (near-core) at $(x,\ y) = (3.80,\ 2.40)$ m.



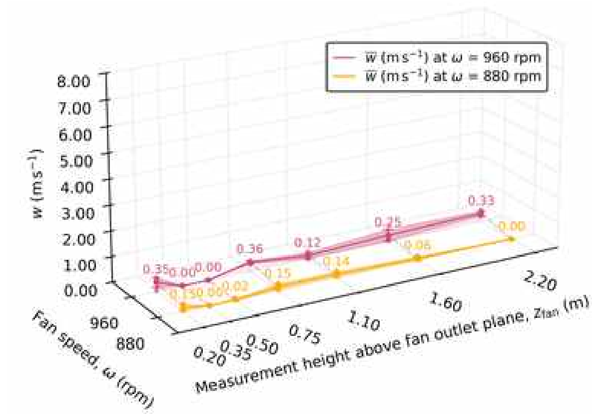
(c) Point 3 (off-core) at $(x,\ y) = (4.20,\ 3.20)$ m.



(d) Point 4 (off-core) at $(x,\ y) = (3.80,\ 1.60)$ m.



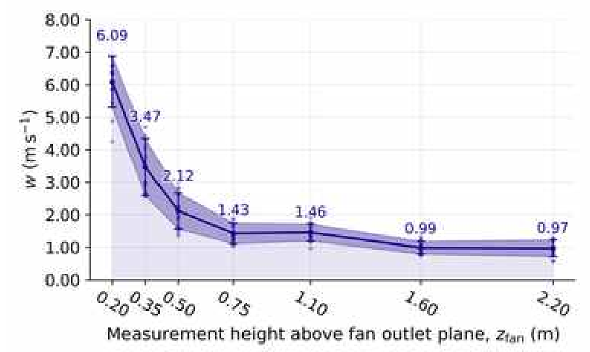
(e) Point 5 (far-field) at $(x,\ y) = (5.40,\ 2.00)$ m.



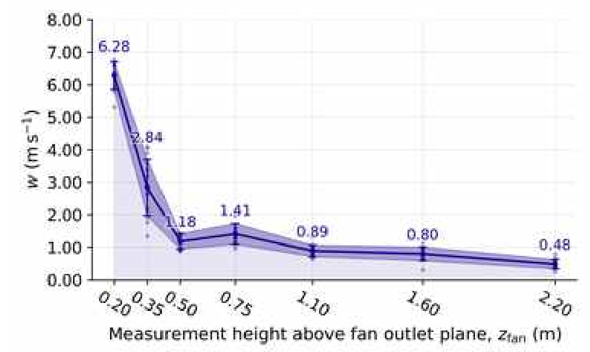
(f) Point 6 (far-field) at $(x,\ y) = (5.00,\ 3.60)$ m.

Fig. C.8. Vertical velocity profiles at six sampling points for two fan speed settings (single fan). The solid line shows the mean, semi-transparent markers show the 15 samples (1 Hz over 15 s) at each measurement height, and shaded bands show $\pm 1\sigma$.

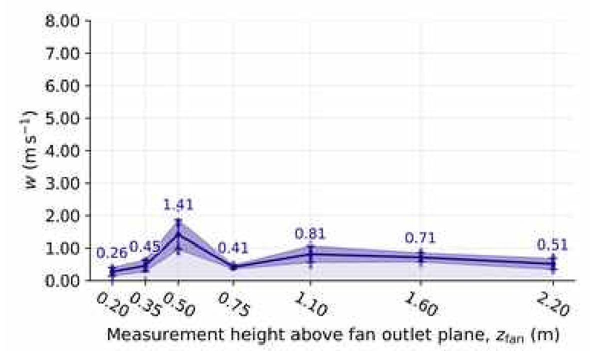
C.3.2 Four-Fan Case



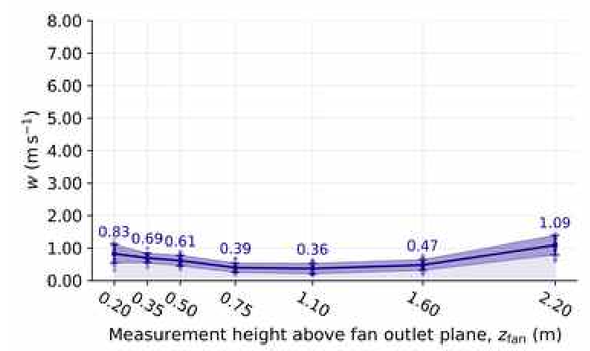
(a) Point 1 (near-core) at $(x, y) = (3.40, 3.60)$ m.



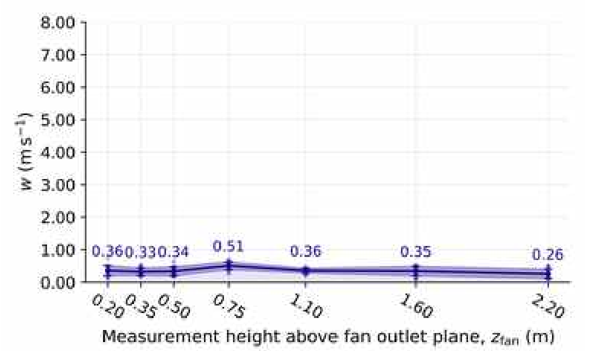
(b) Point 2 (near-core) at $(x, y) = (5.00, 1.20)$ m.



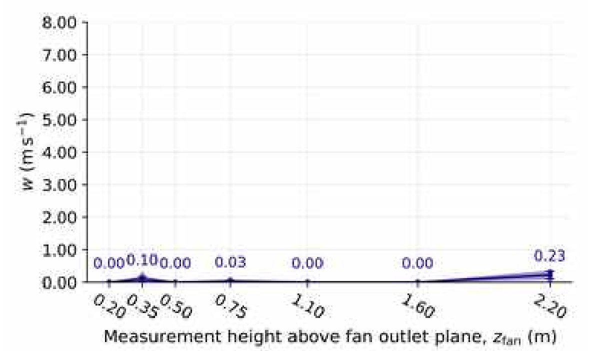
(c) Point 3 (interaction) at $(x, y) = (4.60, 3.20)$ m.



(d) Point 4 (interaction) at $(x, y) = (4.20, 2.40)$ m.



(e) Point 5 (far-field) at $(x, y) = (6.20, 0.00)$ m.



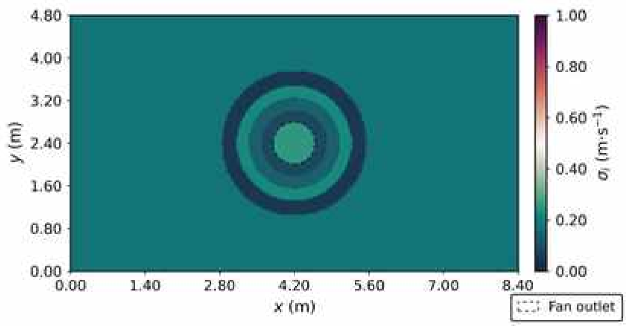
(f) Point 6 (far-field) at $(x, y) = (0.00, 4.80)$ m.

Fig. C.9. Vertical velocity profiles at six sampling points (four fans). The solid line shows the mean, semi-transparent markers show the 15 samples (1 Hz over 15 s) at each measurement height, and shaded bands show $\pm 1\sigma$.

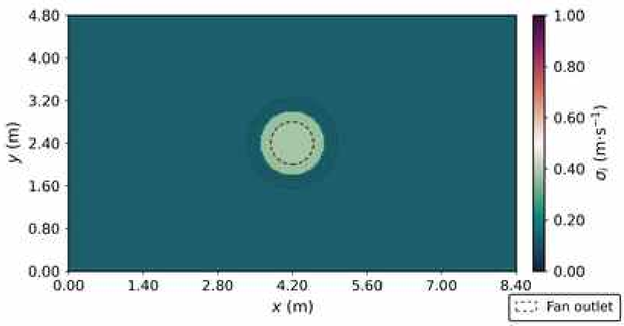
D Supplementary Updraft Surrogate Fits at Measurement Planes

D.1 Empirical Fluctuation Fields Used for Variance Assignment

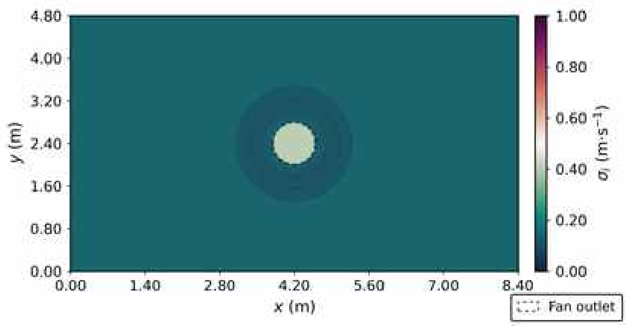
D.1.1 Single-Fan Empirical Fluctuation Field



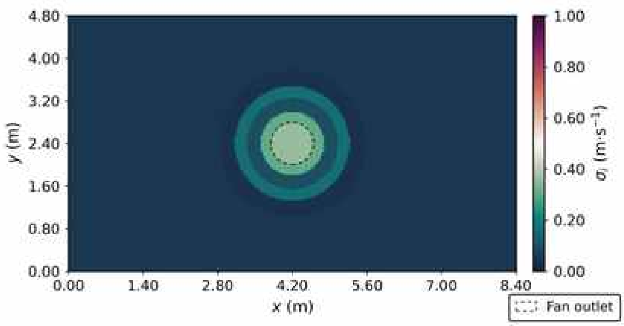
(a) 0.20 m above fan outlet plane.



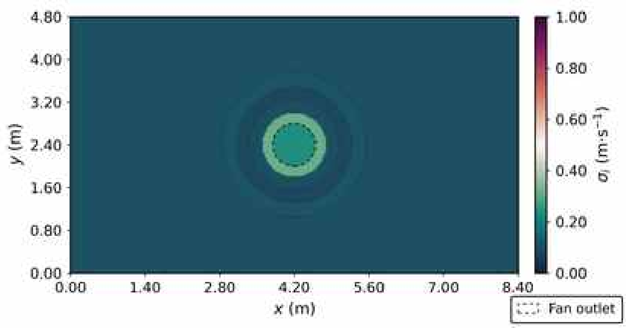
(b) 0.35 m above fan outlet plane.



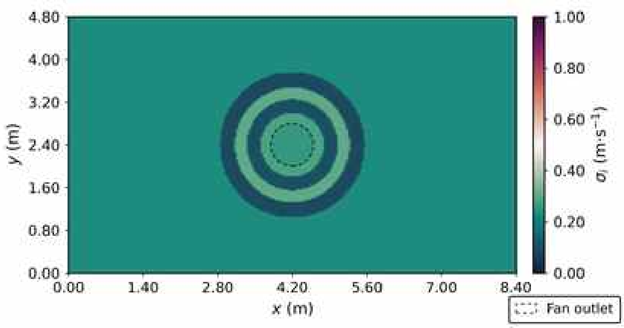
(c) 0.50 m above fan outlet plane.



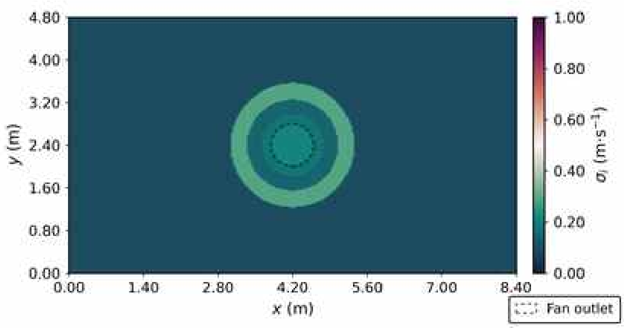
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



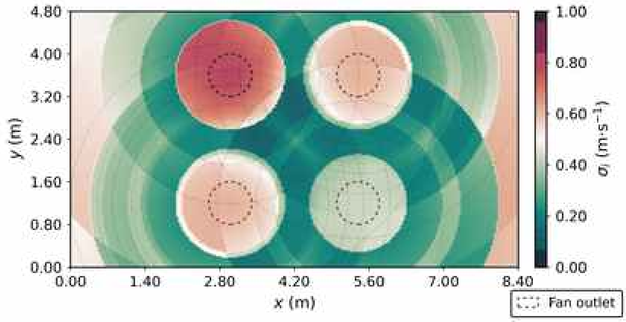
(f) 1.60 m above fan outlet plane.



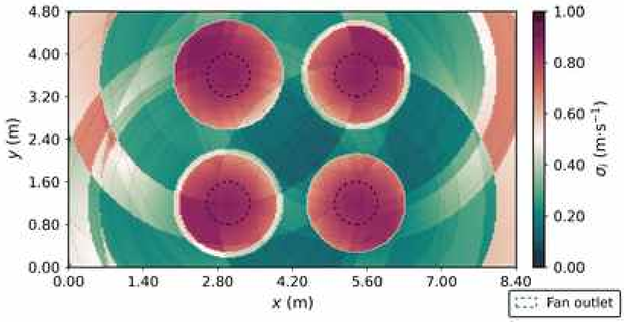
(g) 2.20 m above fan outlet plane.

Fig. D.1. Supplementary empirical fluctuation field, σ_i (single fan). The grey dotted rings indicate the annular regions used to assign the empirical fluctuation level, σ_i, as described at the beginning of Section 4.3.

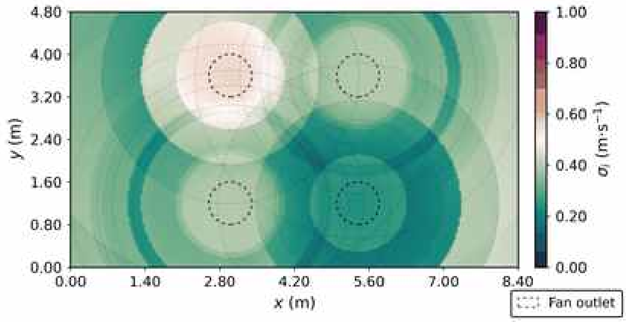
D.1.2 Four-Fan Empirical Fluctuation Field



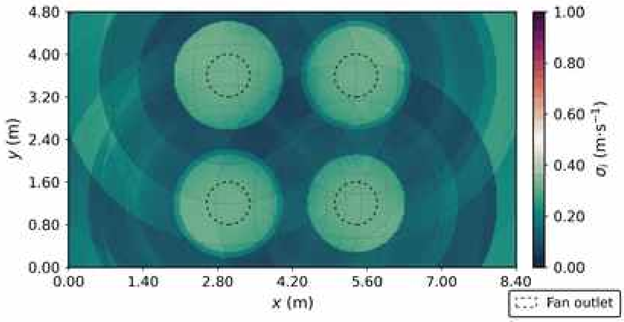
(a) 0.20 m above fan outlet plane.



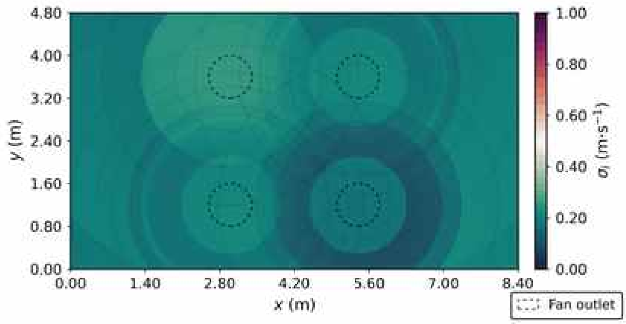
(b) 0.35 m above fan outlet plane.



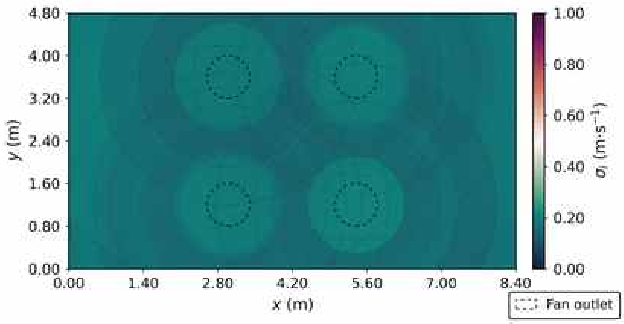
(c) 0.50 m above fan outlet plane.



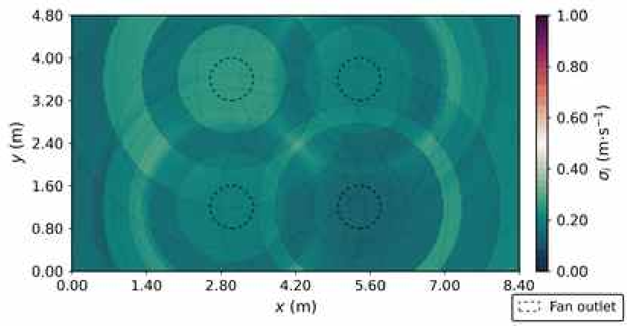
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



(f) 1.60 m above fan outlet plane.

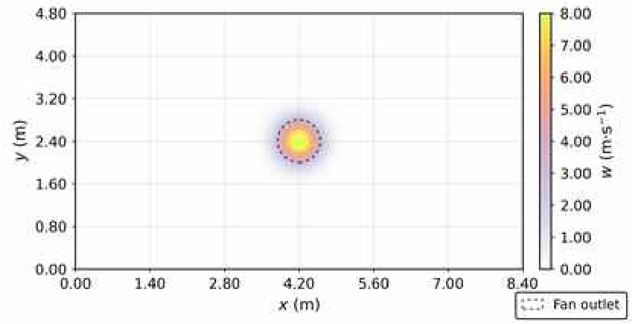


(g) 2.20 m above fan outlet plane.

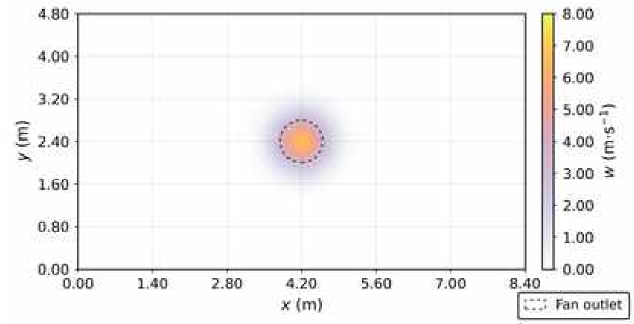
Fig. D.2. Supplementary empirical fluctuation field, σ_i (four fans). The grey dotted rings indicate the annular regions used to assign the empirical fluctuation level, σ_i, as described at the beginning of Section 4.3.

D.2 Axisymmetric Gaussian Plume Baseline Fits

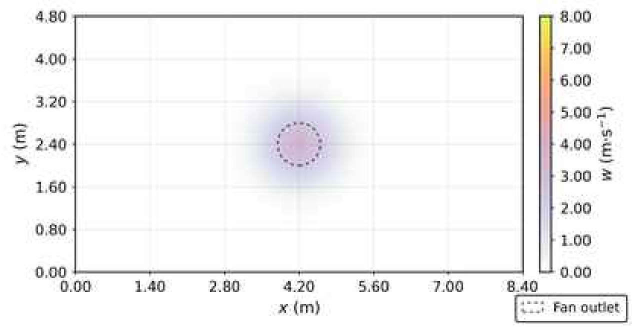
D.2.1 Single-Fan Axisymmetric Gaussian Plume Baseline



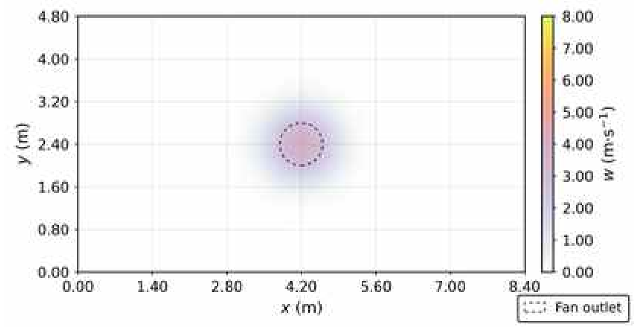
(a) 0.20 m above fan outlet plane.



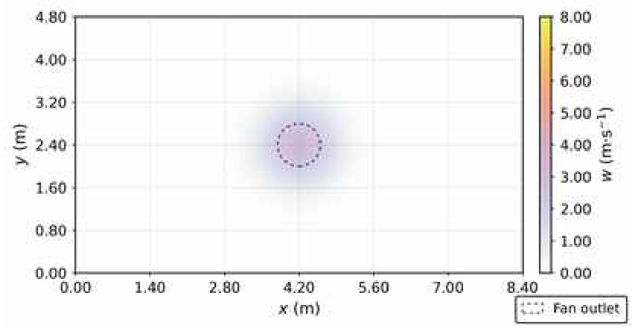
(b) 0.35 m above fan outlet plane.



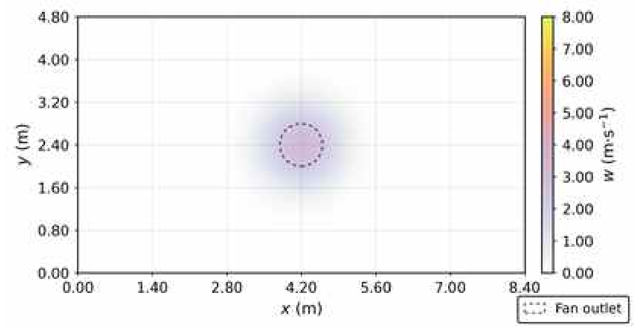
(c) 0.50 m above fan outlet plane.



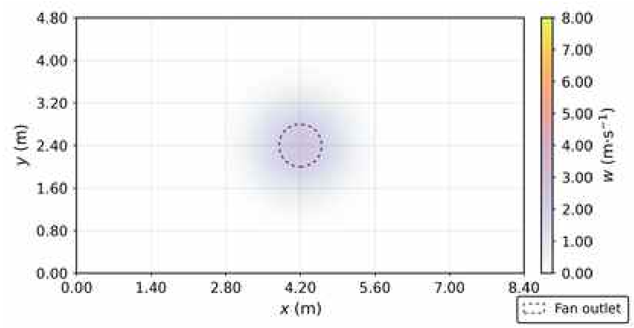
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



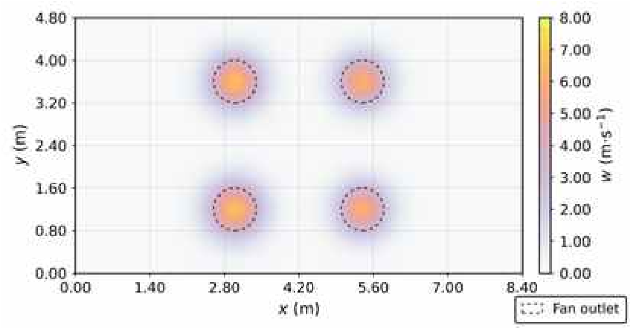
(f) 1.60 m above fan outlet plane.



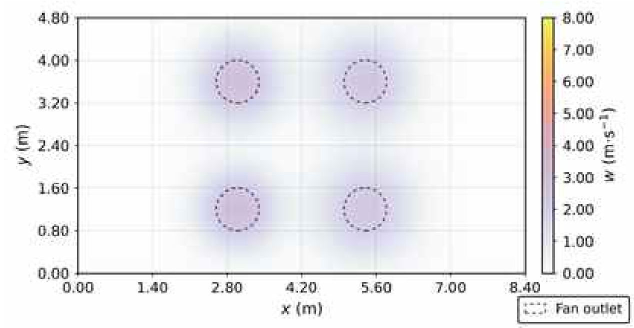
(g) 2.20 m above fan outlet plane.

Fig. D.3. Supplementary spatial distribution of fitted mean vertical velocity from the Gaussian plume baseline model (single fan).

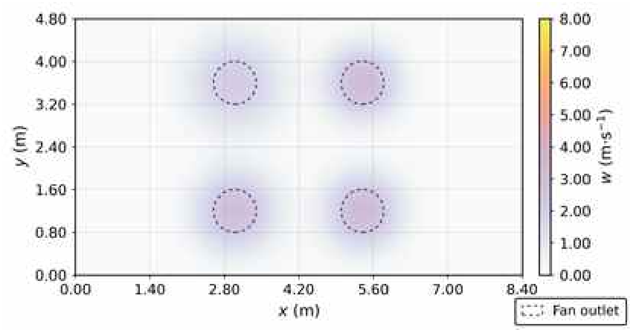
D.2.2 Four-Fan Axisymmetric Gaussian Plume Baseline



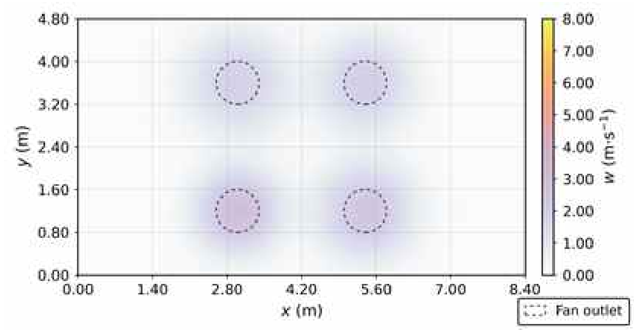
(a) 0.20 m above fan outlet plane.



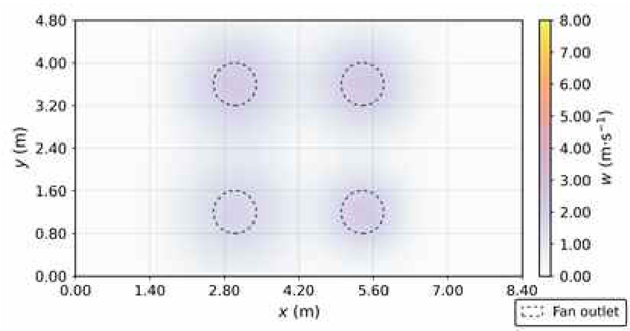
(b) 0.35 m above fan outlet plane.



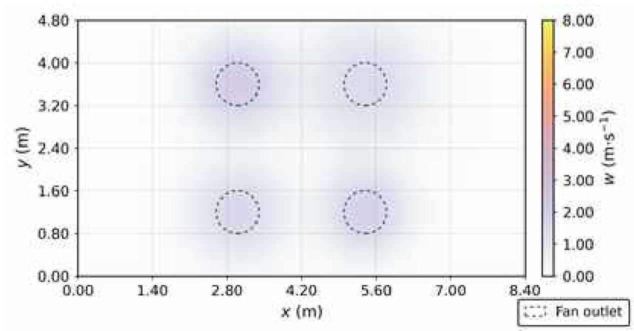
(c) 0.50 m above fan outlet plane.



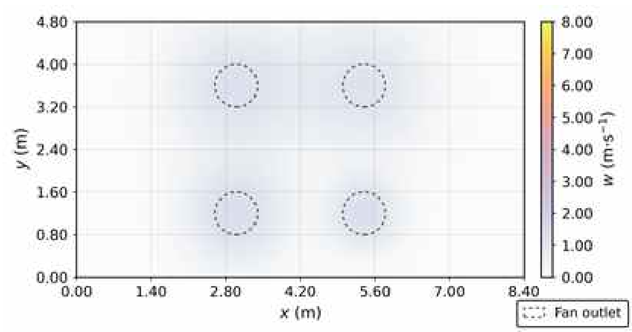
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



(f) 1.60 m above fan outlet plane.

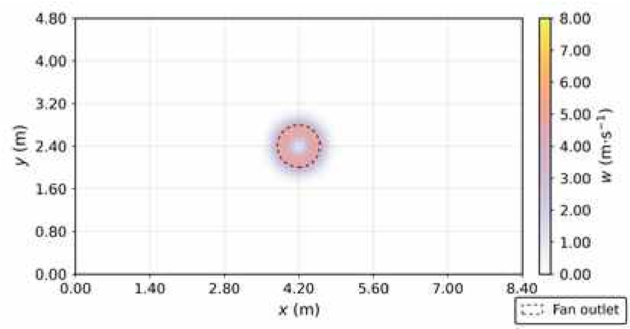


(g) 2.20 m above fan outlet plane.

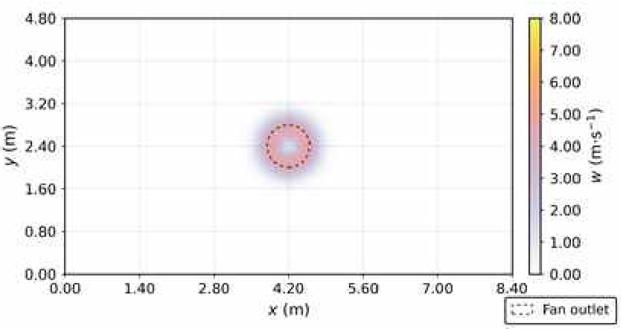
Fig. D.4. Supplementary spatial distribution of fitted mean vertical velocity from the Gaussian plume baseline model (four fans).

D.3 Axisymmetric Annular Gaussian Surrogate Fits

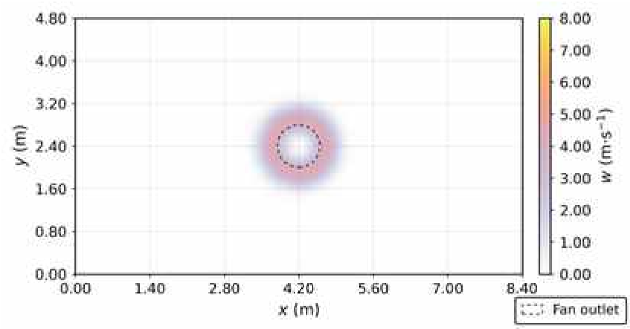
D.3.1 Single-Fan Axisymmetric Annular Gaussian Surrogate



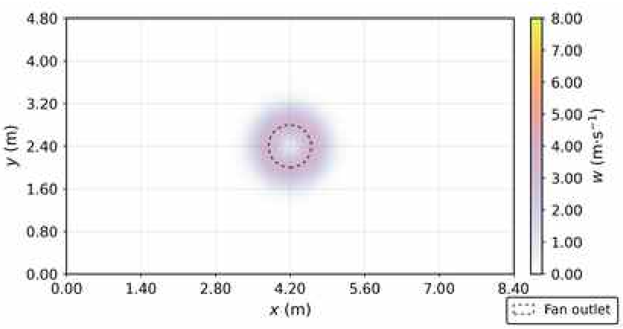
(a) 0.20 m above fan outlet plane.



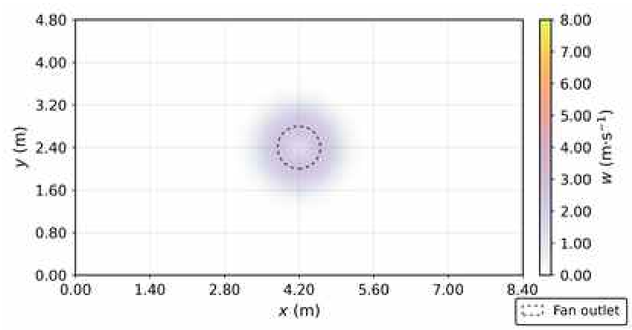
(b) 0.35 m above fan outlet plane.



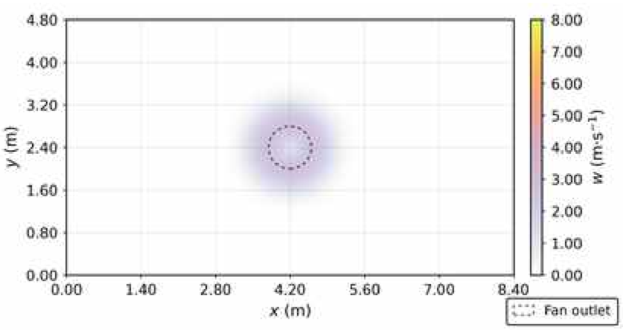
(c) 0.50 m above fan outlet plane.



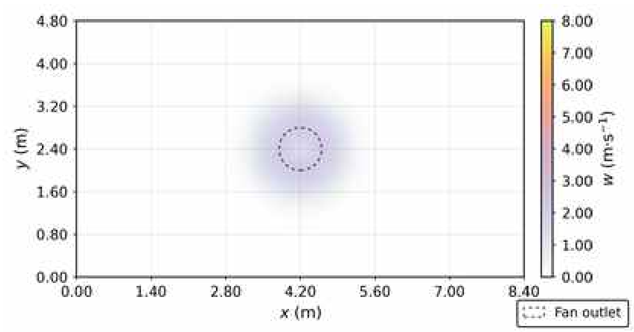
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



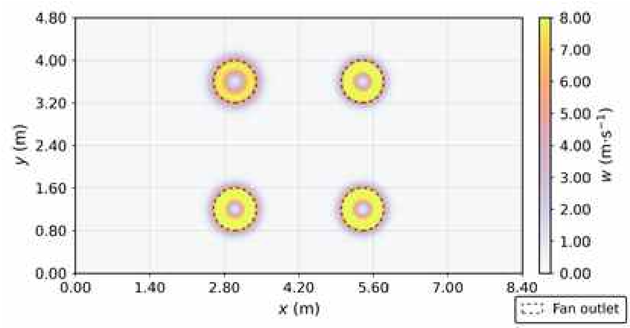
(f) 1.60 m above fan outlet plane.



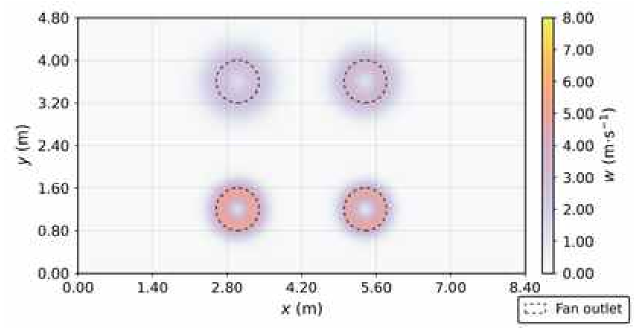
(g) 2.20 m above fan outlet plane.

Fig. D.5. Supplementary spatial distribution of fitted mean vertical velocity from the axisymmetric annular Gaussian surrogate (single fan).

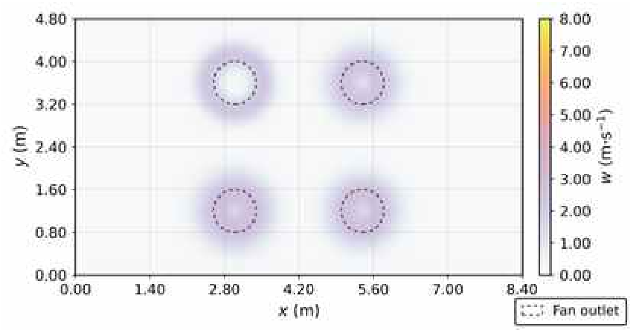
D.3.2 Four-Fan Axisymmetric Annular Gaussian Surrogate



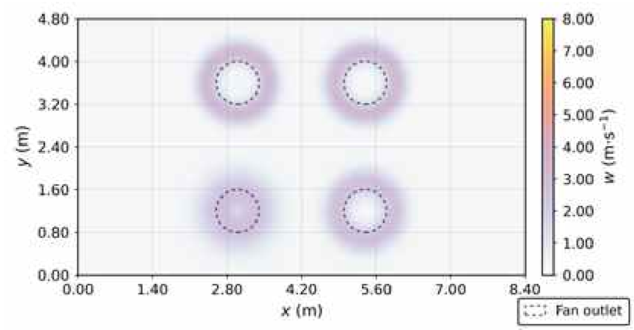
(a) 0.20 m above fan outlet plane.



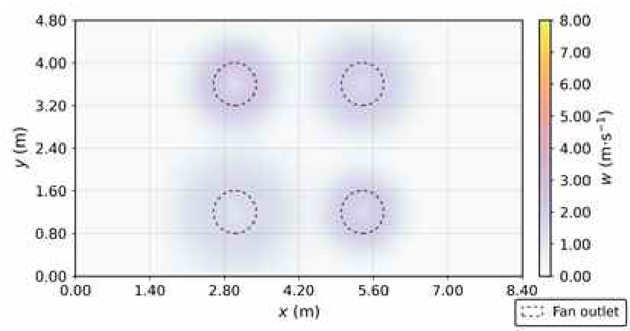
(b) 0.35 m above fan outlet plane.



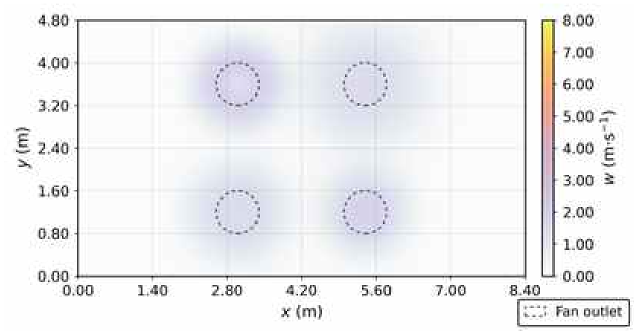
(c) 0.50 m above fan outlet plane.



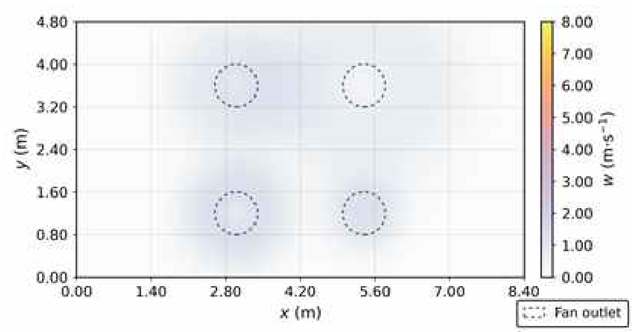
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



(f) 1.60 m above fan outlet plane.

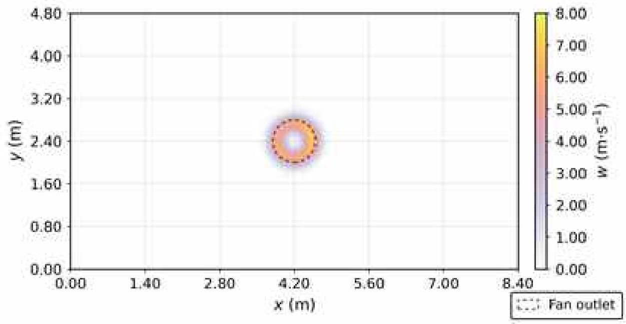


(g) 2.20 m above fan outlet plane.

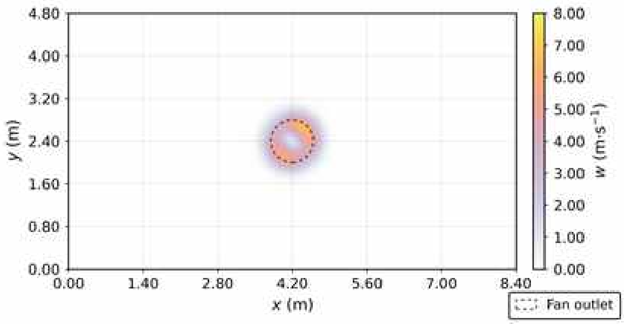
Fig. D.6. Supplementary spatial distribution of fitted mean vertical velocity from the axisymmetric annular Gaussian surrogate (four fans).

D.4 Harmonic Annular Gaussian Surrogate Fits

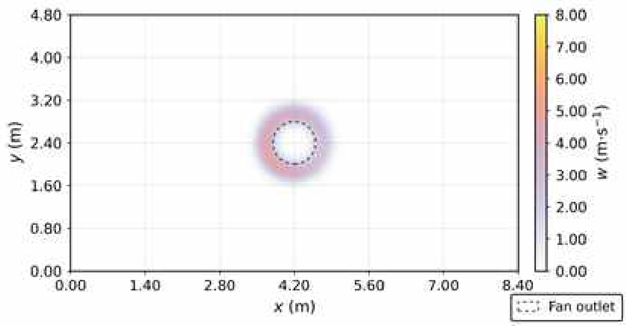
D.4.1 Single-Fan Harmonic Annular Gaussian Surrogate



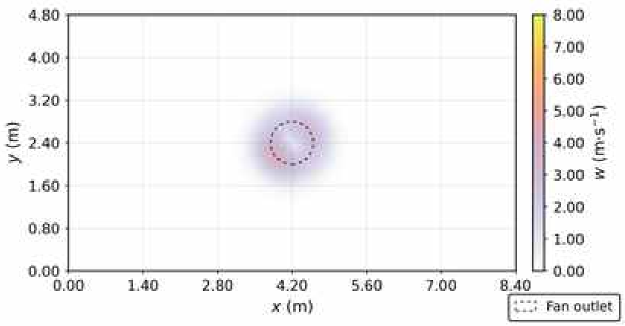
(a) 0.20 m above fan outlet plane.



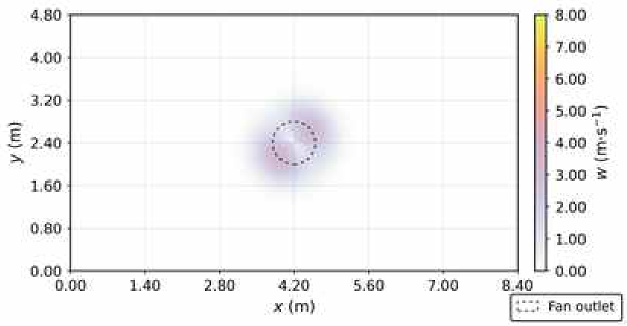
(b) 0.35 m above fan outlet plane.



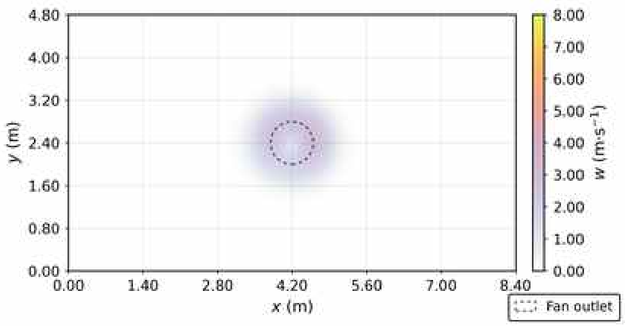
(c) 0.50 m above fan outlet plane.



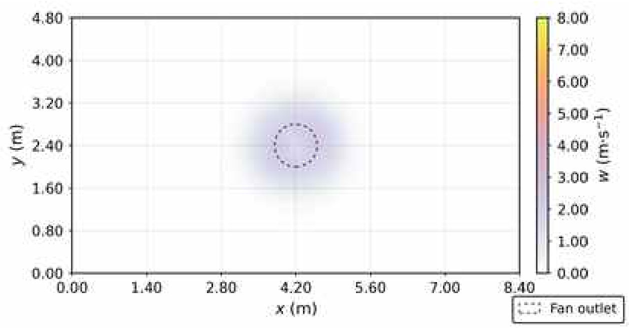
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



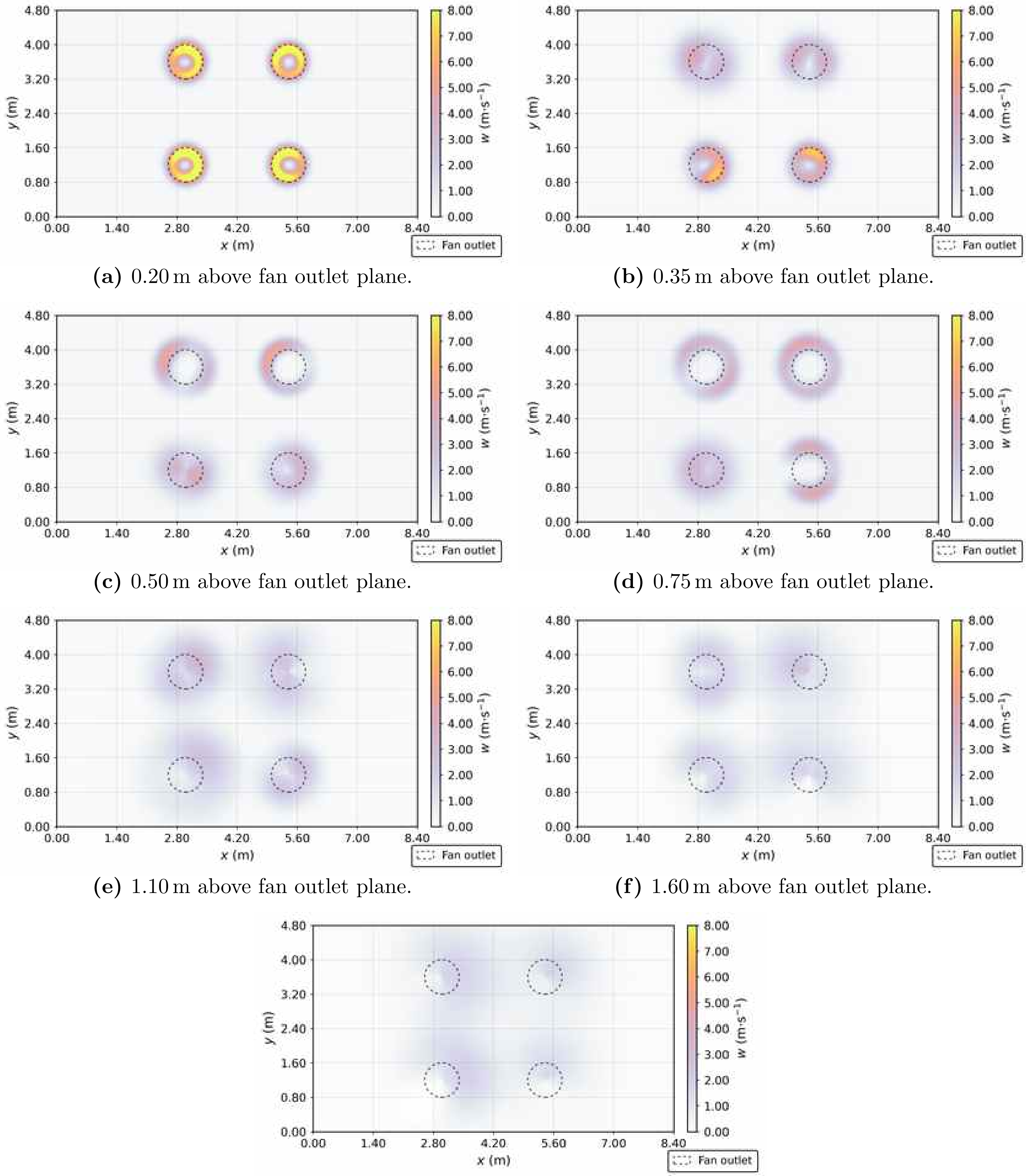
(f) 1.60 m above fan outlet plane.



(g) 2.20 m above fan outlet plane.

Fig. D.7. Supplementary spatial distribution of fitted mean vertical velocity from the harmonic annular Gaussian surrogate (single fan).

D.4.2 Four-Fan harmonic annular Gaussian Surrogate



(a) 0.20 m above fan outlet plane.

(b) 0.35 m above fan outlet plane.

(c) 0.50 m above fan outlet plane.

(d) 0.75 m above fan outlet plane.

(e) 1.10 m above fan outlet plane.

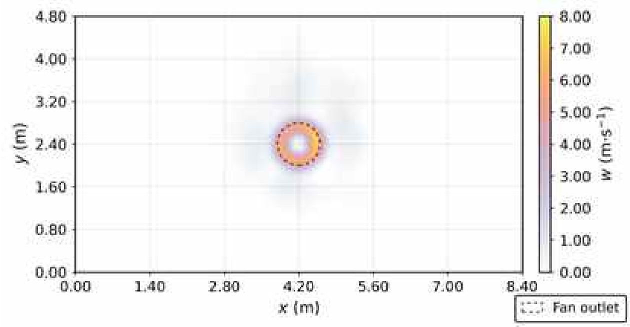
(f) 1.60 m above fan outlet plane.

(g) 2.20 m above fan outlet plane.

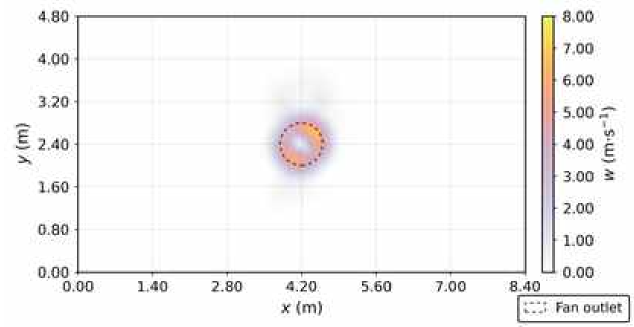
Fig. D.8. Supplementary spatial distribution of fitted mean vertical velocity from the harmonic annular Gaussian surrogate (four fans).

D.5 Residual Annular Gaussian Process Surrogate Predictions

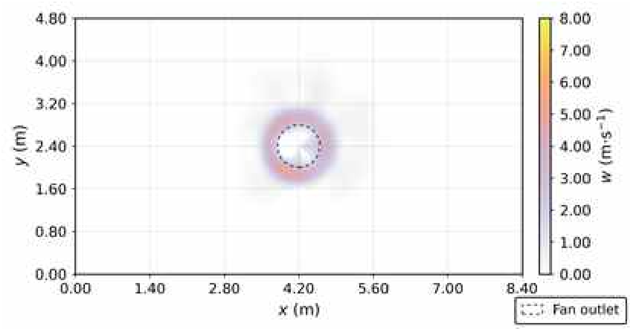
D.5.1 Single-Fan Residual Annular Gaussian Process Surrogate



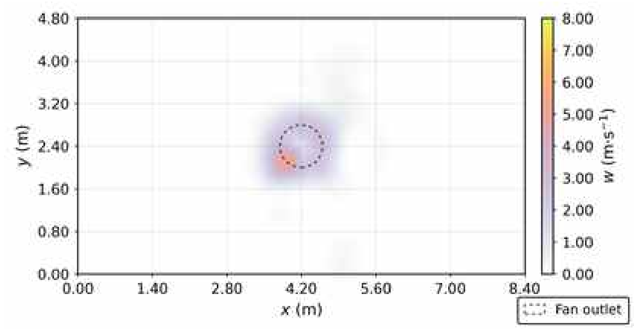
(a) 0.20 m above fan outlet plane.



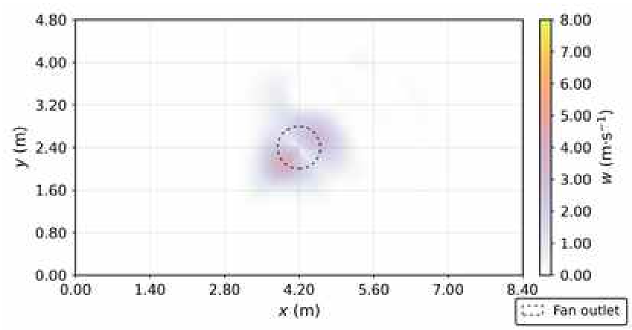
(b) 0.50 m above fan outlet plane.



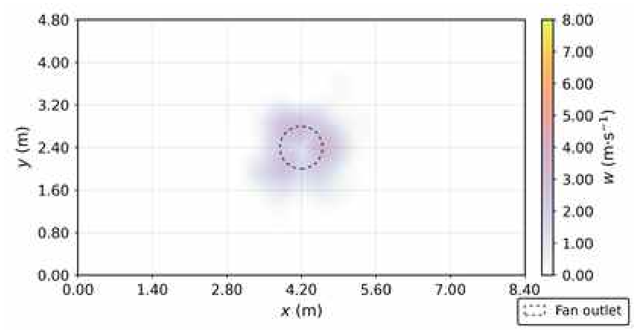
(c) 0.50 m above fan outlet plane.



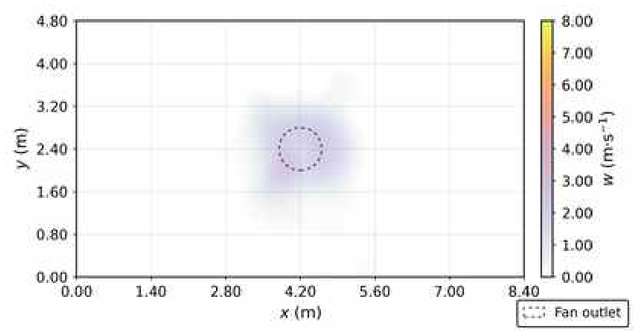
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.

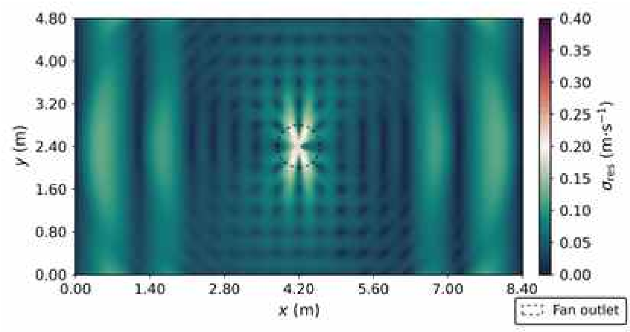


(f) 1.60 m above fan outlet plane.

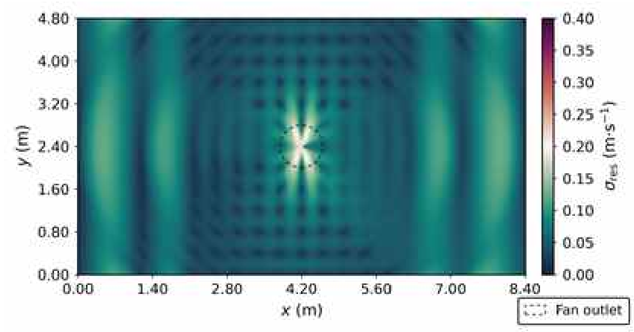


(g) 2.20 m above fan outlet plane.

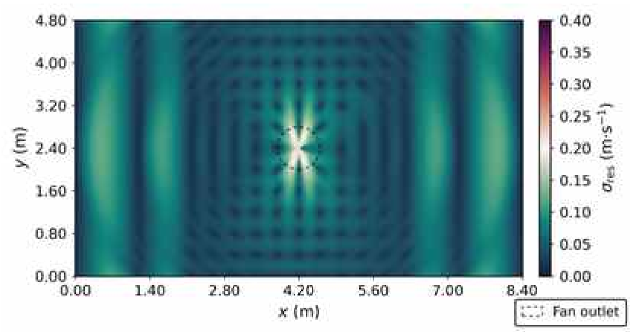
Fig. D.9. Supplementary spatial distribution of fitted mean vertical velocity from the residual annular Gaussian process surrogate (single fan).



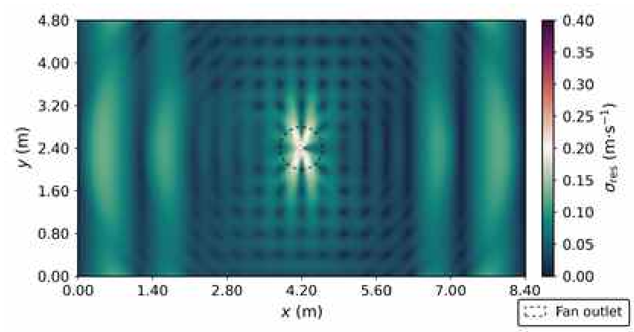
(a) 0.20 m above fan outlet plane.



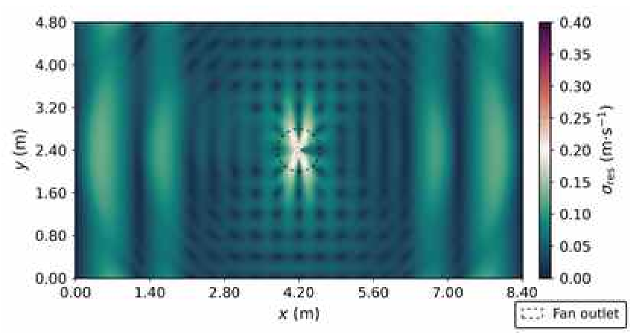
(b) 0.35 m above fan outlet plane.



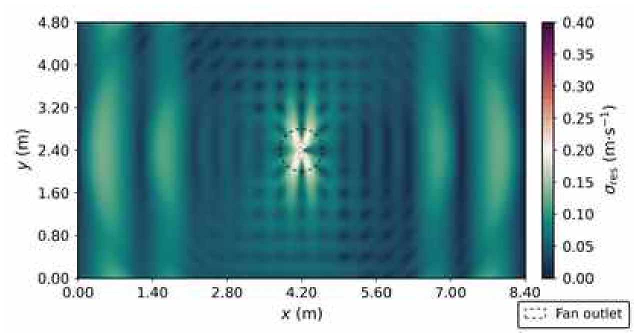
(c) 0.50 m above fan outlet plane.



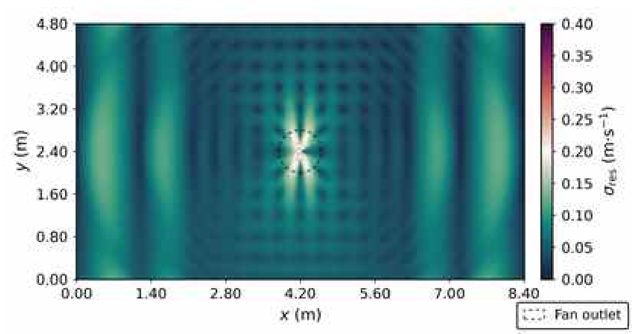
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.

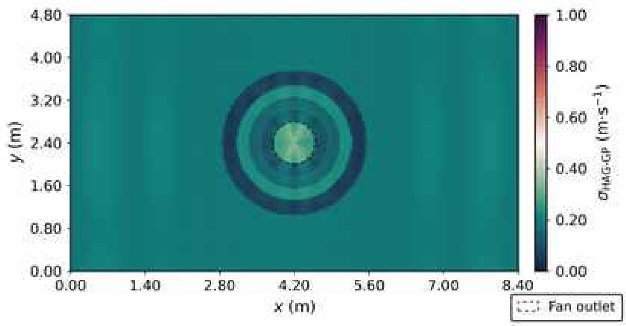


(f) 1.60 m above fan outlet plane.

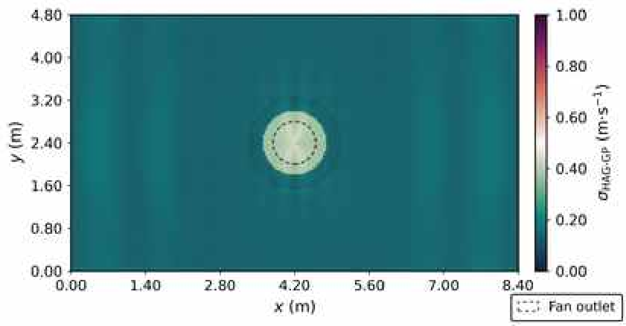


(g) 2.20 m above fan outlet plane.

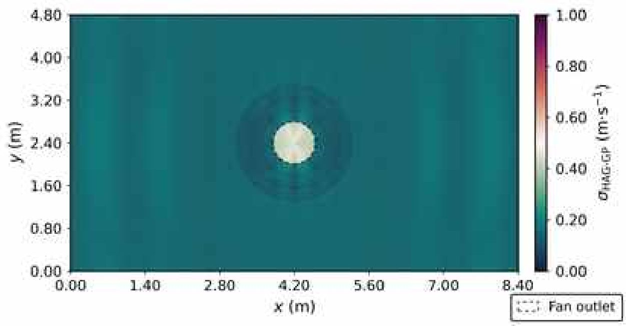
Fig. D.10. Supplementary predictive standard deviation field, σ_{res}, of the residual annular Gaussian process surrogate (single fan).



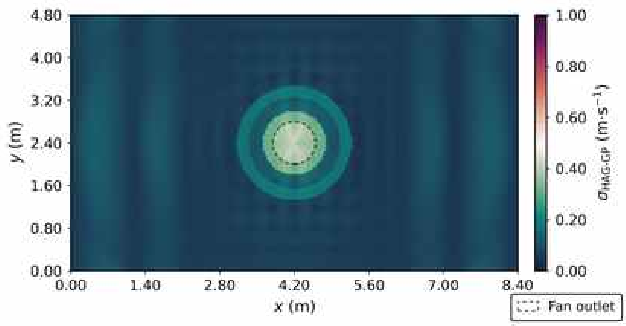
(a) 0.20 m above fan outlet plane.



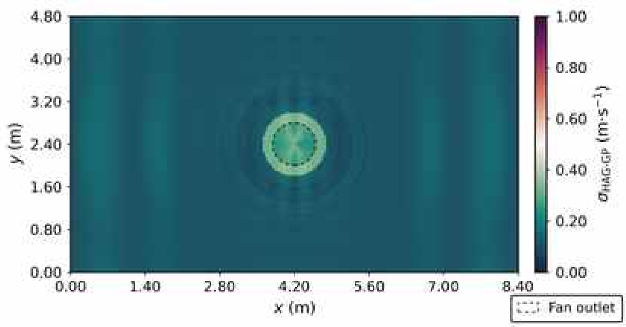
(b) 0.35 m above fan outlet plane.



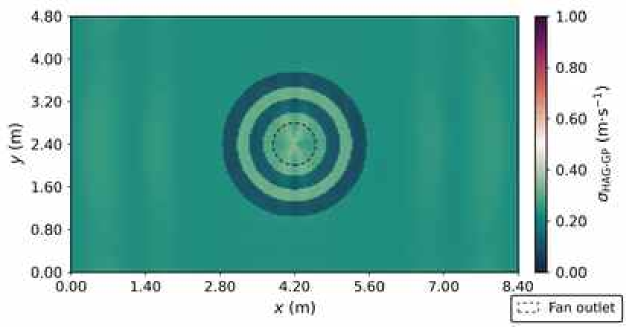
(c) 0.50 m above fan outlet plane.



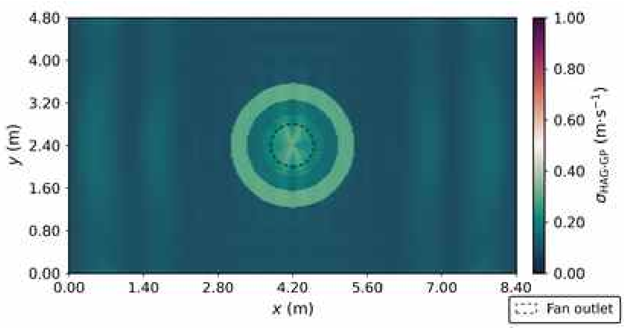
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



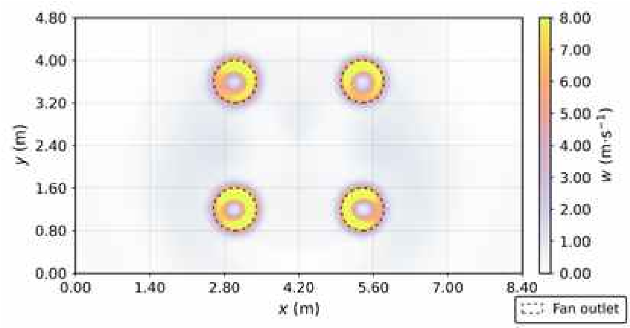
(f) 1.60 m above fan outlet plane.



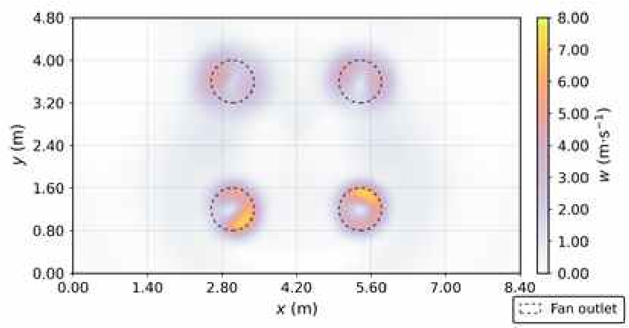
(g) 2.20 m above fan outlet plane.

Fig. D.11. Supplementary total fluctuation field, $\sigma_{\mathrm{HAG-GP}}$, of the residual annular Gaussian process surrogate (single fan). The grey dotted rings indicate the annular regions used to assign the empirical fluctuation level, σ_i, as described at the beginning of Section 4.3.

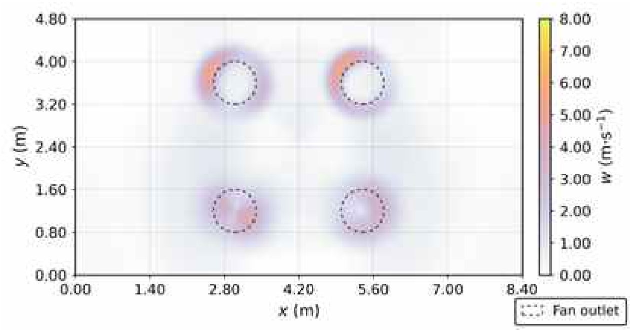
D.5.2 Four-Fan Residual Annular Gaussian Process Surrogate



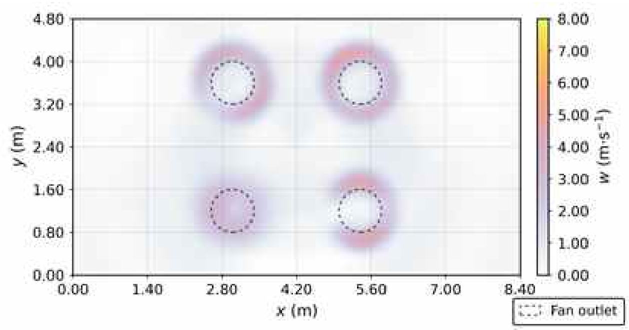
(a) 0.20 m above fan outlet plane.



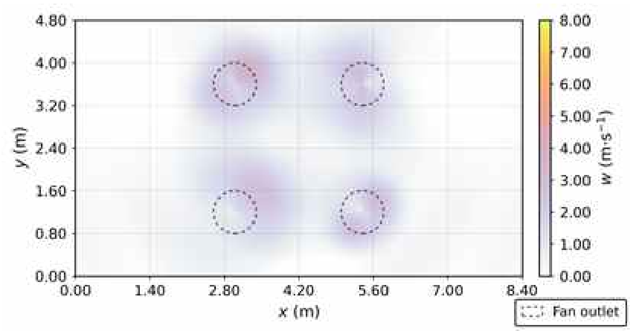
(b) 0.35 m above fan outlet plane.



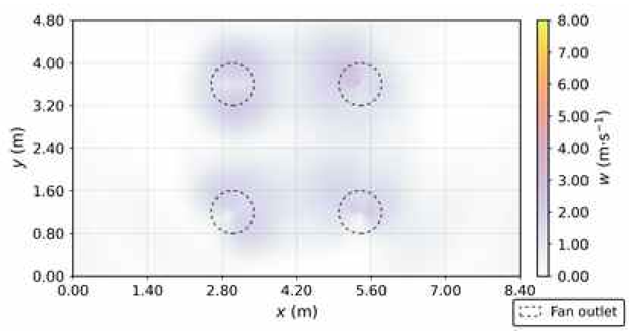
(c) 0.50 m above fan outlet plane.



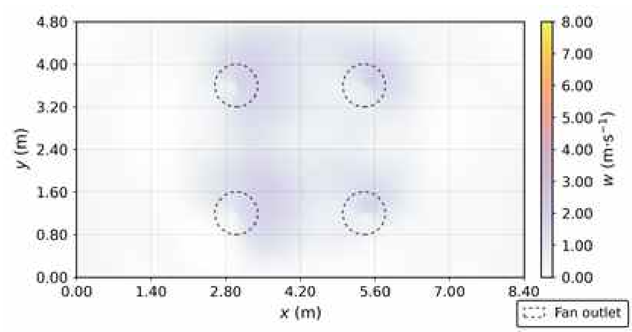
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.

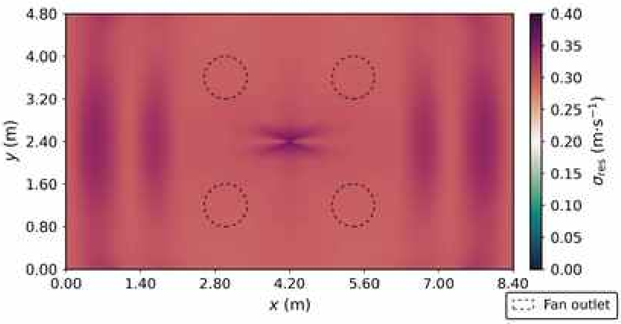


(f) 1.60 m above fan outlet plane.

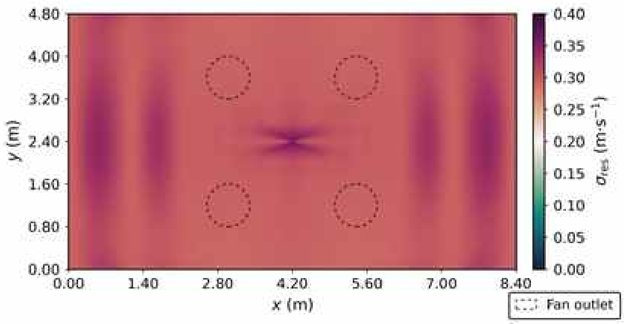


(g) 2.20 m above fan outlet plane.

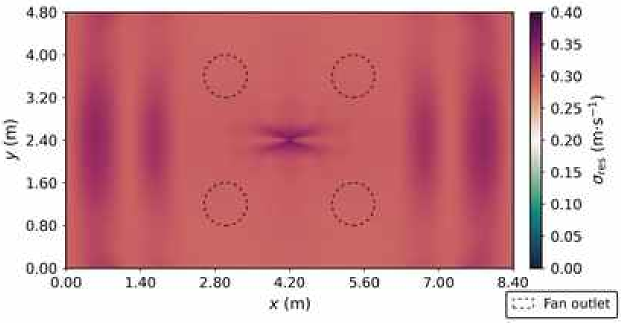
Fig. D.12. Supplementary spatial distribution of fitted mean vertical velocity from the residual annular Gaussian process surrogate (four fans).



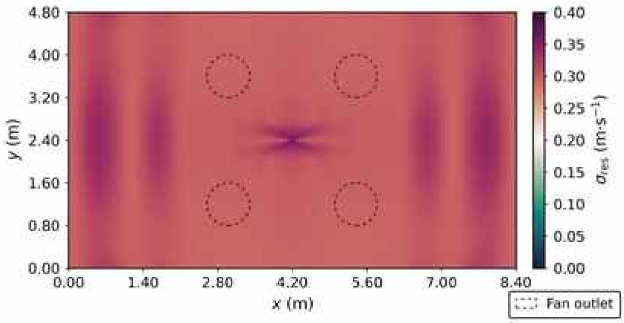
(a) 0.20 m above fan outlet plane.



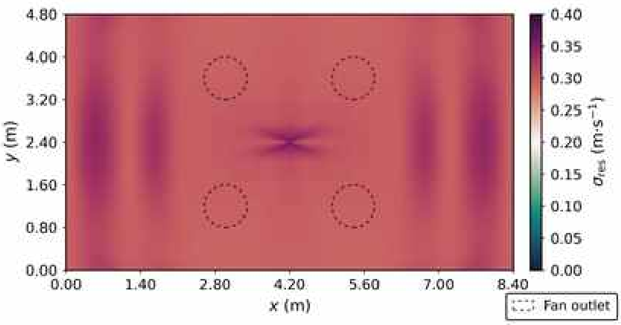
(b) 0.35 m above fan outlet plane.



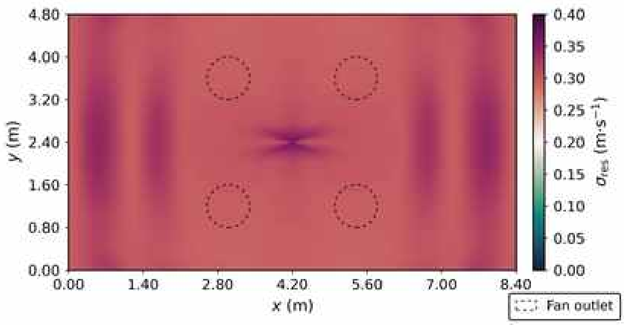
(c) 0.50 m above fan outlet plane.



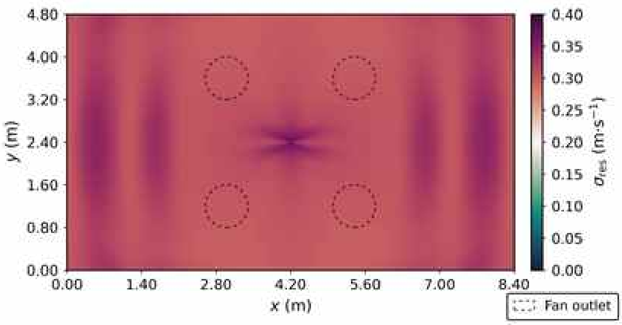
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.

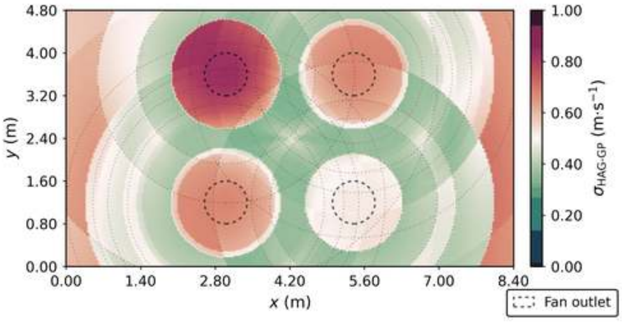


(f) 1.60 m above fan outlet plane.

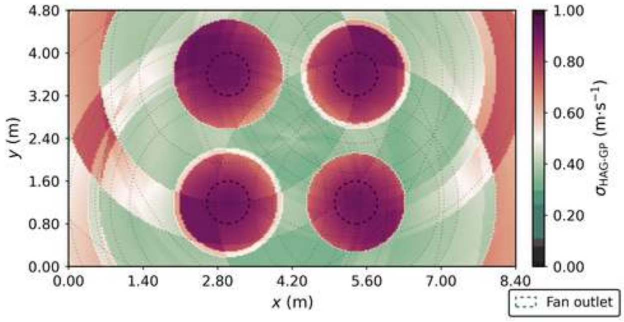


(g) 2.20 m above fan outlet plane.

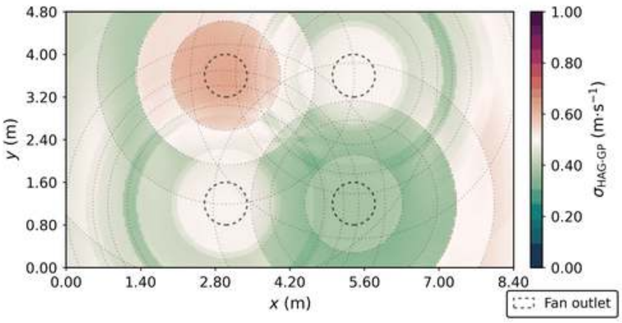
Fig. D.13. Supplementary predictive standard deviation field, σ_{res}, of the residual annular Gaussian process surrogate (four fans).



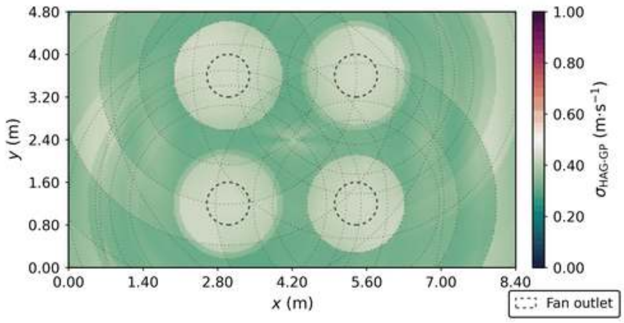
(a) 0.20 m above fan outlet plane.



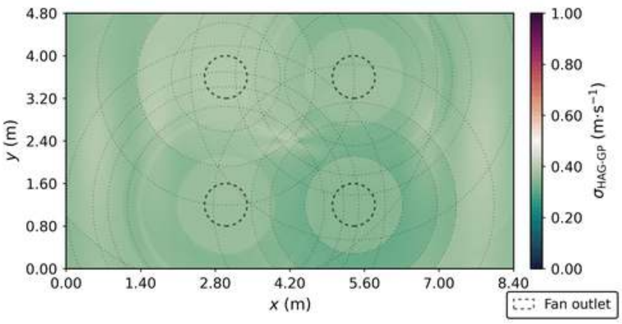
(b) 0.35 m above fan outlet plane.



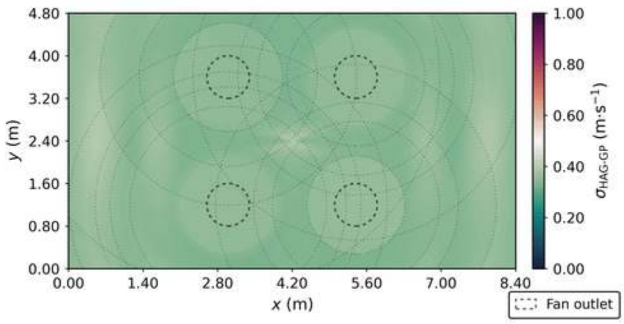
(c) 0.50 m above fan outlet plane.



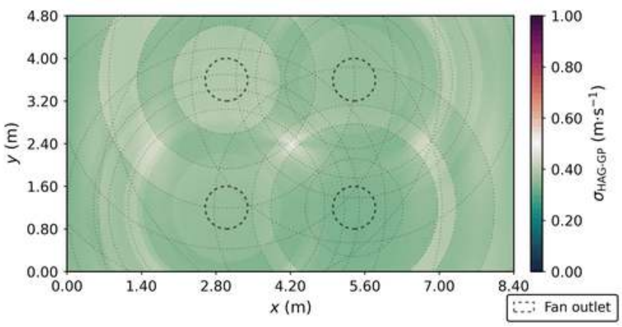
(d) 0.75 m above fan outlet plane.



(e) 1.10 m above fan outlet plane.



(f) 1.60 m above fan outlet plane.



(g) 2.20 m above fan outlet plane.

Fig. D.14. Supplementary total fluctuation field, $\sigma_{\text{HAG}-\text{GP}}$, of the residual annular Gaussian process surrogate (four fans). The grey dotted rings indicate the annular regions used to assign the empirical fluctuation level, σ_i, as described at the beginning of Section 4.3.

E Glider Design and Manufacturing Details

E.1 Glider Optimisation Workflow

E.1.1 Pseudocode for Optimisation Workflow

Algorithm E.1 Glider design optimisation based on AeroSandbox [77].

Input: Design bounds $\mathcal{B}$, fixed parameters θ, mission condition ξ_0, objective J, and requirements $\mathcal{C}$.

Output: Feasible design $d^\star$, solved decision vector $z^\star$, and feasibility audit $A^\star$.

Define $z = \{\alpha, \delta_a, \delta_e, \delta_r, b, c, l_t, b_H, b_V\}$ with $z \in \mathcal{B}$

For a candidate z, construct $G(z;\theta)$, $M(z;\theta)$, and the trim model at ξ_0

Define $J(z)$ and $\mathcal{C}(z)$ using the objective terms and requirements in Tables E.2 and E.3

$z^\dagger \leftarrow \arg\min_{z\in\mathcal{B}} J(z)$ subject to $\mathcal{C}(z) \leq 0$

$A^\dagger \leftarrow$ post-solve feasibility audit of $z^\dagger$

if *the solver converges and* $A^\dagger$ *passes* **then**

  $z^\star \leftarrow z^\dagger$, $d^\star \leftarrow d(z^\star)$, and $A^\star \leftarrow A^\dagger$

else

  $z^\star \leftarrow \emptyset$, $d^\star \leftarrow \emptyset$, and $A^\star \leftarrow A^\dagger$

return $d^\star$, $z^\star$, and $A^\star$, or an infeasibility flag

Algorithm E.2 Multistart uncertainty-aware candidate selection selection.

Input: Initial guesses $\{z_0^{(i)}\}_{i=1}^{N_s}$, retained count K, uncertainty scenarios $\Omega = \{\omega_j\}_{j=1}^{N_\omega}$, and Algorithm E.1.

Output: Robust selected design $d^\star$, selected candidate index $i^\star$, and robust audit $R^\star$.

$\mathcal{F} \leftarrow \emptyset$

for *each initial guess* $z_0^{(i)}$ **do**

  Solve Algorithm E.1 from $z_0^{(i)}$ to obtain (d_i, z_i, A_i)

  if $d_i \neq \emptyset$ *and* A_i *passes* **then**

    Add (d_i, z_i, J_i, A_i) to $\mathcal{F}$

if $\mathcal{F} = \emptyset$ **then**

  return a failure flag

Remove duplicate geometries from $\mathcal{F}$, sort by nominal objective, and retain the best K candidates

for *each retained candidate* $d_i \in \mathcal{F}$ **do**

  $\mathcal{M}_i \leftarrow \emptyset$

  for *each uncertainty scenario* $\omega_j \in \Omega$ **do**

    Apply ω_j to d_i, re-solve trim, and compute scenario feasibility and performance metrics

    Add the scenario result to $\mathcal{M}_i$

  $R_i \leftarrow$ robust audit from $\mathcal{M}_i$

$i^\star \leftarrow \arg\min_i R_i$ using success rate, tail sink metric, and nominal objective as ordered ranking criteria

return $d^\star = d_{i^\star}$, $i^\star$, and $R^\star = R_{i^\star}$

Algorithm E.3 Finite difference sensitivity analysis.

```
Input: Selected design d*, parameter set P, output metrics Q, step ladder H, and stability tolerances.
Output: Sensitivity matrix S, selected step table H*, and stability audit A_S.
S ← ∅, H* ← ∅, A_S ← ∅
Evaluate the baseline model for d* to obtain q_j^0 for all q_j ∈ Q
for each parameter p_i ∈ P do
    Select a central-difference scheme, or a forward scheme if a lower bound is active
    Build the admissible step set H_i ⊆ H
    for each step h_{i,ℓ} ∈ H_i do
        Evaluate feasible perturbed designs at p_i ± h_{i,ℓ}, or p_i + h_{i,ℓ} for a forward scheme
        Store the perturbed metrics and feasibility flags
    for each output metric q_j ∈ Q do
        D_ij ← derivative candidates from feasible perturbation pairs
        e_ij ← truncation and round-off error estimates for D_ij
        if at least one non-round-off step is defensible then
            h*_ij ← arg min_{h_{i,ℓ}} e_ij
            S_ij ← (p_i / max(|q_j^0|, q_{j,min})) ∂q_j/∂p_i at h*_ij
            Add (p_i, q_j, h*_ij, stable) to A_S
        else
            S_ij ← NaN
            Add (p_i, q_j, ∅, unresolved) to A_S
return S, H*, and A_S
```

E.1.2 Optimisation Inputs and Robustness Settings

Table E.1 Design variables and bounds used in the nominal optimisation.

Quantity	Symbol	Unit	Bounds
Trim angle of attack	α_{trim}	deg	[−4.0, 8.0]
Aileron deflection	δ_a	deg	[−30, 30]
Elevator deflection	δ_e	deg	[−30, 30]
Rudder deflection	δ_r	deg	[−30, 30]
Wing span	b	m	[0.30, 0.75]
Wing chord	c	m	[0.10, 0.30]
Tail arm	l_t	m	[0.40, 0.80]
Horizontal tail span	b_H	m	[0.20, 0.50]
Vertical tail height	b_V	m	[0.10, 0.30]
Tail boom length	l_b	m	[0.35, 0.70]

Table E.2 Principal feasibility limits enforced in the optimiser.

Quantity	Symbol	Unit	Limit
Pitching moment tolerance	$\lvert C_m \rvert$	-	1×10^{-6}
Maximum nominal lift coefficient	$C_{L,\max}$	-	1.2
Minimum glide ratio	L/D	-	6.0
Minimum wing Reynolds number	Re_c	-	2×10^4
Wing loading interval	W/S	$\mathrm{N\,m^{-2}}$	[2.0, 20.0]
Static margin interval	SM	MAC	[0.05, 0.10]
Horizontal tail volume	V_H	-	[0.50, 1.00]
Vertical tail volume	V_V	-	[0.03, 0.08]
Roll damping magnitude	$\lvert C_{\ell_p} \rvert$	-	$\geq 1 \times 10^{-4}$
Servo torque rating	T_s	$\mathrm{N\,m}$	0.009

Table E.3 Objective terms and weights used in the nominal optimisation.

Objective	Symbol	Scale	Weight
Sink speed	J_{sink}	$1.0\,\mathrm{m\,s^{-1}}$	1.000
Mass	J_m	$0.10\,\mathrm{kg}$	0.04699
Trim effort	J_{trim}	$10°$	0.01326
Wing deflection	J_{δ_W}	–	2.152
Horizontal tail deflection	J_{δ_H}	–	0.07831
Roll time constant	J_{τ_p}	$0.45\,\mathrm{s}$	0.2471
Mass included in penalty	all aircraft mass		

Table E.4 Solver and candidate selection settings used by the workflow.

Setting	Value
Solver	IPOPT
Maximum iterations	1000
Hessian treatment	limited memory
Multistart solves	30
Retained candidates	10
Multistart seed	20
Uncertainty-aware selection scenarios	100
Scenario seed	30
Geometry deduplication	$[0.010,\ 0.005,\ 0.010]\,\mathrm{m}$
Active set tolerance	1.0×10^{-3}

Table E.5 Uncertainty intervals used in the uncertainty-aware selection.

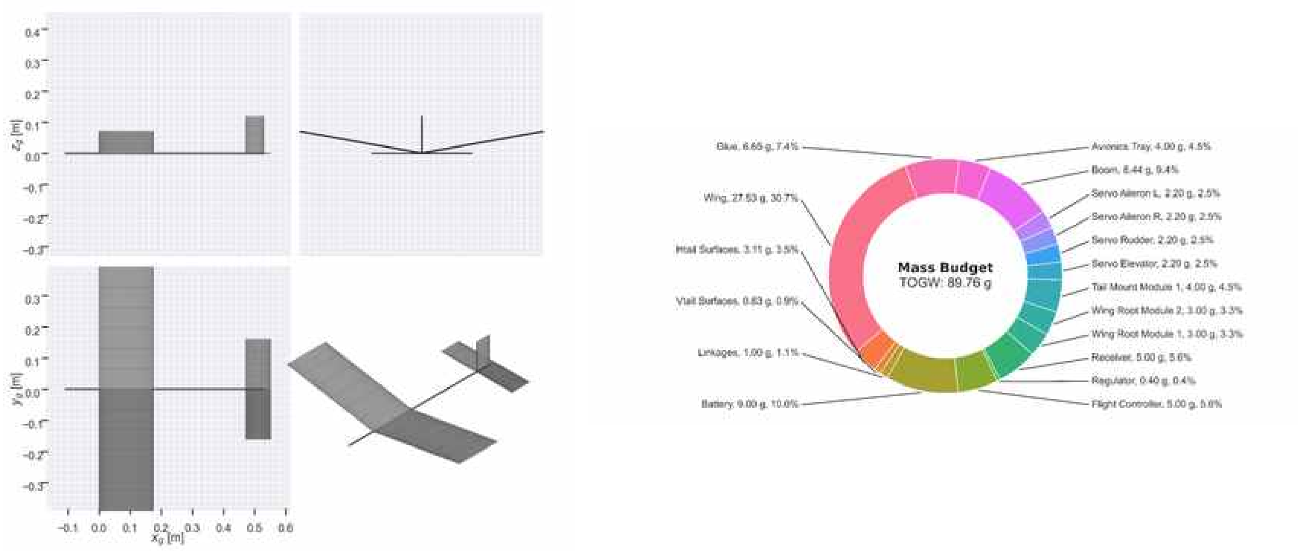
Perturbation	Unit	Interval
Mass scale	-	[1.00, 1.10]
Longitudinal CG shift	MAC	[−0.06, 0.06]
Incidence bias	deg	[−2.0, 2.0]
Control effectiveness	-	[0.85, 1.00]
Control neutral bias	deg	[−3.0, 3.0]
Principal inertia scale	-	[0.90, 1.10]
Surface stiffness scale	-	[0.80, 1.30]
Nominal vertical gust	$\mathrm{m\,s^{-1}}$	[−0.30, 0.30]
Turn vertical gust	$\mathrm{m\,s^{-1}}$	[−0.20, 0.20]
Drag scale	-	[1.00, 1.25]

Table E.6 Robust ranking and rate-check policy used after the nominal solve.

Setting	Value
Selection order	success rate, tail risk, objective
Tail-risk fraction	20%
Default robust-in-loop cases	8
Robust-in-loop case interval	4–8
Scenario families	mild, harsh, gust, compound
Nominal trim time	$1.5\,\mathrm{s}$
Bank entry time	$0.20\,\mathrm{s}$
Rate utilisation allowance	0.80
Gust time constant	$0.50\,\mathrm{s}$
Robust gust flag	on

E.2 Optimised Glider Design Iterations

E.2.1 First Iteration

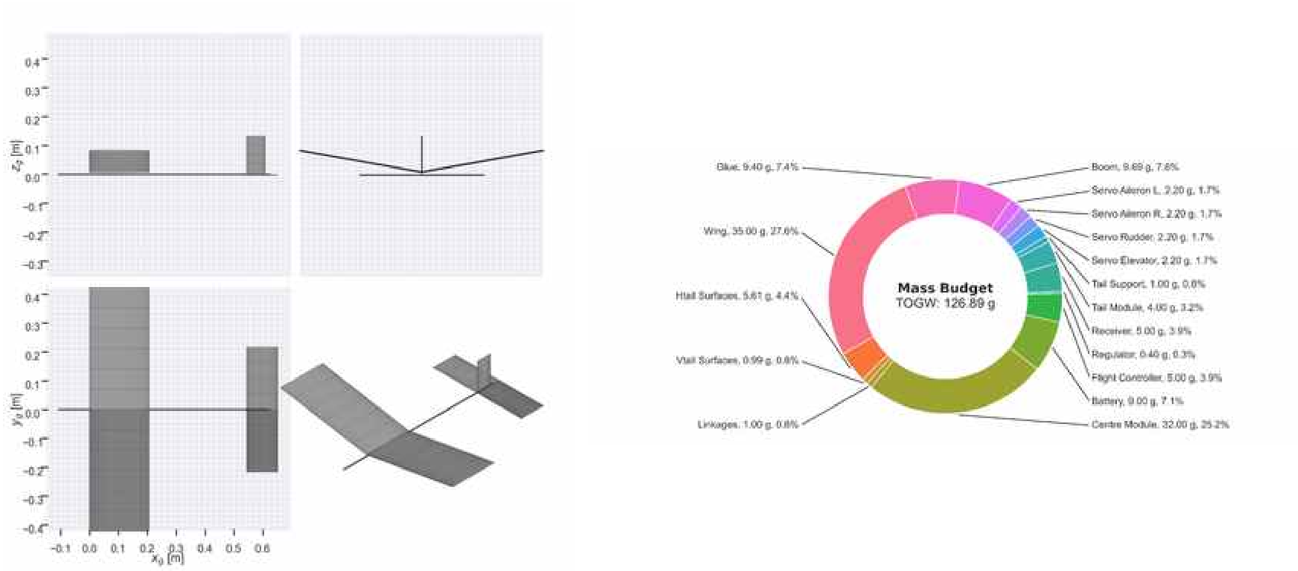


(a) Three-view geometry.

(b) Mass breakdown.

Fig. E.1. First iteration optimised glider configuration.

E.2.2 Second Iteration



(a) Three-view geometry.

(b) Mass breakdown.

Fig. E.2. Second iteration optimised glider configuration.

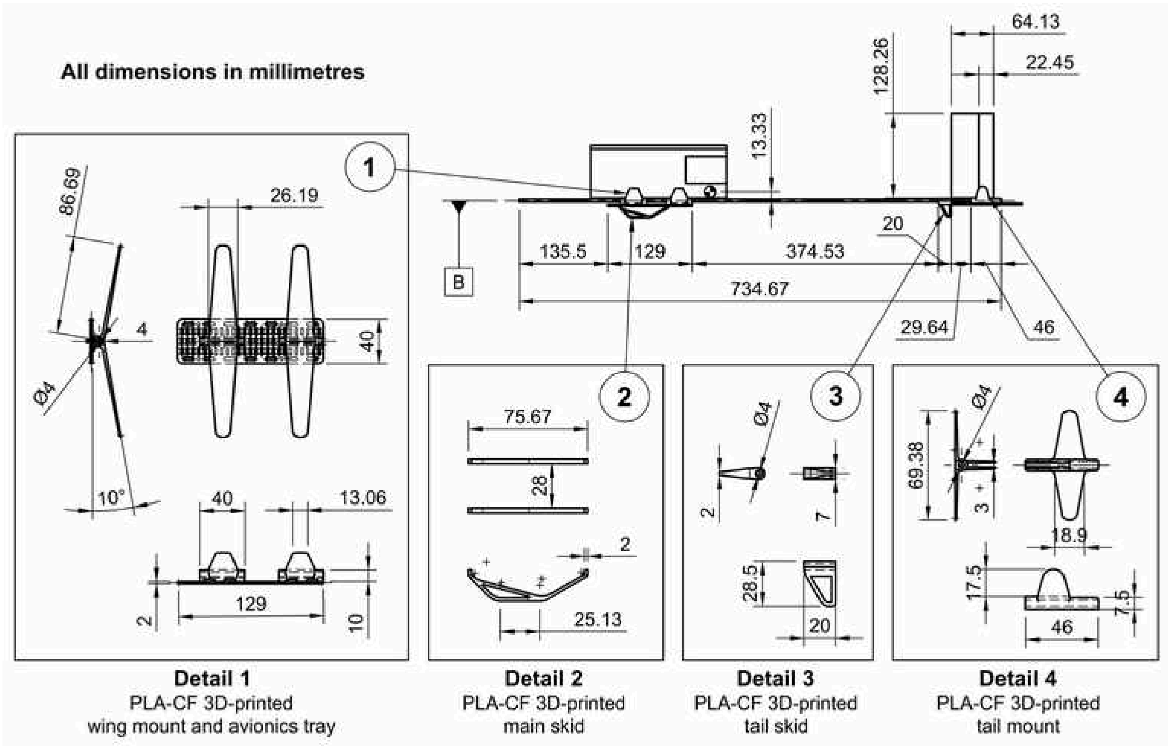
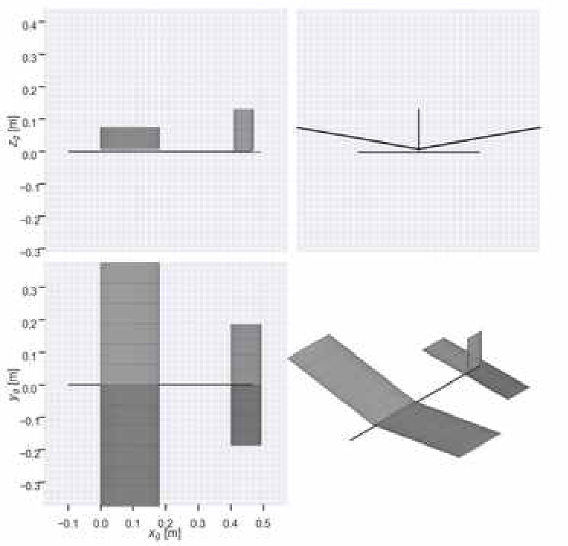
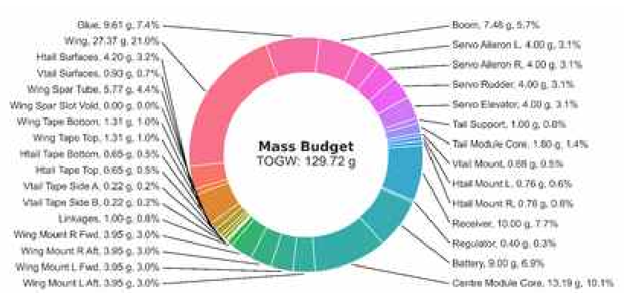


Fig. E.3. Detail engineering drawing of the second iteration glider assembly, corresponding to a half-span of 0.428 m and chord length of 0.207 m. Key 3D-printed mounting and skid components are shown; avionics, wiring, and tape reinforcements are omitted for clarity.

E.2.3 Third Iteration



(a) Three-view geometry.



(b) Mass breakdown.

Fig. E.4. Third iteration optimised glider configuration.

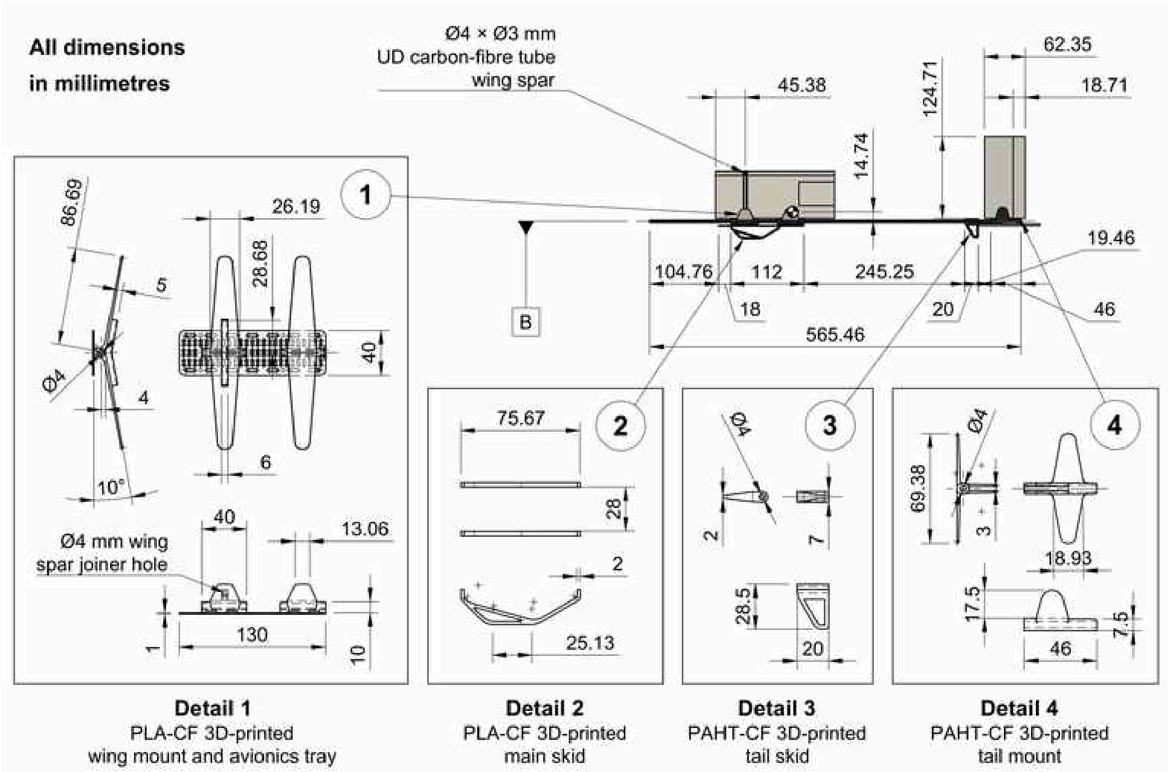
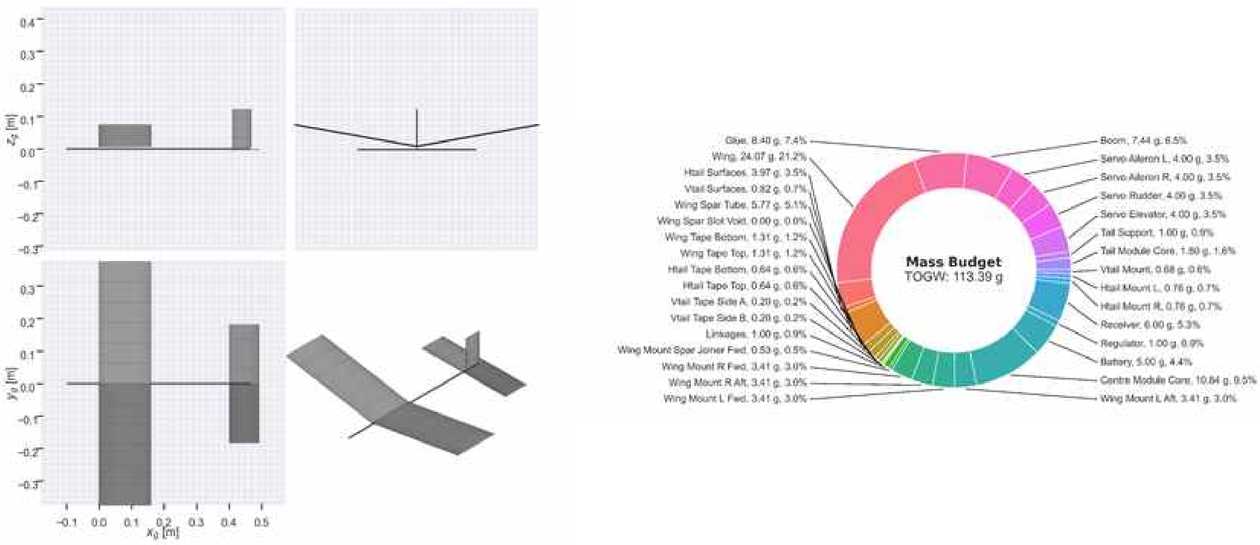


Fig. E.5. Detail engineering drawing of the third iteration glider assembly, corresponding to a half-span of 0.381 m and chord length of 0.182 m. Key 3D-printed mounting and skid components are shown; avionics, wiring, and tape reinforcements are omitted for clarity.

E.2.4 Fourth Iteration



(a) Three-view geometry. **(b)** Mass breakdown.

Fig. E.6. Fourth iteration optimised glider configuration.

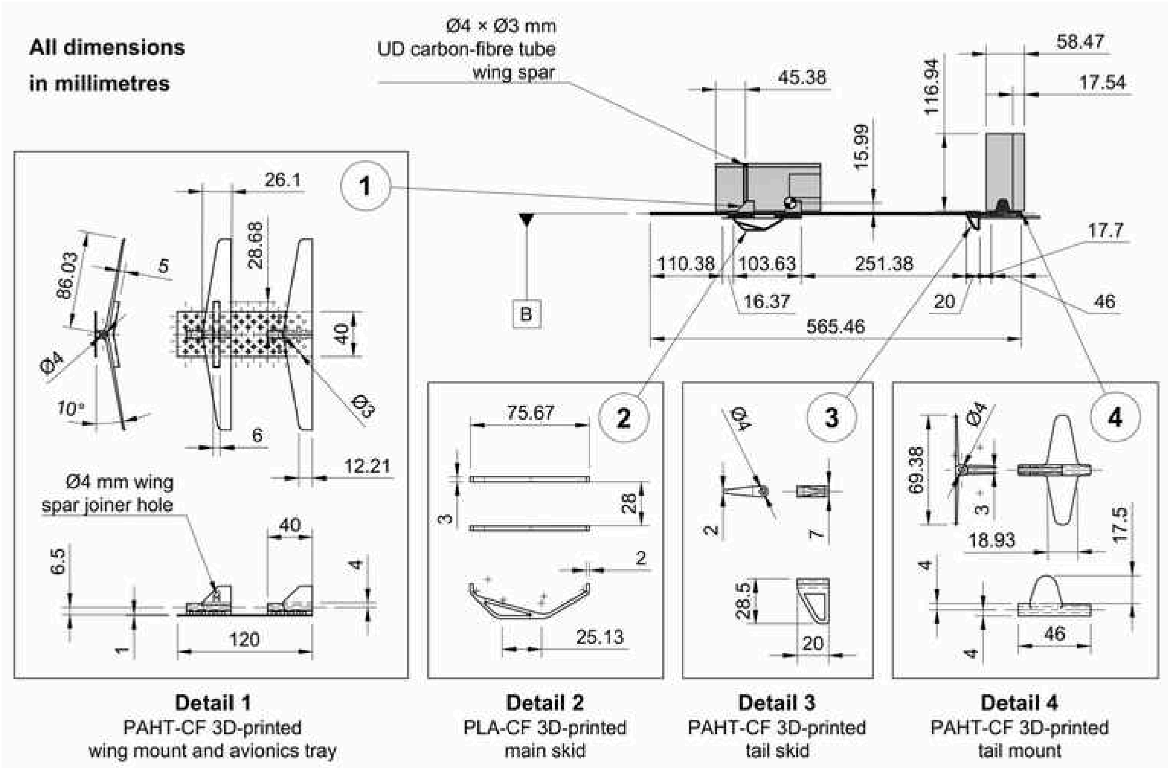
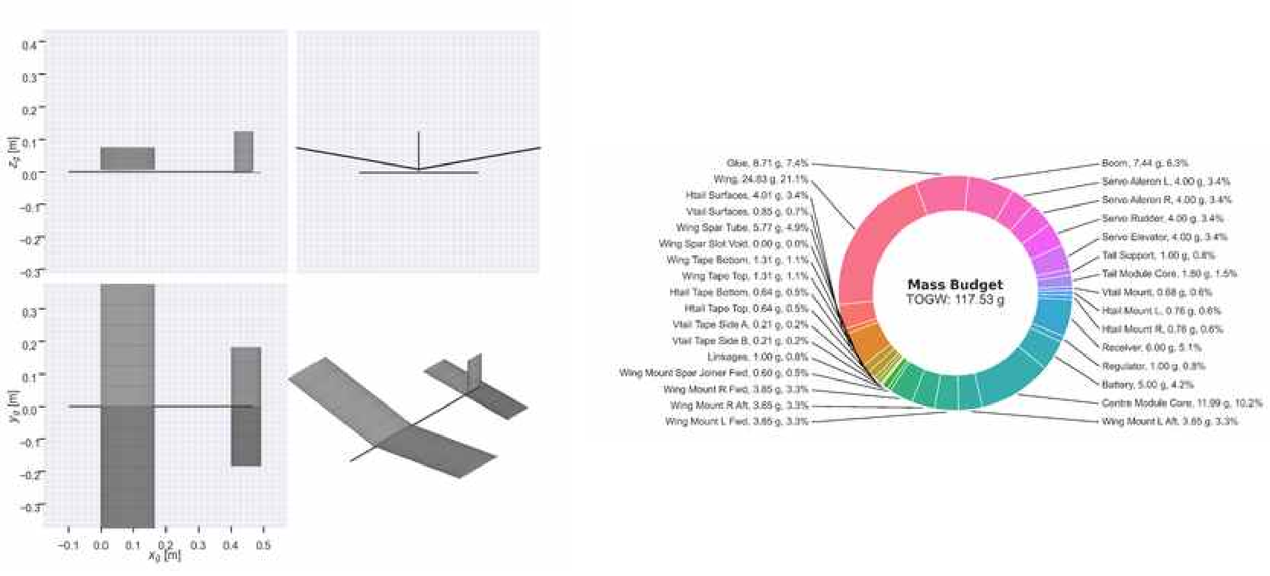


Fig. E.7. Detail engineering drawing of the fourth iteration glider assembly, corresponding to a half-span of 0.381 m and chord length of 0.160 m. Key 3D-printed mounting and skid components are shown; avionics, wiring, and tape reinforcements are omitted for clarity.

E.2.5 Fifth Iteration



(a) Three-view geometry. **(b)** Mass breakdown.

Fig. E.8. Fifth iteration optimised glider configuration.

E.3 Glider Optimisation Model and Selected Design

E.3.1 Model Assumptions

Table E.7 Mission, arena, and aerodynamic settings used by the optimiser.

Quantity	Unit	Value
Nominal design speed	$\mathrm{m\,s^{-1}}$	6.50
Turn diagnostic speed	$\mathrm{m\,s^{-1}}$	4.00
Commanded bank angle	deg	50.0
Bank entry time	s	0.20
Arena length	m	8.40
Arena width	m	4.80
Arena height	m	3.50
Wall clearance allowance	m	0.20
Airfoil section	–	NACA0002
Wing dihedral	deg	10.0

Table E.8 Planform and control surface assumptions used by the optimiser.

Quantity	Unit	Value
Horizontal tail aspect ratio	-	4.0
Vertical tail aspect ratio	-	2.0
Wing section count	count	7
Tail section count	count	5
Aileron span stations	semispan	[0.30, 0.85]
Aileron chord ratio	-	0.30
Elevator chord ratio	-	0.30
Rudder chord ratio	-	0.35
Nose station	m	-0.10
Fuselage radius	m	0.002

Table E.9 Material and reinforcement assumptions used in the mass and stiffness model.

Quantity	Unit	Value
Wing foam density	$\mathrm{kg\,m^{-3}}$	33.0
Tail foam density	$\mathrm{kg\,m^{-3}}$	40.0
Wing foam thickness	mm	6.00
Tail foam thickness	mm	3.00
Wing secant modulus	Pa	1.5×10^6
Tail secant modulus	Pa	1.0×10^6
Wing spar outer diameter	mm	4.00
Wing spar inner diameter	mm	3.00
Spar modulus	Pa	25×10^9
Tape width	mm	10.0
Tape thickness	mm	0.13
Tape areal density	$\mathrm{kg\,m^{-2}}$	0.175

Table E.10 Onboard mass and actuation assumptions used in the design model.

Quantity	Unit	Value
Centre module mass	g	28.0
Tail module mass	g	4.0
Battery mass	g	5.0
Receiver mass	g	6.0
Regulator mass	g	1.0
Servo mass, each	g	4.0
Servo count	count	4
Control deflection limits	deg	± 30
Servo rate limit	$\mathrm{deg\,s^{-1}}$	400
Servo torque rating	N m	0.009

E.3.2 Selected Design

Table E.11 Geometry and mass properties.

Quantity	Symbol	Unit	Value
Wing span	b	m	0.7616
Wing chord	c	m	0.1646
Wing area	S	m^2	0.1254
Wing aspect ratio	Æ	-	4.626
Tail moment arm	l_t	m	0.4000
Horizontal tail span	b_H	m	0.3656
Vertical tail height	b_V	m	0.1188
Aircraft mass	m	g	117.5
Longitudinal CG	x_{CG}	m	0.1137
Lateral CG	y_{CG}	m	0
Vertical CG	z_{CG}	m	0.0158

Table E.12 Trim and performance.

Quantity	Symbol	Unit	Value
Trim angle of attack	α_{trim}	deg	3.808
Trim elevator deflection	$\delta_{e,\mathrm{trim}}$	deg	-0.114
Sink speed	V_{sink}	$\mathrm{m\,s^{-1}}$	0.6703
Glide ratio	L/D	-	9.697
Wing loading	W/S	$\mathrm{N\,m^{-2}}$	9.195
Wing Reynolds number	Re_c	-	7.33×10^4
Static margin	SM	MAC	0.100
Horizontal tail volume	V_H	-	0.647
Vertical tail volume	V_V	-	0.030
Lift coefficients	C_L	-	0.355
Drag coefficients	C_D	-	0.0366

Table E.13 Grouped mass budget.

Group	Share	Mass
Wing and tail surfaces	25.3%	29.68 g
Spar, tape, and slot	8.3%	9.70 g
Mounts and joiners	15.5%	18.22 g
Boom and tail hardware	8.7%	10.24 g
Avionics and servos	34.9%	40.99 g
Glue allowance	7.4%	8.71 g
Total	100%	117.5 g

Table E.14 Objective decomposition.

Objective term	Value
Total objective, J	2.166
Sink speed term	0.670
Mass term	0.0552
Trim effort term	1.72×10^{-6}
Wing deflection term	1.311
Horizontal tail deflection term	0.117
Roll time constant term	0.0126

Table E.15 Turn footprint and roll response.

Quantity	Symbol	Unit	Value
Turn lift coefficient	$C_{L,\mathrm{turn}}$	-	0.355
Allowable turn radius	R_{allow}	m	1.825
Predicted turn radius	R_{ach}	m	2.129
Predicted turn footprint	d_{turn}	m	2.704
Required lateral acceleration	$a_{y,\mathrm{req}}$	$\mathrm{m\,s^{-2}}$	8.767
Predicted lateral acceleration	$a_{y,\mathrm{ach}}$	$\mathrm{m\,s^{-2}}$	7.515
Lateral acceleration margin	Δa_y	$\mathrm{m\,s^{-2}}$	-1.252
Steady roll rate	p_{ss}	$\mathrm{rad\,s^{-1}}$	5.240
Roll time constant	τ_p	s	0.0229
Bank angle in 0.2 s	$\phi(t_{\mathrm{p}})$	deg	25.60

Table E.16 Structural and stability.

Quantity	Symbol	Unit	Value
Wing deflection ratio	$\delta_W/\delta_{W,\max}$	-	0.272
Horizontal tail deflection ratio	$\delta_H/\delta_{H,\max}$	-	0.140
Maximum servo torque utilisation	u_s	-	0.00481
Lift curve slope	C_{L_α}	$\mathrm{rad^{-1}}$	5.648
Pitch stability	C_{m_α}	$\mathrm{rad^{-1}}$	-0.565
Dihedral effect	C_{ℓ_β}	$\mathrm{rad^{-1}}$	-0.199
Directional stability	C_{n_β}	$\mathrm{rad^{-1}}$	0.0257
Roll damping	C_{ℓ_p}	-	-0.779
Pitch damping	C_{m_q}	-	-11.47
Aileron effectiveness	$C_{\ell_{\delta_a}}$	$\mathrm{rad^{-1}}$	-0.827

E.4 Material Properties and Additive Manufacturing Settings

E.4.1 Glider Material Properties

Table E.17 summarises the material and section properties used in the glider sizing model. Since the Depron datasheet [67] gives compressive stress at 10% deformation pnly, the model used the secant estimate $E_{\text{sec},10} = \sigma_{d10}/0.10$. Carbon tube and rod section properties were computed from the listed diameters using standard circular section formulae. For the 3D-printed modules, measured printed mass and CAD volume were used to obtain the effective density used in the mass model, while datasheet values are listed for reference.

Table E.17 Glider materials, datasheet references, and sizing model properties.

Material	Part	Dimension	Datasheet values	Model properties	Source
Depron	Wing lifting surface	6 mm thickness	$\sigma_{d10} = 150\,\text{kPa}$	$\rho = 33\,\text{kg}\,\text{m}^{-3}$, $E_{\text{sec},10} = 1.5\,\text{MPa}$	[67]
	Tail lifting surfaces	3 mm thickness	$\sigma_{d10} = 100\,\text{kPa}$	$\rho = 40\,\text{kg}\,\text{m}^{-3}$, $E_{\text{sec},10} = 1.0\,\text{MPa}$	
UD carbon-fibre tube	Fuselage boom	4/2 mm OD/ID	$\rho = 1400\,\text{kg}\,\text{m}^{-3}$, $E_f = 25\,\text{GPa}$	$A = 9.42\,\text{mm}^2$, $I = 11.8\,\text{mm}^4$, $\mu = 13.2\,\text{g}\,\text{m}^{-1}$, $EI = 0.295\,\text{N}\,\text{m}^2$	[167]
	Wing spar	4/3 mm OD/ID		$A = 5.50\,\text{mm}^2$, $I = 8.59\,\text{mm}^4$, $\mu = 7.70\,\text{g}\,\text{m}^{-1}$, $EI = 0.215\,\text{N}\,\text{m}^2$	
UD carbon-fibre rod	Long linkage member	1.5 mm diameter	$\rho = 1400\,\text{kg}\,\text{m}^{-3}$, $E_f = 25\,\text{GPa}$	$A = 1.77\,\text{mm}^2$, $I = 0.249\,\text{mm}^4$, $\mu = 2.47\,\text{g}\,\text{m}^{-1}$, $EI = 0.00621\,\text{N}\,\text{m}^2$	[167]
304 stainless-steel wire	Pushrod wire	0.8 mm diameter	-	$\rho = 8000\,\text{kg}\,\text{m}^{-3}$, $E = 187.5\,\text{GPa}$, $G = 70.3\,\text{GPa}$	[183]
PLA-CF	Centre module and landing gear	-	$\rho = 1.22\,\text{g}\,\text{cm}^{-3}$, $E_{XY} = 2.79\,\text{GPa}$, $E_{b,XY} = 3.95\,\text{GPa}$	$\rho_{\text{eff}} = 1192\,\text{kg}\,\text{m}^{-3}$	[168]
PAHT-CF	Tail module and tail skid	-	$\rho = 1.06\,\text{g}\,\text{cm}^{-3}$, $E_{XY} = 3.86\,\text{GPa}$, $E_{b,XY} = 4.23\,\text{GPa}$	$\rho_{\text{eff}} = 1108\,\text{kg}\,\text{m}^{-3}$	[169]
WEST 105/205 epoxy	Bonding allowance	-	$\rho = 1.11\,\text{g}\,\text{cm}^{-3}$, $E_t = 2.81\,\text{GPa}$, $E_b = 3.18\,\text{GPa}$	8% glue mass allowance	[181]
Cross-weave filament tape	Tape reinforcement	0.13 mm thickness	$t = 0.125\,\text{mm}$, $T_w = 135\,\text{N}\,\text{cm}^{-1}$	$\bar{m} = 0.175\,\text{kg}\,\text{m}^{-2}$, $E_{\text{eff}} = 10\,\text{GPa}$	[182]

E.4.2 Additive Manufacturing Settings

Table E.18 Printer, material, and geometric print settings used for 3D-printed components.

Setting	Option
Printer	Bambu Lab X1 Carbon
Nozzle diameter	0.4 mm
Filament	PLA-CF [168]
	PAHT-CF [169]
Process preset	0.20 mm Strength
Layer height	0.20 mm
Initial layer height	0.20 mm
Default line width	0.42 mm
Initial layer line width	0.50 mm
Inner wall line width	0.45 mm
Sparse infill line width	0.45 mm
Elephant foot compensation	0.15 mm
X-Y hole and contour compensation	0 mm

Table E.19 Shell and infill settings used for 3D-printed components.

Setting	Option
Wall generator	Classic
Wall loops	4
Wall order	Internal perimeter first
Top surface pattern	Monotonic
Top shell layers	5
Top shell thickness	1 mm
Bottom surface pattern	Monotonic
Bottom shell layers	3
Internal solid infill pattern	Rectilinear
Sparse infill density	15%
Sparse infill pattern	Gyroid
Infill/wall overlap	15%
Infill direction	45°
Bridge direction	0°

Table E.20 Speed and acceleration settings used for 3D-printed components.

Setting	Option
Initial layer speed	$50\,\mathrm{mm\,s^{-1}}$
Initial layer infill speed	$105\,\mathrm{mm\,s^{-1}}$
Outer wall speed	$60\,\mathrm{mm\,s^{-1}}$
Inner wall speed	$300\,\mathrm{mm\,s^{-1}}$
Sparse infill speed	$270\,\mathrm{mm\,s^{-1}}$
Internal solid infill speed	$250\,\mathrm{mm\,s^{-1}}$
Top surface speed	$200\,\mathrm{mm\,s^{-1}}$
Bridge speed	$50\,\mathrm{mm\,s^{-1}}$
Gap infill speed	$250\,\mathrm{mm\,s^{-1}}$
Travel speed	$500\,\mathrm{mm\,s^{-1}}$
Normal printing acceleration	$10000\,\mathrm{mm\,s^{-2}}$
Travel acceleration	$10000\,\mathrm{mm\,s^{-2}}$
Initial layer travel acceleration	$6000\,\mathrm{mm\,s^{-2}}$
Slow down for overhangs	On

Table E.21 Support and bed-adhesion settings used for 3D-printed components.

Setting	Option
Enable support	On
Support type	Tree
Threshold angle	30°
On build plate only	On
Remove small overhangs	On
Raft layers	0
Top and bottom Z distance	0.20 mm
Top interface layers	2
Top interface spacing	0.5 mm
Support/object X-Y distance	0.35 mm
Support/object first layer gap	0.20 mm
Brim type	Outer brim
Brim width	5 mm
Brim-object gap	0.24 mm

E.5 Manufactured Glider Configuration

E.5.1 Glider Configuration Before and After Added Ballast

Table E.22 Manufactured glider configuration before and after added ballast.

Quantity	Symbol	Unit	Pre-ballast	Ballasted
Total mass	m	g	133.56	148.56
Added nose ballast	m_b	g	-	15.00
Ballast station from wing leading edge	x_b	m	-	-0.075
Wing span	b	m	0.764	
Wing chord	c	m	0.165	
Wing area	S	m^2	0.126	
Wing loading	$W\,S^{-1}$	$\mathrm{N\,m^{-2}}$	10.4	11.6
Longitudinal CG from wing leading edge	x_{CG}	m	0.126	0.106
Longitudinal CG	$x_{\mathrm{CG}}\,c^{-1}$	%MAC	76.4	63.9
Vertical CG from fuselage rod centreline	z_{CG}	m	-0.00894	-0.00804
Body-axis roll inertia	I_{xx}	$\mathrm{kg\,m^2}$	2.702×10^{-3}	2.703×10^{-3}
Body-axis pitch inertia	I_{yy}	$\mathrm{kg\,m^2}$	2.459×10^{-3}	2.988×10^{-3}
Body-axis yaw inertia	I_{zz}	$\mathrm{kg\,m^2}$	5.079×10^{-3}	5.606×10^{-3}
Body-axis product of inertia	I_{xz}	$\mathrm{kg\,m^2}$	4.763×10^{-5}	2.315×10^{-5}

E.5.2 Mass Distribution Model

The mass distribution model in Tables E.25 to E.26 lists the component-level inputs used to compute the vehicle mass, centre of gravity, and inertia in Table E.22.

Each manufactured part is represented by a lumped mass located at its estimated component centre. The coordinate system follows the build frame used during measurement: x is measured aft from the wing leading edge, y is positive to starboard, and z is positive downward from the fuselage rod centreline. Therefore, negative entries indicate components ahead of the wing leading edge, to port, or above the fuselage rod centreline.

Table E.23 Mass distribution model for avionics, servos, and pushrods in the final configuration.

Component	Mass	x location	y location	z location
Battery	9.380 g	2.400 cm	0 cm	1.625 cm
Receiver	8.900 g	12.100 cm	0 cm	1.285 cm
Right/left aileron servo	4.000 g	5.700 cm	±19.934 cm	−1.968 cm
Rudder servo	4.000 g	12.000 cm	1.500 cm	−0.350 cm
Elevator servo	4.000 g	12.000 cm	−1.500 cm	−0.350 cm
Rudder pushrod	2.000 g	12.600 cm	2.700 cm	0 cm
Elevator pushrod	2.000 g	12.600 cm	−2.700 cm	0 cm

Table E.24 Mass distribution model for mounts, boom, and centre structure in the final configuration.

Component	Mass	x location	y location	z location
Right/left forward wing mount	3.854 g	4.618 cm	±3.743 cm	−1.054 cm
Right/left aft wing mount	3.854 g	11.813 cm	±3.743 cm	−1.054 cm
Wing spar joiner	0.596 g	4.300 cm	0 cm	−1.050 cm
Centre module core	11.986 g	6.555 cm	0 cm	0.295 cm
Right/left horizontal tail mount	0.763 g	44.875 cm	±1.469 cm	0.675 cm
Vertical tail mount	0.679 g	43.815 cm	0 cm	−1.456 cm
Tail module core	0.795 g	44.385 cm	0 cm	0.400 cm
Tail skid	1.000 g	39.431 cm	0 cm	0.937 cm
Fuselage rod	8.236 g	18.400 cm	0 cm	0 cm

Table E.25 Mass distribution model for lifting surfaces and reinforcements in the final configuration.

Component	Mass	x location	y location	z location
Right/left wing foam half	12.480 g	8.250 cm	±19.037 cm	−2.695 cm
Right/left horizontal tail foam half	1.987 g	44.875 cm	±9.100 cm	0.600 cm
Vertical tail foam	0.843 g	44.775 cm	0 cm	−6.450 cm
Wing spar tube	6.326 g	4.300 cm	0 cm	−2.600 cm
Wing spar adhesive	1.898 g	4.300 cm	0 cm	−2.600 cm
Removed spar slot foam	−0.403 g	4.300 cm	0 cm	−2.600 cm
Bottom wing tape	1.438 g	4.250 cm	0 cm	−2.845 cm
Top wing tape	0.693 g	4.250 cm	0 cm	−4.008 cm
Horizontal tail bottom tape	1.274 g	45.425 cm	0 cm	0.743 cm
Horizontal tail top tape	1.274 g	45.425 cm	0 cm	0.456 cm
Vertical tail side tape pair	0.315 g	45.625 cm	±0.157 cm	−7.900 cm

Table E.26 Mass distribution model for tracking markers, ballast, and residual mass in the final configuration.

Component	Mass	x location	y location	z location
Forward fuselage Vicon marker	1.600 g	−10.750 cm	0 cm	0 cm
Wing leading edge Vicon marker	1.600 g	−0.750 cm	0 cm	−1.200 cm
Right/left wingtip Vicon marker	1.600 g	4.500 cm	±38.825 cm	−3.940 cm
Nose ballast	15.000 g	−7.500 cm	0 cm	0 cm
Residual adhesive and manufacturing mass	5.737 g	38.372 cm	0 cm	0 cm

E.5.3 Manufactured Airframe Views

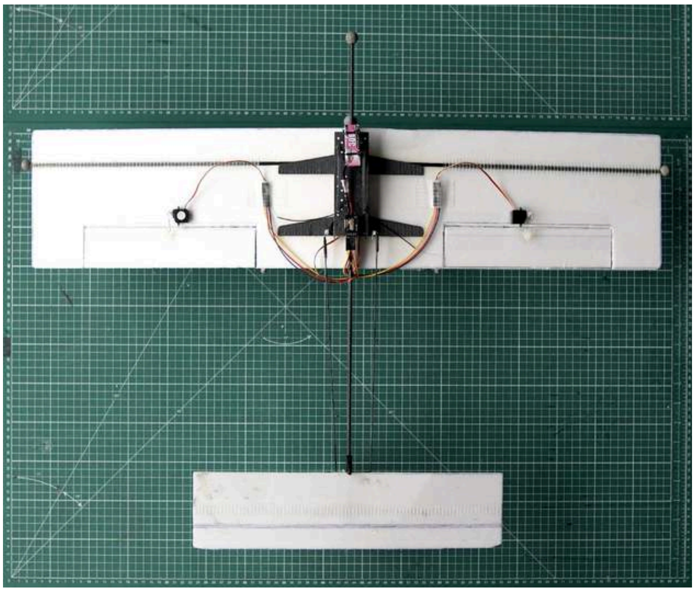


Fig. E.9. Bottom view of the manufactured glider, showing the centre module, receiver wiring, aileron servos, pushrods, wing spar, and tape reinforcement.

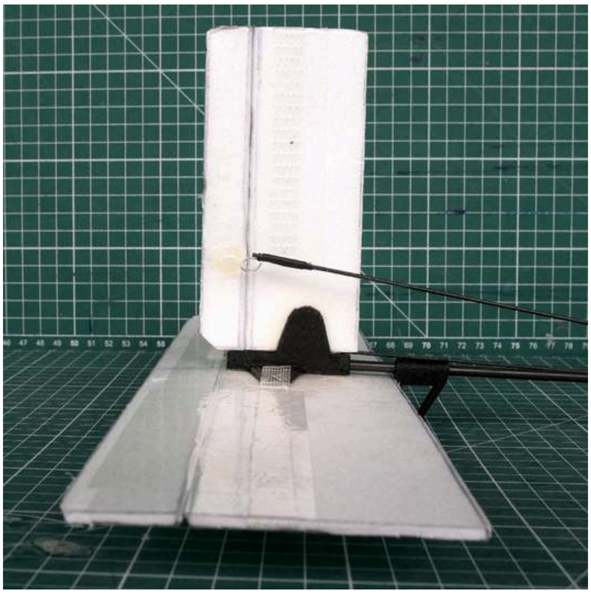


Fig. E.10. Side view of the tail assembly, showing the vertical tail, rudder linkage, tail skid, and reinforcement around the tail mount.

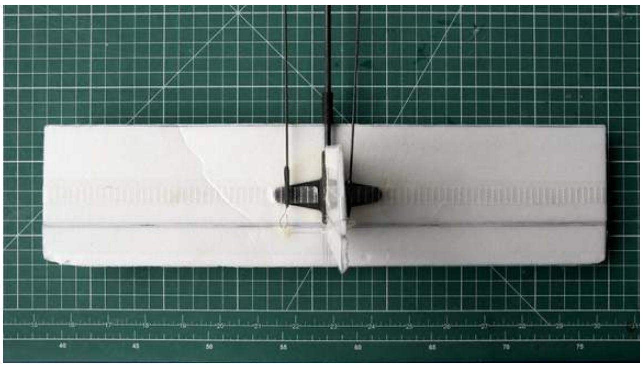


Fig. E.11. Top view of the tail assembly, showing the horizontal tail, tail boom attachment, pushrod routing, and 3D-printed tail mount structure.

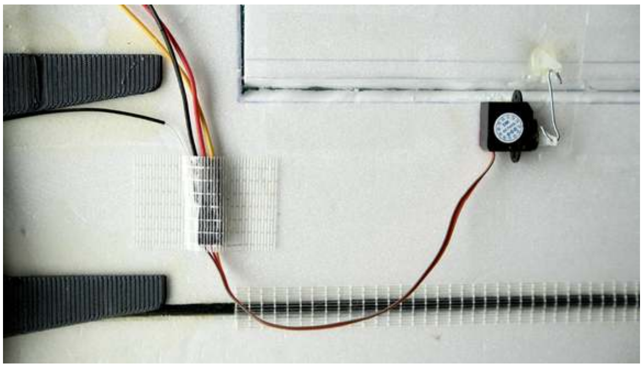


Fig. E.12. Aileron installation detail, showing the wing-mounted servo, pushrod linkage, hinge line, and tape reinforcement.

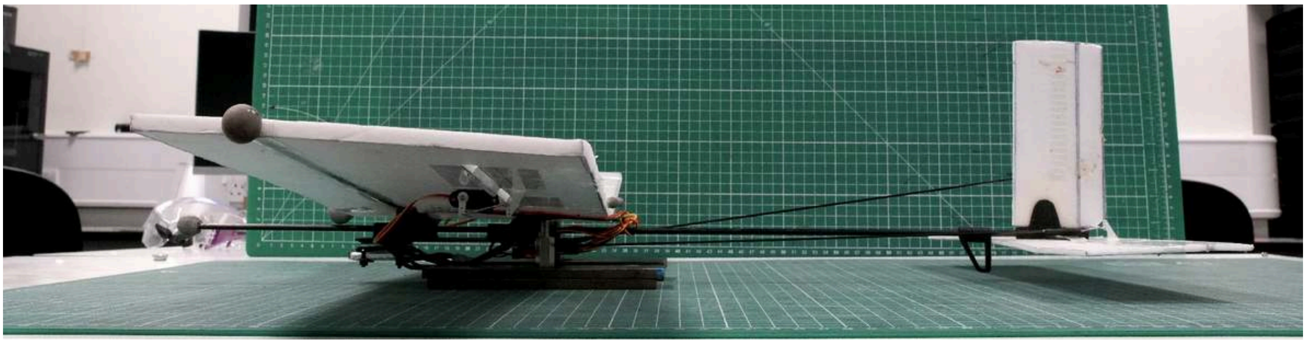


Fig. E.13. Side view of the manufactured glider during longitudinal CG measurement.

F Controller Development and Validation Ladder Setup

F.1 Control-Oriented Glider Model

Section 6.2 presents the control-oriented glider model compactly, the present appendix writes the same equations in scalar flight-dynamics form so that the signs, aerodynamic strip construction, and calibrated residual terms can be checked directly against the reported tables.

The body axes follow the standard aircraft convention used throughout the control model: x_b is forward, y_b is starboard, and z_b is downward, as illustrated in Fig. 3.5. As shown in Fig. 3.1, the arena frame uses z positive upward, so vertical position changes acquire the opposite sign after rotating body-axis velocity into the arena frame. With $\mathbf{v}_b = [u_b,\, v_b,\, w_b]^\top$, $\boldsymbol{\omega}_b = [p_b,\, q_b,\, r_b]^\top$, and $\boldsymbol{\eta} = [\phi,\, \theta,\, \psi]^\top$, the position kinematics are

$$
\begin{aligned}
\dot{x} &= u_b c_\theta c_\psi + v_b \left(s_\phi s_\theta c_\psi - c_\phi s_\psi \right) + w_b \left(c_\phi s_\theta c_\psi + s_\phi s_\psi \right), \\
\dot{y} &= u_b c_\theta s_\psi + v_b \left(s_\phi s_\theta s_\psi + c_\phi c_\psi \right) + w_b \left(c_\phi s_\theta s_\psi - s_\phi c_\psi \right), \\
\dot{z} &= u_b s_\theta - v_b s_\phi c_\theta - w_b c_\phi c_\theta,
\end{aligned}
\tag{F.1}
$$

where $c_\phi = \cos\phi$, $s_\phi = \sin\phi$, and similarly for θ and ψ. The corresponding Euler-angle rates are

$$
\begin{aligned}
\dot{\phi} &= p_b + q_b s_\phi \tan\theta + r_b c_\phi \tan\theta, \\
\dot{\theta} &= q_b c_\phi - r_b s_\phi, \\
\dot{\psi} &= \frac{q_b s_\phi + r_b c_\phi}{c_\theta}.
\end{aligned}
\tag{F.2}
$$

Let X_b, Y_b, and Z_b be the total body-axis forces, including aerodynamic and gravitational terms, and let L_b, M_b, and N_b be the total aerodynamic moments about the centre of gravity. The translational dynamics are then

$$
\begin{aligned}
\dot{u}_b &= \frac{X_b}{m} + r_b v_b - q_b w_b, \\
\dot{v}_b &= \frac{Y_b}{m} + p_b w_b - r_b u_b, \\
\dot{w}_b &= \frac{Z_b}{m} + q_b u_b - p_b v_b.
\end{aligned}
\tag{F.3}
$$

Equivalently, if only the aerodynamic force components X_{aero}, Y_{aero}, and Z_{aero} are listed, the gravity terms are added as $-mg\sin\theta$, $mg\sin\phi\cos\theta$, and $mg\cos\phi\cos\theta$, respectively.

The manufactured glider inertia has the retained product of inertia I_{xz} and negligible I_{xy} and I_{yz}. Defining

$$
\begin{aligned}
R_L &= L_b - I_{xz} p_b q_b + \left(I_{yy} - I_{zz} \right) q_b r_b, \\
R_M &= M_b + \left(I_{zz} - I_{xx} \right) p_b r_b + I_{xz} \left(p_b^2 - r_b^2 \right), \\
R_N &= N_b + \left(I_{xx} - I_{yy} \right) p_b q_b + I_{xz} q_b r_b, \\
\Delta_I &= I_{xx} I_{zz} - I_{xz}^2,
\end{aligned}
\tag{F.4}
$$

the rotational accelerations are

$$
\begin{aligned}
\dot{p}_b &= \frac{I_{zz} R_L - I_{xz} R_N}{\Delta_I}, \\
\dot{q}_b &= \frac{R_M}{I_{yy}}, \\
\dot{r}_b &= \frac{-I_{xz} R_L + I_{xx} R_N}{\Delta_I}.
\end{aligned}
\tag{F.5}
$$

The three realised control-surface states follow the first-order timing model described in Section B.3.3:

$$\dot{\delta}_a = \frac{\delta_a^{\text{cmd}} - \delta_a}{\tau_{\delta,a}}, \qquad \dot{\delta}_e = \frac{\delta_e^{\text{cmd}} - \delta_e}{\tau_{\delta,e}}, \qquad \dot{\delta}_r = \frac{\delta_r^{\text{cmd}} - \delta_r}{\tau_{\delta,r}}. \tag{F.6}$$

F.1.1 Mass, Geometry, and Panelwise Aerodynamic Model

The first group of tables defines the rigid-body and strip geometry used by the scalar model above. The centre of gravity fixes the origin of the body frame, the measured mass and inertia define the rigid-body response, and the lifting surfaces are then discretised into local aerodynamic strips.

Table F.1 Mass and reference geometry.

Quantity	Symbol	Unit	Value
Mass	m	kg	0.14856
CG from wing leading edge	x_{CG}	m	0.10550
CG from fuselage rod	z_{CG}	m	−0.008037
Reference area	S_{ref}	m^2	0.12606
Reference span	b_{ref}	m	0.764
Reference chord	c_{ref}	m	0.165
Fuselage drag area	$S_{\text{drag,fuse}}$	m^2	2.4488×10^{-4}

Table F.2 Body-axis inertia.

Quantity	Symbol	Unit	Value
Roll inertia	I_{xx}	kg m^2	2.7027×10^{-3}
Pitch inertia	I_{yy}	kg m^2	2.9876×10^{-3}
Yaw inertia	I_{zz}	kg m^2	5.6062×10^{-3}
Product of inertia	I_{xz}	kg m^2	2.3151×10^{-5}

For a horizontal lifting surface, the strip centre is placed at a spanwise distance s from the centreline. On the starboard side $r_{y,b} = s\cos\Gamma$, whereas on the port side $r_{y,b} = -s\cos\Gamma$. The body-axis vertical coordinate is $r_{z,b} = z_{\text{LE},b} - s\sin\Gamma$, and the chordwise centre is evaluated at the quarter chord as $r_{x,b} = x_{\text{LE},b} - c_i/4$.

For the vertical tail, the strips are stacked in the body z_b direction. Given that, $r_{y,b} = 0$ and $r_{z,b} = z_{\text{LE},b} - s$.

Table F.3 Lifting-surface geometry and strip discretisation. The location vector is the root leading edge position in body axes, relative to the CG; S_i is the area of each generated strip.

Lifting surface	Strips	c_i	Span	S_i	$\mathbf{r}_{\text{LE},b}$	Dihedral
Wing	12	0.165 m	0.764 m	0.0105 m^2	(0.10550, 0, −0.00346) m	4.64°
Horizontal tail	8	0.091 m	0.364 m	0.00414 m^2	(−0.29775, 0, 0.01404) m	0°
Vertical tail	4	0.059 m	0.119 m	0.00176 m^2	(−0.31275, 0, 0.00304) m	0°

The lifting-surface aerodynamic and control parameters in Table F.4 are used by a local two-dimensional section model.

Table F.4 Lifting surfaces aerodynamic and control parameters.

Lifting surface	$C_{D0,i}$	e_i	$\alpha_{0,i}$	Control surface
Wing	0.054	0.2542	0.057°	Aileron, 0.30c, $\eta = 0.30 - 0.85$
Horizontal tail	0.060	0.2418	−3.000°	Elevator, 0.30c, $\eta = 0 - 1$
Vertical tail	0.060	0.2325	0°	Rudder, 0.35c, $\eta = 0 - 1$

For strip i, the local relative airflow is first evaluated at the strip centre. If the body-axis wind velocity at that strip is written as $(W_{x,i,b}, W_{y,i,b}, W_{z,i,b})$, then the relative airflow components before removing spanwise flow are

$$\begin{aligned} V_{x,i,b} &= u_b + q_b r_{z,i,b} - r_b r_{y,i,b} - W_{x,i,b}, \\ V_{y,i,b} &= v_b + r_b r_{x,i,b} - p_b r_{z,i,b} - W_{y,i,b}, \\ V_{z,i,b} &= w_b + p_b r_{y,i,b} - q_b r_{x,i,b} - W_{z,i,b}. \end{aligned} \tag{F.7}$$

Only the chord-normal component of this velocity is used for the strip lift and drag. With $\mathbf{s}_{i,b} = (s_{x,i,b},\ s_{y,i,b},\ s_{z,i,b})$, the spanwise velocity is

$$V_{s,i} = V_{x,i,b} s_{x,i,b} + V_{y,i,b} s_{y,i,b} + V_{z,i,b} s_{z,i,b}, \tag{F.8}$$

and the chord-normal components are

$$\begin{aligned} V^{\perp}_{x,i,b} &= V_{x,i,b} - V_{s,i} s_{x,i,b}, \\ V^{\perp}_{y,i,b} &= V_{y,i,b} - V_{s,i} s_{y,i,b}, \\ V^{\perp}_{z,i,b} &= V_{z,i,b} - V_{s,i} s_{z,i,b}. \end{aligned} \tag{F.9}$$

The dynamic pressure and local angle of attack are then

$$\begin{aligned} q_i &= \frac{1}{2}\rho \left[\left(V^{\perp}_{x,i,b}\right)^2 + \left(V^{\perp}_{y,i,b}\right)^2 + \left(V^{\perp}_{z,i,b}\right)^2 \right], \\ \alpha_i &= \operatorname{atan2}\left(-\left(V^{\perp}_{x,i,b} n_{x,i,b} + V^{\perp}_{y,i,b} n_{y,i,b} + V^{\perp}_{z,i,b} n_{z,i,b} \right), V^{\perp}_{x,i,b} \right). \end{aligned} \tag{F.10}$$

The local control incidence is obtained from the realised aileron, elevator, and rudder deflections, not from the raw command. With the control effectiveness scales γ_a, γ_e, and γ_r selected by the current angle of attack regime, the local incidence increment is

$$\delta_i = \gamma_a m_{i,a} \delta_a + \gamma_e m_{i,e} \delta_e + \gamma_r m_{i,r} \delta_r, \qquad \alpha_{i,\text{eff}} = \alpha_i - \alpha_{0,i} + \lambda_i \delta_i. \tag{F.11}$$

The finite aspect ratio lift slope, attached flow coefficients, and post-stall flat-plate coefficients are

$$\begin{aligned} a_i &= \frac{2\pi Æ_i}{Æ_i + 2}, \\ C^{\text{att}}_{L,i} &= a_i \alpha_{i,\text{eff}}, \\ C^{\text{att}}_{D,i} &= C_{D0,i} + \frac{\left(C^{\text{att}}_{L,i}\right)^2}{\pi e_i Æ_i}, \\ C^{\text{post}}_{L,i} &= 2 \sin \alpha_{i,\text{eff}} \cos \alpha_{i,\text{eff}}, \\ C^{\text{post}}_{D,i} &= C_{D0,i} + k_D \sin^2 \alpha_{i,\text{eff}}. \end{aligned} \tag{F.12}$$

The stall activation σ_i blends attached and post-stall behaviour:

$$C_{L,i} = (1-\sigma_i) C^{\text{att}}_{L,i} + \sigma_i C^{\text{post}}_{L,i}, \qquad C_{D,i} = (1-\sigma_i) C^{\text{att}}_{D,i} + \sigma_i C^{\text{post}}_{D,i}. \tag{F.13}$$

The generated strip tables list the numerical values that feed the preceding calculation. Once $C_{L,i}$ and $C_{D,i}$ are known, the body-axis force from strip i is evaluated component by component. Taking $\boldsymbol{\ell}_{i,b} = (\ell_{x,i,b},\ \ell_{y,i,b},\ \ell_{z,i,b})$ and defining $\mathbf{d}_{i,b} = (d_{x,i,b},\ d_{y,i,b},\ d_{z,i,b})$ in the drag force direction,

$$\begin{aligned} X_{i,b} &= q_i S_i \left(C_{L,i} \ell_{x,i,b} + C_{D,i} d_{x,i,b} \right), \\ Y_{i,b} &= q_i S_i \left(C_{L,i} \ell_{y,i,b} + C_{D,i} d_{y,i,b} \right), \\ Z_{i,b} &= q_i S_i \left(C_{L,i} \ell_{z,i,b} + C_{D,i} d_{z,i,b} \right). \end{aligned} \tag{F.14}$$

The moment of the same strip about the centre of gravity is

$$\begin{aligned} L_{i,b} &= r_{y,i,b} Z_{i,b} - r_{z,i,b} Y_{i,b}, \\ M_{i,b} &= r_{z,i,b} X_{i,b} - r_{x,i,b} Z_{i,b}, \\ N_{i,b} &= r_{x,i,b} Y_{i,b} - r_{y,i,b} X_{i,b}. \end{aligned} \tag{F.15}$$

Summing (F.14)–(F.15) over the 24 strips gives the baseline aerodynamic force and moment.

Table F.5 Generated strip geometry and scalar aerodynamic parameters. Strip positions are body-axis centre locations relative to the CG.

i	**Surface**	$r_{x,b}$	$r_{y,b}$	$r_{z,b}$	$C_{D0,i}$	$\alpha_{0,i}$	e_i
1	Wing	0.0643	0.0317	−0.00604	0.054	0.057	0.254
2	Wing	0.0643	0.0952	−0.0112	0.054	0.057	0.254
3	Wing	0.0643	0.159	−0.0163	0.054	0.057	0.254
4	Wing	0.0643	0.222	−0.0215	0.054	0.057	0.254
5	Wing	0.0643	0.286	−0.0266	0.054	0.057	0.254
6	Wing	0.0643	0.349	−0.0318	0.054	0.057	0.254
7	Wing	0.0643	−0.0317	−0.00604	0.054	0.057	0.254
8	Wing	0.0643	−0.0952	−0.0112	0.054	0.057	0.254
9	Wing	0.0643	−0.159	−0.0163	0.054	0.057	0.254
10	Wing	0.0643	−0.222	−0.0215	0.054	0.057	0.254
11	Wing	0.0643	−0.286	−0.0266	0.054	0.057	0.254
12	Wing	0.0643	−0.349	−0.0318	0.054	0.057	0.254
13	Horizontal tail	−0.321	0.0228	0.0140	0.060	−3.00	0.242
14	Horizontal tail	−0.321	0.0683	0.0140	0.060	−3.00	0.242
15	Horizontal tail	−0.321	0.114	0.0140	0.060	−3.00	0.242
16	Horizontal tail	−0.321	0.159	0.0140	0.060	−3.00	0.242
17	Horizontal tail	−0.321	−0.0228	0.0140	0.060	−3.00	0.242
18	Horizontal tail	−0.321	−0.0683	0.0140	0.060	−3.00	0.242
19	Horizontal tail	−0.321	−0.114	0.0140	0.060	−3.00	0.242
20	Horizontal tail	−0.321	−0.159	0.0140	0.060	−3.00	0.242
21	Vertical tail	−0.328	0	−0.0118	0.060	0	0.233
22	Vertical tail	−0.328	0	−0.0416	0.060	0	0.233
23	Vertical tail	−0.328	0	−0.0713	0.060	0	0.233
24	Vertical tail	−0.328	0	−0.101	0.060	0	0.233

Table F.6 Generated strip orientation and control surface parameters. The control effectiveness vector is ordered as aileron, elevator, rudder.

i	**Lifting surface**	$\mathbf{s}_{i,b}$	$\mathbf{n}_{i,b}$	$\mathbf{m}_i$	**Flap scale**
1	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(0, 0, 0)	0
2	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(0, 0, 0)	0
3	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(1, 0, 0)	0.661
4	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(1, 0, 0)	0.661
5	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(1, 0, 0)	0.661
6	Wing	(0, 0.997, −0.081)	(0, −0.081, −0.997)	(0, 0, 0)	0
7	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(0, 0, 0)	0
8	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(0, 0, 0)	0
9	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(−1, 0, 0)	0.661
10	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(−1, 0, 0)	0.661
11	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(−1, 0, 0)	0.661
12	Wing	(0, −0.997, −0.081)	(0, 0.081, −0.997)	(0, 0, 0)	0
13	Horizontal tail	(0, 1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
14	Horizontal tail	(0, 1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
15	Horizontal tail	(0, 1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
16	Horizontal tail	(0, 1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
17	Horizontal tail	(0, −1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
18	Horizontal tail	(0, −1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
19	Horizontal tail	(0, −1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
20	Horizontal tail	(0, −1, 0)	(0, 0, −1)	(0, −1, 0)	0.661
21	Vertical tail	(0, 0, −1)	(0, −1, 0)	(0, 0, 1)	0.707
22	Vertical tail	(0, 0, −1)	(0, −1, 0)	(0, 0, 1)	0.707
23	Vertical tail	(0, 0, −1)	(0, −1, 0)	(0, 0, 1)	0.707
24	Vertical tail	(0, 0, −1)	(0, −1, 0)	(0, 0, 1)	0.707

F.1.2 System Identification Procedure

For each accepted flight record s, the measured state at launch gate defines the replay initial condition. The simulation is propagated with the scalar dynamics in (F.1)–(F.6) and the strip forces in (F.14). At the comparison time, the replay error is recorded in the physical coordinates used for flight assessment:

$$\begin{aligned} e_{x,s} &= x_s^{\text{sim}} - x_s^{\text{flight}}, \quad e_{y,s} = y_s^{\text{sim}} - y_s^{\text{flight}}, \quad e_{z,s} = z_s^{\text{sim}} - z_s^{\text{flight}}, \\ e_{\phi,s} &= \phi_s^{\text{sim}} - \phi_s^{\text{flight}}, \quad e_{\theta,s} = \theta_s^{\text{sim}} - \theta_s^{\text{flight}}, \quad e_{\psi,s} = \psi_s^{\text{sim}} - \psi_s^{\text{flight}}. \end{aligned} \tag{F.16}$$

The selected parameter set is the local Pareto candidate that improves or preserves the held-out longitudinal replay while reducing the relevant lateral-directional error. This conservative selection rule is important because the final controller uses the model for local feedback design, where excessive control surface authority or over-fitted post-stall damping would produce unsafe optimism.

Algorithm F.1 Neutral still air residual calibration and Pareto selection.

Input: Neutral launch records $\mathcal{D}_0$, $\mathcal{X}_{\text{launch}}$, baseline $\boldsymbol{\theta}_{\text{aero},0}$, residual family (6.17), objective $\mathcal{J}$ (6.18), and held-out seed.

Output: Accepted residual parameter $\boldsymbol{\theta}^\star_{\text{aero}}$ and replay audit.

Load accepted neutral open-loop records and processed states $\mathbf{x}_{s,k}$
for *each record* $s \in \mathcal{D}_0$ **do**
  Interpolate the launch state, check $\mathbf{x}_{s,0} \in \mathcal{X}_{\text{launch}}$, and align replay start to the launch-handoff window
  Reject s if the aligned start fails the replay-start envelope; otherwise compute its lateral launch-confidence weight
Split kept records into $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{hold}}$ by session-stratified sampling
Initialise the scalar residual force and moment coefficients in (F.18)–(F.21) from $\boldsymbol{\theta}_{\text{aero},0}$
for *each staged candidate* $\boldsymbol{\theta}^{(j)}_{\text{aero}}$ **do**
  Replay (6.5)–(6.9) on $\mathcal{D}_{\text{train}}$ and evaluate $\mathcal{J}(\boldsymbol{\theta}^{(j)}_{\text{aero}})$
  Keep $\boldsymbol{\theta}^{(j)}_{\text{aero}}$ only if held-out longitudinal replay is improved or preserved
Generate joint candidates by adding selected lateral and coupling terms to accepted longitudinal proposals
for *each joint candidate* **do**
  Replay $\mathcal{D}_{\text{hold}}$ and record $(\Delta x, \Delta y, \Delta z_{\text{loss}}, \text{sink}, \phi, \theta, \psi)$
Reject dominated candidates and select the conservative local Pareto candidate that preserves longitudinal replay while reducing relevant lateral error
return $\boldsymbol{\theta}^\star_{\text{aero}}$ and audit tables

---

Algorithm F.2 Commanded control surface effectiveness calibration.

Input: Commanded launch records $\mathcal{D}_\delta$, $\boldsymbol{\theta}^\star_{\text{aero}}$, surface model (6.7), incidence term (6.11), and candidate schedules for $\boldsymbol{\theta}_\delta$ in (6.17).

Output: Conservative surface-effectiveness schedule $\boldsymbol{\theta}^\star_\delta$.

Load aileron, elevator, and rudder pulse-launch records with logged aggregate commands
Keep command conversion, servo signs, actuator timing, and $\boldsymbol{\theta}^\star_{\text{aero}}$ fixed
Assign launch-level train/held-out splits within each surface and command-magnitude ladder
for *each candidate schedule* $\boldsymbol{\theta}^{(j)}_\delta$ **do**
  Modify only the aerodynamic effectiveness scales γ_a, γ_e, and γ_r in the scalar incidence mapping (F.11)
  Propagate $\boldsymbol{\delta}$ with (6.7) and replay (6.5)–(6.9)
  Reject $\boldsymbol{\theta}^{(j)}_\delta$ if held-out trajectory replay degrades or conservative schedule bounds are violated
Freeze the normal, transition, and post-stall effectiveness schedule for controller development
return $\boldsymbol{\theta}^\star_\delta$ in Table F.9

F.1.3 Calibration and Residual Parameters

Table F.7 Still air calibration metadata.

Quantity	**Value**
Calibration status	Active
Calibration class	Neutral still air residual replay alignment
Throw records available	105
Filtered throw records used	9
Held-out throw records	14
Held-out policy	Randomised, stratified by session label
Held-out seed	606
Alignment window	0.040 s
Parameter count	4 total, 2 lateral

Table F.8 Held-out replay metrics for the accepted calibration.

Metric	**Unit**	**Value**
Held-out score	-	9.1195
Longitudinal score	-	6.9168
Lateral score	-	2.2027
Δx MAE	m	0.2988
Δy MAE	m	0.5866
Altitude-loss MAE	m	0.3260
Sink MAE	$\mathrm{m\,s^{-1}}$	0.2814
Roll MAE	deg	8.1588
Pitch MAE	deg	3.8974
Yaw MAE	deg	3.9455

Table F.9 Scalar aerodynamic calibration parameters.

Quantity	**Value**
C_{D0} strip scale	3.0000
Fuselage drag-area scale	5.0000
Strip efficiency scale	0.3100
C_m^{att}	0.11310
C_m^{tr}	0.05712
Global C_m bias	0
Post-stall lift residual	0
Post-stall drag residual	0
C_m^{post}	0.07586
C_{mq}^{post}	4.0000

Table F.10 Post-stall residual blend and surface schedule parameters.

Quantity	**Value**
Residual blend start	14.0 deg
Residual blend full	18.0 deg
RBF centres	(20, 45, 70) deg
RBF width	15.0 deg
Neutral surface trim	(0, 0, 0) rad
Normal effectiveness	(0.85, 0.75, 0.85)
Transition effectiveness	(0.55, 0.55, 0.55)
Post-stall effectiveness	(0.45, 0.45, 0.40)

Table F.11 Attached and transition lateral-directional residual coefficients.

Regime	**Component**	**Bias**	β	$\hat{p}$	$\hat{r}$
Attached	Side force	0	−1.98	0	0
	Roll moment		0		
	Yaw moment		−0.0343		
Transition	Side force	0	0	0	−3.00
	Roll moment		0	0	0
	Yaw moment		−0.0147	−0.146	0

Table F.12 Post-stall RBF residual coefficient vectors at $\alpha = (20,\ 45,\ 70)\,\mathrm{deg}$.

Component	**Term**	**Coefficient vector**
Lift/Drag Pitch moment/Pitch damping	Direct residual	All zero
Side force	Bias, β, $\hat{p}$, $\hat{r}$	All zero
Roll moment	Bias, β, $\hat{p}$	All zero
	$\hat{r}$	$(-0.460,\ 0,\ 0)$
Yaw moment	Bias, $\hat{r}$	All zero
	β	$(-0.0187,\ 0,\ 0)$
	$\hat{p}$	$(-0.0712,\ 0,\ 0)$

The calibration parameters above enter the force and moment calculation as nondimensional residual coefficients. Let V be the airspeed at CG, $\bar{q} = \frac{1}{2}\rho V^2$, and

$$\hat{p} = \frac{p_b b_{\mathrm{ref}}}{2V}, \qquad \hat{q} = \frac{q_b c_{\mathrm{ref}}}{2V}, \qquad \hat{r} = \frac{r_b b_{\mathrm{ref}}}{2V}. \tag{F.17}$$

The attached-flow lateral-directional residual coefficients have the textbook derivative form [33]

$$\begin{aligned} C_Y^{\mathrm{att}} &= C_{Y,0}^{\mathrm{att}} + C_{Y,\beta}^{\mathrm{att}}\beta + C_{Y,\hat{p}}^{\mathrm{att}}\hat{p} + C_{Y,\hat{r}}^{\mathrm{att}}\hat{r}, \\ C_l^{\mathrm{att}} &= C_{l,0}^{\mathrm{att}} + C_{l,\beta}^{\mathrm{att}}\beta + C_{l,\hat{p}}^{\mathrm{att}}\hat{p} + C_{l,\hat{r}}^{\mathrm{att}}\hat{r}, \\ C_n^{\mathrm{att}} &= C_{n,0}^{\mathrm{att}} + C_{n,\beta}^{\mathrm{att}}\beta + C_{n,\hat{p}}^{\mathrm{att}}\hat{p} + C_{n,\hat{r}}^{\mathrm{att}}\hat{r}. \end{aligned} \tag{F.18}$$

The transition terms use the same derivative form, but are multiplied by the transition weight generated from the post-stall activation. The pitch residual is separated into attached, transition, and post-stall:

$$C_m^{\mathrm{res}} = C_m^0 + W_{\mathrm{att}}C_m^{\mathrm{att}} + W_{\mathrm{tr}}C_m^{\mathrm{tr}} + W_{\mathrm{post}}\left(C_m^{\mathrm{post}} + C_{mq}^{\mathrm{post}}\hat{q}\right) + C_m^{\mathrm{RBF}} + C_{mq}^{\mathrm{RBF}}\hat{q}. \tag{F.19}$$

The post-stall lift and drag residuals are applied in the chord-normal plane at CG. If $\boldsymbol{\ell}_{CG,b}$ and $\mathbf{d}_{CG,b}$ are the corresponding lift and drag-force directions at the centre of gravity, their scalar contribution is

$$\begin{aligned} X_{\mathrm{LD,res}} &= \bar{q}S_{\mathrm{ref}}\left(C_L^{\mathrm{res}}\ell_{x,CG,b} + C_D^{\mathrm{res}}d_{x,CG,b}\right), \\ Z_{\mathrm{LD,res}} &= \bar{q}S_{\mathrm{ref}}\left(C_L^{\mathrm{res}}\ell_{z,CG,b} + C_D^{\mathrm{res}}d_{z,CG,b}\right). \end{aligned} \tag{F.20}$$

The active coefficients for these two force residuals are zero in Table F.9, but the calculation is retained so that the model form is explicit. The remaining residual loads are added to the strip sums as

$$\begin{aligned} X_{\mathrm{res}} &= X_{\mathrm{LD,res}}, \\ Y_{\mathrm{res}} &= \bar{q}S_{\mathrm{ref}}\left(C_Y^{\mathrm{att}} + W_{\mathrm{tr}}C_Y^{\mathrm{tr}} + C_Y^{\mathrm{RBF}}\right), \\ L_{\mathrm{res}} &= \bar{q}S_{\mathrm{ref}}b_{\mathrm{ref}}\left(C_l^{\mathrm{att}} + W_{\mathrm{tr}}C_l^{\mathrm{tr}} + C_l^{\mathrm{RBF}}\right), \\ M_{\mathrm{res}} &= \bar{q}S_{\mathrm{ref}}c_{\mathrm{ref}}C_m^{\mathrm{res}}, \\ N_{\mathrm{res}} &= \bar{q}S_{\mathrm{ref}}b_{\mathrm{ref}}\left(C_n^{\mathrm{att}} + W_{\mathrm{tr}}C_n^{\mathrm{tr}} + C_n^{\mathrm{RBF}}\right), \\ Z_{\mathrm{res}} &= Z_{\mathrm{LD,res}}. \end{aligned} \tag{F.21}$$

The post-stall RBF entries in Table F.12 are evaluated at the angle of attack at CG using

$$\mathcal{R}_h(\alpha) = \exp\left[-\frac{1}{2}\left(\frac{\alpha - \alpha_h}{\Delta\alpha_{\mathrm{RBF}}}\right)^2\right], \qquad \alpha_h \in \{20^\circ,\ 45^\circ,\ 70^\circ\}. \tag{F.22}$$

For example, a listed post-stall yaw coefficient vector $(c_{20},\ c_{45},\ c_{70})$ contributes $c_{20}\mathcal{R}_{20}(\alpha) + c_{45}\mathcal{R}_{45}(\alpha) + c_{70}\mathcal{R}_{70}(\alpha)$ before being multiplied by the appropriate force or moment scale.

F.2 Randomised Simulation Settings

F.2.1 Environment Factors and Updraft Settings

Table F.13 Factors used to define one finite-horizon simulated flight.

Factor	Meaning	Source
$w_{\mathbf{e}}$	Vertical-flow field	Still air, fixed fan surrogate, or randomised fan surrogate
$\boldsymbol{\theta}_p$	Glider properties	Mass, CG, inertia, aerodynamic scale, and control effectiveness multipliers
$\boldsymbol{\theta}_\tau$	Timing parameters	State-feedback delay, command delay, actuator lag, and timing jitter
$\boldsymbol{\theta}_\delta$	Surface implementation	Neutral bias, limit scale, and aileron asymmetry
$\mathbf{x}_0$	Initial state	Launch, in-flight, boundary, or recovery sample

Table F.14 Initial state family allocation used by the sampler.

Start state family	Weight	Slots per 20
Launch gate	0.40	8
In-flight nominal	0.25	5
In-flight lift region	0.15	3
Boundary-near	0.10	2
Recovery-edge	0.10	2

Table F.15 Fan and updraft settings used by the simulation environments.

Quantity	Symbol or policy	Value
Still air field	$w_{\mathbf{e}}(\mathbf{p})$	0
Single-fan base centre	$\mathbf{c}_1^0$	$(4.2,\ 2.4)\,\mathrm{m}$
Four-fan base centres	$\mathbf{c}_f^0$	$(3.0,\ 3.6),\ (5.4,\ 3.6),\ (3.0,\ 1.2),\ (5.4,\ 1.2)\,\mathrm{m}$
Local fan-layout shift	$\Delta\mathbf{c}$	$\mathcal{U}([-0.20,\ 0.20]^2)\,\mathrm{m}$
Arena-wide fan centre	$\mathbf{c}_f$	$\mathcal{U}([0,\ 8]\times[0,\ 4.8])\,\mathrm{m}$
Arena-wide separation	$d_{\min}$	$0.5\,\mathrm{m}$
Position rejection attempts	–	10000
Per-fan strength scale	a_f	$\mathcal{U}(0.85,\ 1.15)$
Global amplitude scale	–	1.0
Width scale	κ_f	$\mathcal{U}(0.85,\ 1.15)$
Global centre shift	–	$(0,\ 0)\,\mathrm{m}$
Local uncertainty scale	κ_σ	$\mathcal{U}(1.0,\ 2.5)$
Default active-fan mask	μ_f	At least one active fan
Scheduled active-fan counts	N_{active}	0, 1, 2, 3, 4
Residual field	–	Single-layer annular-GP randomisation without duplicate strength or centre shift

F.2.2 Plant and Implementation Randomisation

Table F.16 Glider model randomisation ranges.

Quantity	Fixed and nominal cases	Randomised cases
Mass scale	1.0	$\mathcal{U}(0.95,\ 1.05)$
CG offset	(0, 0, 0)	$\mathcal{U}([-0.01,\ 0.01]^3)$ m
Common and principal inertia scales	1.0	$\mathcal{U}(0.92,\ 1.08)$
Cross-inertia perturbation	None	None
Aerodynamic coefficient scale	1.0	$\mathcal{U}(0.95,\ 1.05)$
Surface-calibration scale	1.0	1.0
Global control-effectiveness multiplier	1.0	$\mathcal{U}(0.75,\ 1.25)$
Aileron, elevator, and rudder effectiveness multipliers	1.0	$\mathcal{U}(0.85,\ 1.15)$

Table F.17 Timing and surface implementation randomisation ranges.

Quantity	Diagnostic cases	Nominal transfer cases	Randomised transfer case
Latency case	None or selected nominal	Nominal or conservative	Nominal or conservative, sampled with equal probability
State-feedback delay scale	1.0	1.0	$\mathcal{U}(0.90,\ 1.20)$
Command-onset delay scale	1.0	1.0	$\mathcal{U}(0.90,\ 1.25)$
Command-transport delay scale	1.0	1.0	$\mathcal{U}(0.90,\ 1.25)$
Latency jitter	0	0	$\mathcal{U}(0,\ 0.015)$ s
Actuator time-constant scale	1.0	1.05	$\mathcal{U}(0.90,\ 1.25)$
Surface neutral bias	$(0, 0, 0)$	$(0, 0, 0)$	$\mathcal{U}([-0.015,\ 0.015]^3)$ rad
Surface-limit scale	1.0	1.0	$\mathcal{U}(0.90,\ 1.00)$
Left-right aileron asymmetry	1.0	1.0	$\mathcal{U}(0.95,\ 1.05)$
Command quantisation flag	None	None	Fixed 20% lattice

F.3 Controller Algorithm Specification

Algorithm F.3 Offline construction of validated manoeuvre primitives.

Input: $(\chi, \mathcal{C})$, $\mathcal{P}$, $\{\mathcal{X}_c\}_{c\in\mathcal{C}}$ in Table F.18, $p_{\mathcal{E}}$, f_Δ, T, $\mathcal{U}$, N_{gen}, N_{val}, and Θ_{acc}.
Output: Validated primitive set $\mathcal{V}$ with evidence records.

$\mathcal{J} \leftarrow \emptyset$, $\mathcal{V} \leftarrow \emptyset$
for $c \in \mathcal{C}$, $\pi \in \mathcal{P}$, $n = 1, \ldots, N_{\text{gen}}$ **do**
  $x_0 \sim \mathcal{X}_c$, $\mathcal{E} \sim p_{\mathcal{E}}$
  $r_j \leftarrow \mathsf{Reference}(\pi, c, x_0, \mathcal{E})$
  $(K_j, \text{ok}) \leftarrow \mathsf{SynthesiseLQR}(f_\Delta, r_j, T)$
  if ok **then**
    $\tau_j \leftarrow \mathsf{Rollout}(f_\Delta, x_0, \mathcal{E}, r_j, K_j, T)$
    $\eta_j \leftarrow \mathsf{Metrics}(\tau_j, \chi)$
    if $\mathsf{Retain}(\tau_j, \eta_j, \mathcal{J}, \Theta_{\text{acc}})$ **then**
      $\mathcal{J} \leftarrow \mathcal{J} \cup \{(j, \pi, c, r_j, K_j, T)\}$
for $(j, u) \in \mathsf{SpeedExpand}(\mathcal{J}, \mathcal{U})$ **do**
  $\mathcal{D}_{j,u} \leftarrow \{\mathsf{Rollout}(f_\Delta, x_m, \mathcal{E}_m, r_j, K_j, T, u)\}_{m=1}^{N_{\text{val}}}$, where $x_m \sim \mathcal{X}_{c_j}$ and $\mathcal{E}_m \sim p_{\mathcal{E}}$
  $E_{j,u}(c_j) \leftarrow \mathsf{Evidence}(\mathcal{D}_{j,u})$
  if $\mathsf{Accept}(E_{j,u}(c_j), \Theta_{\text{acc}})$ **then**
    $\mathcal{V} \leftarrow \mathcal{V} \cup \{\mathsf{Primitive}(j, u, r_j, K_j, T, E_{j,u})\}$
return $\mathcal{V}$

Algorithm F.4 Online viability governor primitive selection.

Input: $\hat{x}_k$, $\mathcal{L}_B$, Θ_g, $\mathcal{M}_k$, active latency model Λ_k, and mission target.
Output: $j_k^\star$, $\mu_{k:k+N-1}$, $\mathcal{M}_{k+1}$, and no-viable flag b_{nv}.

$c_k \leftarrow \chi(\hat{x}_k)$, $b_{\text{nv}} \leftarrow$ false
if $c_k \in \{\text{terminal}, \text{failure}\}$ **then**
  return $\emptyset$, $\mathsf{TerminalCommand}(\hat{x}_k)$, $\mathcal{M}_k$, b_{nv}
$\mathcal{A}_k \leftarrow \emptyset$
for $j \in \{i \in \mathcal{L}_B : c_i^{\text{in}} = c_k\}$ **do**
  $\hat{\tau}_{j,k} \leftarrow \mathsf{Predict}(\hat{x}_k, j, \Lambda_k)$, $\zeta_{j,k} \leftarrow \mathsf{MemoryQuery}(\mathcal{M}_k, \hat{\tau}_{j,k})$
  if $\mathsf{Admissible}(j, \hat{x}_k, \hat{\tau}_{j,k}, \Lambda_k, \Theta_g)$ **then**
    $S_{j,k}^0 \leftarrow \mathsf{Score}(j, \hat{x}_k, \hat{\tau}_{j,k}, 0, \Theta_g)$, $S_{j,k} \leftarrow \mathsf{Score}(j, \hat{x}_k, \hat{\tau}_{j,k}, \zeta_{j,k}, \Theta_g)$
    $\mathcal{A}_k \leftarrow \mathcal{A}_k \cup \{j\}$
if $\mathcal{A}_k = \emptyset$ **then**
  $b_{\text{nv}} \leftarrow$ true
  return $\emptyset$, $\mathsf{RecoveryCommand}(\hat{x}_k)$, $\mathcal{M}_k$, b_{nv}
$j_k^0 \leftarrow \arg\max_{j\in\mathcal{A}_k} S_{j,k}^0$
$\mathcal{A}_k^{\text{sh}} \leftarrow \{j \in \mathcal{A}_k : \mathsf{Shield}(j, j_k^0, \hat{x}_k, \hat{\tau}_{j,k}, \zeta_{j,k}, \Theta_g)\}$
$j_k^\star \leftarrow j_k^0$ if $\mathcal{A}_k^{\text{sh}} = \emptyset$; otherwise $j_k^\star \leftarrow \arg\max_{j\in\mathcal{A}_k^{\text{sh}}} S_{j,k}$
$\mu_{k:k+N-1} \leftarrow \mathsf{Track}(\hat{x}_k, j_k^\star, \Lambda_k)$, $\tau_k^{\text{obs}} \leftarrow \mathsf{Execute}(\mu_{k:k+N-1})$
$\mathcal{M}_{k+1} \leftarrow \mathsf{UpdateMemory}(\mathcal{M}_k, \tau_k^{\text{obs}}, \hat{\tau}_{j_k^\star,k})$
return $j_k^\star$, $\mu_{k:k+N-1}$, $\mathcal{M}_{k+1}$, b_{nv}

F.4 Validation Ladder Setup

Table F.18 Entry classes sampling ranges.

Manoeuvre family	Position	Attitude	Body velocity	Body rate	Control surface state
Launch	$x[1.2, 1.4]$ m, $y[1.8, 2.2]$ m, $z[1.3, 1.8]$ m	$\phi \pm 20°$, $\theta[-10, 20]°$, $\psi \pm 20°$	$u_b[4, 8]\,\mathrm{m\,s^{-1}}$, $v_b \pm 1.5\,\mathrm{m\,s^{-1}}$, $w_b \pm 0.9\,\mathrm{m\,s^{-1}}$	$p_b \pm 1.2\,\mathrm{rad\,s^{-1}}$, $q_b \pm 1.2\,\mathrm{rad\,s^{-1}}$, $r_b \pm 1.8\,\mathrm{rad\,s^{-1}}$	0
Stable	$x[1.6, 5.8]$ m, $y[0.5, 3.9]$ m, $z[0.9, 2.8]$ m	$\phi \pm 18°$, $\theta \pm 10°$, $\psi \pm 25°$	$u_b[3.0, 8.2]\,\mathrm{m\,s^{-1}}$, $v_b \pm 0.35\,\mathrm{m\,s^{-1}}$, $w_b \pm 0.25\,\mathrm{m\,s^{-1}}$	$p_b \pm 0.35\,\mathrm{rad\,s^{-1}}$, $q_b \pm 0.245\,\mathrm{rad\,s^{-1}}$, $r_b \pm 0.28\,\mathrm{rad\,s^{-1}}$	$\delta \pm 0.22$ rad
Stable, updraft region	$x[2.0, 4.3]$ m, $y[1.2, 3.2]$ m, $z[1.2, 2.4]$ m	$\phi \pm 22°$, $\theta \pm 12°$, $\psi \pm 35°$	$u_b[3.2, 8.0]\,\mathrm{m\,s^{-1}}$, $v_b \pm 0.30\,\mathrm{m\,s^{-1}}$, $w_b \pm 0.22\,\mathrm{m\,s^{-1}}$	$p_b \pm 0.30\,\mathrm{rad\,s^{-1}}$, $q_b \pm 0.21\,\mathrm{rad\,s^{-1}}$, $r_b \pm 0.24\,\mathrm{rad\,s^{-1}}$	$\delta \pm 0.20$ rad
Boundary	Nominal, with $x[6.35, 6.58]$ m or $y[0.04, 0.20]$ m	Nominal	$u_b[3.0, 8.0]\,\mathrm{m\,s^{-1}}$; otherwise nominal	Nominal	$\delta \pm 0.22$ rad
Recovery	$x[1.6, 5.8]$ m, $y[0.5, 3.9]$ m, $z[0.9, 2.8]$ m	$\phi \pm 42°$, $\theta[-35, 25]°$, $\psi \pm 25°$	$u_b[2.2, 5.2]\,\mathrm{m\,s^{-1}}$, $v_b \pm 1.0\,\mathrm{m\,s^{-1}}$, $w_b[-0.3, 2.0]\,\mathrm{m\,s^{-1}}$	$p_b \pm 1.2\,\mathrm{rad\,s^{-1}}$, $q_b \pm 1.2\,\mathrm{rad\,s^{-1}}$, $r_b \pm 1.0\,\mathrm{rad\,s^{-1}}$	$\delta \pm 0.35$ rad

Algorithm F.5 Validation ladder assembly for controller development.

Input: Validated primitive set $\mathcal{V}$, library tiers $\mathcal{B}_{\text{lib}} = \{\text{no-cluster}, \text{heavy}, \text{balanced}, \text{light}, \text{super-light}\}$, environment distribution $p_{\mathcal{E}}$ in (6.20), start-state ranges in Table F.18, and environment cases in Table 6.1.

Output: Compressed libraries $\mathcal{L}_B$, frozen viability-governor setting, held-out validation schedules, and flight-test handoff.

for *each library tier* $B \in \mathcal{B}_{\text{lib}}$ **do**
  Build $\mathcal{L}_B \subseteq \mathcal{V}$ using the medoid compression in (6.33)
  Use wider representative budgets for launch-entry groups to preserve human-release variation
for *each* $\mathcal{L}_B$ **do**
  Simulate the complete mission in Section 6.1.1 under the full-domain randomised case in Table 6.1
  Tune the viability-governor thresholds, scoring weights, memory-shield margins, and spatial-lift-memory settings in (6.28)–(6.31)
  Select settings that improve the objective in (6.4) while remaining effective across the tested library tiers
Freeze the plant model, selected primitive library, viability-governor settings, latency model, and spatial-memory policy
for *each held-out environment case* **do**
  Sample start states from Table F.18 and environment instances from $p_{\mathcal{E}}$
  Replay the frozen controller over the complete mission without retuning local gains or governor settings
  Record safety, terminal-route, lift-use, memory, timing, and library-tier metrics
Accept the controller for real-flight testing only if the held-out complete-mission replay satisfies the safety and route criteria
return $\mathcal{L}_B$, frozen governor setting, held-out schedules, and flight-test handoff

F.5 Real Flight Runtime Protocol

Algorithm F.6 Real flight launch handoff and controller runtime.

Input: Frozen library $\mathcal{L}_B$, governor parameters Θ_g, latency model Λ, launch gate $\mathcal{G}_{\text{launch}}$, arena bounds $\mathcal{X}_{\text{arena}}$, mission terminal set $\mathcal{X}_{\text{goal}}$, state stream y_k, control period Δt_c, dropout limit N_{drop}, runtime limit $T_{\max}$, and neutral/fail-safe command μ_{safe}.

Output: Flight log $\mathcal{D}_{\text{flight}}$, termination reason ρ, and command history $\mathcal{U}_{\text{cmd}}$.

$\mathcal{D}_{\text{flight}} \leftarrow \emptyset$, $\mathcal{U}_{\text{cmd}} \leftarrow \emptyset$, $\mathcal{M}_0 \leftarrow \emptyset$, $k \leftarrow 0$, $n_{\text{drop}} \leftarrow 0$, $\rho \leftarrow$ none
InitialiseLog(); ArmInterface(); Command(μ_{safe})
while $\rho =$ none **do**
  (y_k, q_k, t_k) $\leftarrow$ ReadTracking()
  if $\neg$Valid(y_k, q_k) **then**
    Command(μ_{safe}); $\rho \leftarrow$ AbortIfTimeout(ρ, t_k); **continue**
  $\hat{x}_k \leftarrow$ InitialiseState(y_k, Λ)
  if LaunchGate($\hat{x}_k, \mathcal{G}_{\text{launch}}$) **then**
    $\mathcal{M}_k \leftarrow \mathcal{M}_0$; $t_0 \leftarrow t_k$; **break**
  Command(μ_{safe})
while $\rho =$ none **do**
  (y_k, q_k, t_k) $\leftarrow$ ReadTracking()
  if Valid(y_k, q_k) **then**
    $n_{\text{drop}} \leftarrow 0$; $\hat{x}_k \leftarrow$ EstimateState(y_k, Λ)
  else
    $n_{\text{drop}} \leftarrow n_{\text{drop}} + 1$; $\hat{x}_k \leftarrow$ PredictState($\hat{x}_{k-1}, \mu_{k-1}, \Lambda$)
  if $n_{\text{drop}} > N_{\text{drop}}$ **then**
    $\rho \leftarrow$ tracking loss; **break**
  if $\hat{x}_k \notin \mathcal{X}_{\text{arena}}$ **then**
    $\rho \leftarrow$ arena boundary; **break**
  if $\hat{x}_k \in \mathcal{X}_{\text{goal}}$ **then**
    $\rho \leftarrow$ mission complete; **break**
  if $t_k - t_0 > T_{\max}$ **then**
    $\rho \leftarrow$ runtime limit; **break**
  ($j_k^\star, \mu_{k:k+N-1}, \mathcal{M}_{k+1}, b_{\text{nv}}$) $\leftarrow$ GovernorSelect($\hat{x}_k, \mathcal{L}_B, \Theta_g, \mathcal{M}_k, \Lambda$)
  $\mu_k \leftarrow$ SaturateRateLimit($\mu_{k:k+N-1}[0]$)
  if $b_{\text{nv}} \wedge$ UnsafeRecovery($\hat{x}_k$) **then**
    $\rho \leftarrow$ no viable primitive; **break**
  Command(μ_k)
  $\mathcal{U}_{\text{cmd}} \leftarrow \mathcal{U}_{\text{cmd}} \cup \{(t_k, \mu_k)\}$
  $\mathcal{D}_{\text{flight}} \leftarrow \mathcal{D}_{\text{flight}} \cup$ Log($t_k, y_k, q_k, \hat{x}_k, j_k^\star, \mu_k, b_{\text{nv}}, \mathcal{M}_{k+1}$)
  WaitUntil($t_k + \Delta t_c$); $k \leftarrow k + 1$
Command(μ_{safe}); DisarmInterface(); CloseLog($\mathcal{D}_{\text{flight}}, \rho$)
return $\mathcal{D}_{\text{flight}}$, ρ, and $\mathcal{U}_{\text{cmd}}$

G Supplementary Controller Validation and Flight Test Results

G.1 Simulation Validation Ladder Result

G.1.1 Supplementary Held-Out Primitive Validation Statistics

Table G.1 Retained held-out primitive validation by manoeuvre family. Distances are in metres; time spent in useful lift is in seconds.

Manoeuvre family	Variants	Replays	Continuation valid	Terminal useful	Hard failure	Wall margin	Mean ΔH	Mean T_ℓ
Glide	200	32000	81.8%	18.1%	0%	−0.07	−0.02	0.04
Safety margin recovery	199	31840	82%	18%	0%	−0.07	−0.02	0.04
Updraft entry	206	32960	82.4%	17.6%	0%	−0.07	−0.02	0.04
Updraft exploitation	193	30880	81.2%	18.8%	0%	−0.07	−0.02	0.04
Mild left turn	207	33120	82.5%	17.5%	0%	−0.07	−0.02	0.04
Mild right turn	193	30880	81.2%	18.8%	0%	−0.07	−0.02	0.04
Energy-retaining bank	200	32000	81.8%	18.1%	0%	−0.07	−0.02	0.04
Safe exit	207	33120	82.5%	17.5%	0%	−0.07	−0.02	0.04

Table G.2 Retained held-out primitive validation by entry class. Distances are in metres; time spent in useful lift is in seconds.

Entry class	Variants	Replays	Continuation valid	Terminal useful	Hard failure	Wall margin	Mean ΔH	Mean T_ℓ
Boundary	527	84320	45%	55%	0%	−0.07	−0.03	0.03
In flight	528	84480	100%	0%	0%	0.21	−0.00	0.06
Launch	432	69120	100%	0%	0%	0.00	−0.01	0.02
Recovery	118	18880	100%	0%	0%	0.43	−0.08	0.06

Table G.3 Held-out transition compatibility by entry and exit class.

Entry class	Exit class	Replays	Compatible	Hard failure	Validation result
Boundary	Boundary	25296	0%	0%	non-continuing class
	In flight	7115	100%	0%	compatible continuation
	Recovery	5533	0%	0%	non-continuing class
	Safe terminal	46376	100%	0%	compatible continuation
In flight	Boundary	42677	100%	0%	compatible continuation
	In flight	19260	100%	0%	compatible continuation
	Recovery	22543	0%	0%	non-continuing class

Continued on the next page.

Table G.3 Held-out transition compatibility by entry and exit class. (continued)

Entry class	Exit class	Replays	Compatible	Hard failure	Validation result
Launch	Boundary	3119	0%	0%	non-continuing class
	In flight	27703	100%	0%	compatible continuation
	Post-launch degraded	38298	100%	0%	compatible continuation
ecovery	Boundary	3945	0%	0%	non-continuing class
	In flight	4438	100%	0%	compatible continuation
	Recovery	10497	5.7%	0%	partly compatible

G.1.2 Supplementary Viability Governor Tuning Statistics

Table G.4 Viability governor tuning result by library tier and spatial memory history length. Mean terminal energy reserve is in metres.

Library	History h	Runs	Safe	Exit face	Hard failure	No viable	Lateral wall	Memory switch	Mean H_{res}	Mean J_{sim}
No-cluster	0	50	58%	54%	16%	6%	8%	0%	1.87	47.2
	3	50	58%	54%	16%	6%	8%	10%	1.87	47.2
	10	50	58%	54%	16%	6%	8%	10%	1.87	47.2
	30	50	58%	54%	16%	6%	8%	12%	1.87	47.2
Super-light	0	50	58%	54%	14%	8%	8%	0%	1.87	48.4
	3	50	58%	54%	14%	8%	8%	6%	1.87	48.4
	10	50	58%	54%	14%	8%	8%	4%	1.87	48.4
	30	50	58%	54%	14%	8%	8%	8%	1.87	48.4
Light	0	50	58%	54%	16%	6%	8%	0%	1.86	47.2
	3	50	58%	54%	16%	6%	8%	4%	1.86	47.2
	10	50	58%	54%	16%	6%	8%	4%	1.86	47.2
	30	50	58%	54%	16%	6%	8%	6%	1.86	47.2
Balanced	0	50	58%	54%	14%	8%	8%	0%	1.87	48.4
	3	50	58%	54%	14%	8%	8%	6%	1.87	48.4
	10	50	58%	54%	14%	8%	8%	6%	1.87	48.4
	30	50	58%	54%	14%	8%	8%	8%	1.87	48.4
Heavy	0	50	54%	50%	24%	8%	10%	0%	1.83	33.6
	3	50	54%	50%	24%	8%	10%	0%	1.83	33.6
	10	50	54%	50%	24%	8%	10%	0%	1.83	33.6
	30	50	54%	50%	24%	8%	10%	0%	1.83	33.6

Table G.5 Selector runtime audit by primitive library tier. Timing values are in milliseconds.

Library	Size	Decisions	Mean candidates	Max candidates	Mean time	99th time	Max time	$T < 0.100$
No-cluster	1605	1976	108.9	144	278.1	1228.2	1895.0	33.7%
Super-light	374	1980	28.1	41	155.4	796.9	1170.3	62.3%
Light	208	1980	15.3	23	95.4	510.4	909.6	67.9%
Balanced	112	1988	7.4	18	60.4	355.1	616.5	81.5%
Heavy	40	1960	3.0	10	49.6	195.8	5223.6	93.5%

G.1.3 Supplementary Simulated Mission Statistics

Table G.6 Mission simulation by held-out environment ladder and launch speed for the balanced controller. Mean terminal specific energy is in metres.

Environment	$\lVert v(t_0) \rVert$	Runs	Safe	Exit face	Hard failure	No viable	Lateral wall	Mean H_{res}	Mean J_{sim}
Still-air	4.0–5.0	35	0%	0%	0%	34.3%	0%	1.17	−13.7
	5.0–6.0	45	40%	31.1%	37.8%	22.2%	8.9%	1.37	−15.8
	6.0–7.0	41	82.9%	73.2%	12.2%	4.9%	9.8%	1.96	68.0
	≥ 7.0	59	79.7%	74.6%	11.9%	8.5%	5.1%	2.65	71.7
Fixed single-fan	4.0–5.0	35	5.7%	5.7%	0%	28.6%	0%	1.15	−3.5
	5.0–6.0	45	71.1%	60%	22.2%	6.7%	11.1%	1.39	47.3
	6.0–7.0	41	97.6%	90.2%	0%	2.4%	7.3%	2.09	112.3
	≥ 7.0	59	93.2%	83.1%	5.1%	1.7%	10.2%	2.80	96.0
Fixed four-fan	4.0–5.0	35	51.4%	51.4%	2.9%	8.6%	2.9%	1.29	68.7
	5.0–6.0	45	93.3%	75.6%	4.4%	2.2%	17.8%	1.75	94.8
	6.0–7.0	41	100%	92.7%	0%	0%	7.3%	2.31	126.9
	≥ 7.0	59	100%	94.9%	0%	0%	5.1%	3.03	128.4
Fan parameter variation	4.0–5.0	35	22.9%	22.9%	5.7%	17.1%	5.7%	1.14	21.5
	5.0–6.0	45	82.2%	57.8%	13.3%	4.4%	31.1%	1.49	52.7
	6.0–7.0	41	95.1%	87.8%	4.9%	0%	9.8%	2.21	113.6
	≥ 7.0	59	100%	93.2%	0%	0%	6.8%	2.96	124.5
Local fan-position variation	4.0–5.0	35	28.6%	28.6%	5.7%	17.1%	5.7%	1.10	29.4
	5.0–6.0	45	75.6%	60%	15.6%	8.9%	24.4%	1.49	56.4
	6.0–7.0	41	90.2%	85.4%	9.8%	0%	12.2%	2.18	105.4
	≥ 7.0	59	98.3%	93.2%	1.7%	0%	5.1%	2.97	124.2
Active-fan-count variation	4.0–5.0	35	11.4%	8.6%	0%	22.9%	2.9%	1.16	1.2
	5.0–6.0	45	60%	46.7%	35.6%	4.4%	15.6%	1.41	15.6
	6.0–7.0	41	92.7%	82.9%	2.4%	4.9%	9.8%	2.07	98.3
	≥ 7.0	59	91.5%	86.4%	0%	8.5%	5.1%	2.78	105.9

Continued on the next page.

Table G.6 Mission simulation by held-out environment ladder and launch speed for the balanced controller. Mean terminal specific energy is in metres. (continued)

Environment	$\lVert \mathbf{v}(t_0) \rVert$	**Runs**	**Safe**	**Exit face**	**Hard failure**	**No viable**	**Lateral wall**	**Mean H_{res}**	**Mean J_{sim}**
Full-environment variation	4.0–5.0	35	5.7%	2.9%	0%	20%	2.9%	1.16	−5.3
	5.0–6.0	45	62.2%	46.7%	28.9%	8.9%	15.6%	1.41	19.5
	6.0–7.0	41	92.7%	82.9%	4.9%	2.4%	9.8%	2.08	97.7
	≥ 7.0	59	89.8%	84.7%	3.4%	6.8%	5.1%	2.79	100.5
Full-domain variation	4.0–5.0	35	0%	0%	2.9%	22.9%	2.9%	1.09	−12.0
	5.0–6.0	45	48.9%	40%	35.6%	15.6%	11.1%	1.29	2.4
	6.0–7.0	41	80.5%	68.3%	12.2%	7.3%	14.6%	1.90	65.7
	≥ 7.0	59	81.4%	76.3%	11.9%	6.8%	10.2%	2.63	78.0

Table G.7 Mission simulation by primitive library tier and spatial memory history length for $\lVert \mathbf{v}(t_0) \rVert \geq 5.0\,\mathrm{m\,s^{-1}}$. Mean terminal specific energy is in metres.

Library	**History h**	**Runs**	**Safe**	**Exit face**	**Hard failure**	**No viable**	**Lateral wall**	**Memory switch**	**Mean H_{res}**	**Mean J_{sim}**
No-cluster	0	1160	83.8%	75.1%	11.9%	4.3%	10.3%	0%	2.19	80.9
	3	1160	83.9%	75.2%	11.9%	4.2%	10.2%	11.4%	2.19	81.1
	10	1160	83.9%	75.2%	11.9%	4.2%	10.2%	12.1%	2.19	81.1
	30	1160	83.7%	75.1%	12.1%	4.2%	10.3%	12.8%	2.19	80.8
Super-light	0	1160	82.7%	73.4%	12.9%	4.4%	11.9%	0%	2.19	77.4
	3	1160	82.8%	73.5%	12.8%	4.4%	11.8%	9.3%	2.19	77.6
	10	1160	82.8%	73.5%	12.8%	4.4%	11.8%	9.7%	2.19	77.6
	30	1160	82.7%	73.5%	12.9%	4.4%	11.8%	10.3%	2.19	77.6
Light	0	1160	83.3%	74.2%	12.2%	4.6%	10.9%	0%	2.19	79.3
	3	1160	83.4%	74.3%	12.1%	4.6%	10.8%	7.1%	2.19	79.6
	10	1160	83.4%	74.3%	12.1%	4.6%	10.8%	7.5%	2.19	79.6
	30	1160	83.3%	74.2%	12.2%	4.6%	10.9%	8.1%	2.19	79.3
Balanced	0	1160	83.9%	74.6%	10.9%	5.3%	10.8%	0%	2.19	80.6
	3	1160	84.1%	74.8%	10.7%	5.2%	10.7%	7.1%	2.19	81.2
	10	1160	84.1%	74.8%	10.7%	5.2%	10.7%	7.6%	2.19	81.2
	30	1160	83.9%	74.6%	10.9%	5.2%	10.8%	7.7%	2.19	80.6
Heavy	0	1160	80%	70.3%	14.2%	5.8%	11%	0%	2.13	71.2
	3	1160	80%	70.3%	14.6%	5.4%	11%	2.7%	2.13	71.0
	10	1160	80%	70.3%	14.6%	5.4%	11%	2.8%	2.13	71.0
	30	1160	80%	70.3%	14.6%	5.4%	11%	2.9%	2.13	71.0

G.2 Real-Flight Experiment Result

Table G.8 Still-air flight test records.

Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Open-loop	10	3/10 (30.0%)	7/10 (70.0%)	0/10 (0.0%)	0/10 (0.0%)	1.63 m
Closed-loop	30	28/30 (93.3%)	2/30 (6.7%)	0/30 (0.0%)	0/30 (0.0%)	1.75 m
- with memory	30	21/30 (70.0%)	9/30 (30.0%)	0/30 (0.0%)	0/30 (0.0%)	1.77 m

Table G.9 Single-fan updraft flight test records.

Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Open-loop	10	6/10 (60.0%)	4/10 (40.0%)	0/10 (0.0%)	0/10 (0.0%)	1.51 m
Closed-loop	30	24/30 (80.0%)	6/30 (20.0%)	0/30 (0.0%)	0/30 (0.0%)	1.64 m
- with memory 1st	30	21/30 (70.0%)	9/30 (30.0%)	0/30 (0.0%)	0/30 (0.0%)	1.59 m
- with memory 2nd	30	23/30 (76.7%)	5/30 (16.7%)	0/30 (0.0%)	2/30 (6.7%)	1.66 m

Table G.10 Four-fan updraft flight test records.

Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Open-loop	10	1/10 (10.0%)	9/10 (90.0%)	0/10 (0.0%)	0/10 (0.0%)	1.12 m
Closed-loop	30	18/30 (60.0%)	11/30 (36.7%)	0/30 (0.0%)	1/30 (3.3%)	1.77 m
- with memory 1st	30	16/30 (53.3%)	14/30 (46.7%)	0/30 (0.0%)	0/30 (0.0%)	1.62 m
- with memory 2nd	30	18/30 (60.0%)	12/30 (40.0%)	0/30 (0.0%)	0/30 (0.0%)	1.58 m

Table G.11 Randomised three-fan updraft flight test records.

Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Open-loop	10	3/10 (30.0%)	6/10 (60.0%)	1/10 (10.0%)	0/10 (0.0%)	1.63 m
Closed-loop	30	26/30 (86.7%)	4/30 (13.3%)	0/30 (0.0%)	0/30 (0.0%)	1.90 m
- with memory	30	21/30 (70.0%)	5/30 (16.7%)	2/30 (6.7%)	2/30 (6.7%)	1.85 m

Table G.12 Randomised four-fan updraft flight test records.

Controller	Throws	Exit face	Floor impact	Lateral wall	Other	Mean H_{res}
Open-loop	10	2/10 (20.0%)	7/10 (70.0%)	1/10 (10.0%)	0/10 (0.0%)	1.74 m
Closed-loop	20	18/20 (90.0%)	0/20 (0.0%)	2/20 (10.0%)	0/20 (0.0%)	1.85 m
Closed-loop	10	10/10 (100.0%)	0/10 (0.0%)	0/10 (0.0%)	0/10 (0.0%)	1.80 m
- with memory	30	27/30 (90.0%)	3/30 (10.0%)	0/30 (0.0%)	0/30 (0.0%)	1.96 m