Chapter Outline

Other Vehicle Types and Advanced Control Cases

Chapter 5 Filtering, State Estimation, and Low-Level Control

5.1 Filtering and Low-Level Control

5.2 Sensor Calibration and Estimation

5.3 Controller Design and Allocation

5.4 Control Performance and Tuning

5.5 PX4 Architecture and Interfaces

5.6 Control Development and Testing

5.7 Other Vehicles and Advanced Cases

5.8 Exercises and Classroom Discussion

4 Units in This Section

5.7.1 Other Vehicle Extensions

Textbook: Sec. 5.2.2.4

5.7.2 Low-Level Fixed-Wing Flight Control

Textbook: Sec. 5.2.2.4

5.7.3 Control Cases for Complex Configurations

Textbook: Secs. 5.3.1–5.3.2

5.7.4 Advanced Cases and Experiment Selection

Textbook: Secs. 5.3.1–5.3.2

5.7.1 Other Vehicle Extensions: Overview of Vehicle Types

A Different Vehicle Changes the Available Forces and Moments

Vehicle type	Main control inputs	Control conditions
Multicopter	Individual motor thrusts	Thrust range, response speed, and configuration
Fixed-wing aircraft	Throttle and aerodynamic control surfaces	Airspeed, angle of attack, and trim
Helicopter **Tiltrotor**	Blade pitch, servos, or tilt mechanisms	Rotor coupling and operating mode
Unmanned ground vehicle **Unmanned surface vehicle**	Propulsive force and steering	Ground-contact / hydrodynamic constraints and speed

Prerequisites: trim (slide unit 4.4.1); airspeed, angle of attack, dynamic pressure (4.8.2); blade pitch (4.8.4).

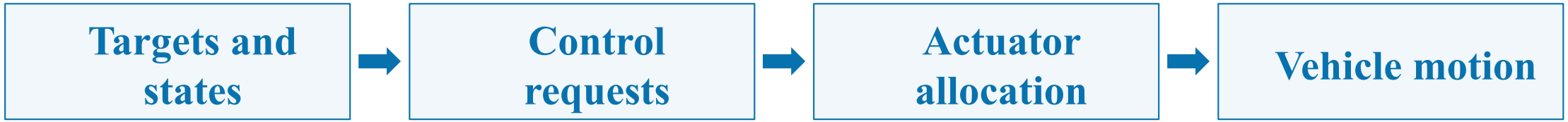
5.7.1 Other Vehicle Extensions: Common Comparison Dimensions

Rebuild Each Part of the Same Closed-Loop Framework

Comparison	Quadrotor baseline	Changes for fixed-wing aircraft
States and measurements	Attitude, angular rate, position	Airspeed and angle of attack: measure or estimate for control and protection
Requested quantities	Total thrust and three-axis moments	Thrust and moments from control surfaces
Allocation and constraints	Motor thrust and speed limits	Surface deflection, rate limits, and control effectiveness
Validation conditions	Near-hover operation and disturbances	Near trim airspeed and in turns

5.7.1 Other Vehicle Extensions: Interfaces and Physical Plants

Interfaces Carry Targets; the Vehicle Delivers Control Authority



- **Reuse:** subscription/publication structure, log formats, and staged validation methods.
- **Rebuild:** operating points, feedback states, control effectiveness, and actuator limits.
- **Check directions first:** apply a small single-channel perturbation and compare state changes with output directions.

Common interfaces reduce integration work;
control design must still fit the plant.

Chapter Outline

Other Vehicle Types and Advanced Control Cases

Chapter 5 Filtering, State Estimation, and Low-Level Control

5.1 Filtering and Low-Level Control

5.2 Sensor Calibration and Estimation

5.3 Controller Design and Allocation

5.4 Control Performance and Tuning

5.5 PX4 Architecture and Interfaces

5.6 Control Development and Testing

5.7 Other Vehicles and Advanced Cases

5.8 Exercises and Classroom Discussion

4 Units in This Section

5.7.1 Other Vehicle Extensions

Textbook: Sec. 5.2.2.4

5.7.2 Low-Level Fixed-Wing Flight Control

Textbook: Sec. 5.2.2.4

5.7.3 Control Cases for Complex Configurations

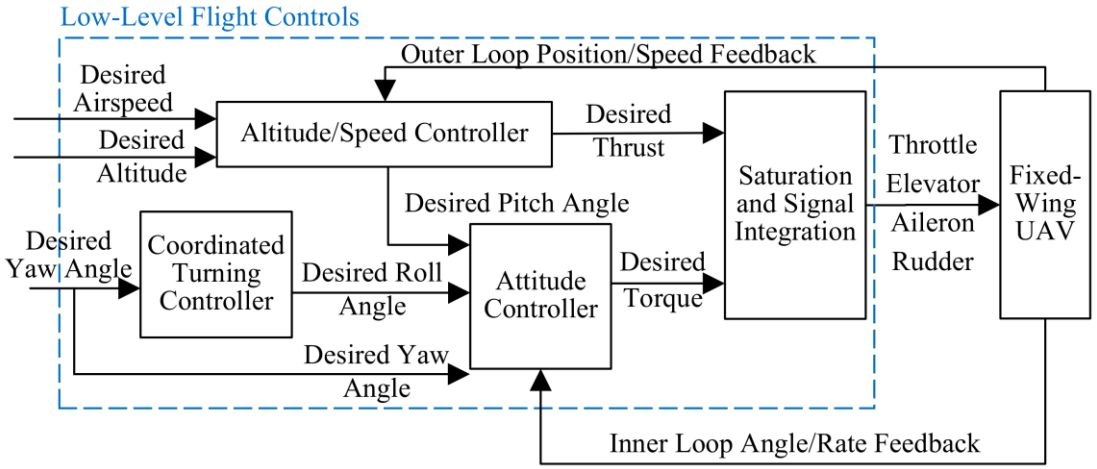
Textbook: Secs. 5.3.1–5.3.2

5.7.4 Advanced Cases and Experiment Selection

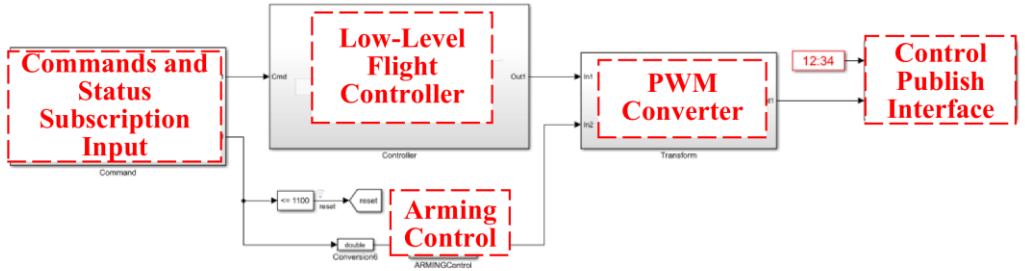
Textbook: Secs. 5.3.1–5.3.2

5.7.2 Low-Level Fixed-Wing Flight Control: Figure 5.13 Overview

Outer Loops Set Motion Targets; Inner Loops Control Attitude



(a) Small Fixed-Wing UAV Flight Control Structure Diagram



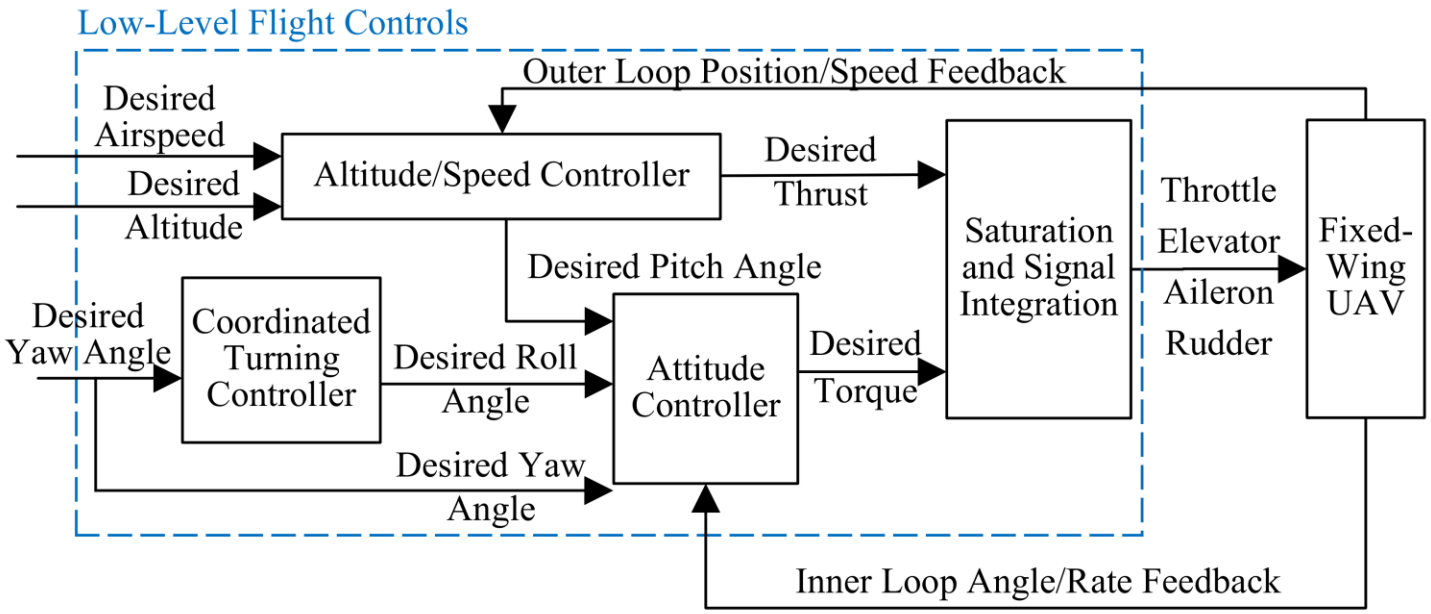
(b) Fixed-Wing Low-Level Control Simulink Model Block Diagram

- **Figure 5.13(a):** altitude/speed, coordinated turns, attitude, actuator integration.
- **Figure 5.13(b):** subscribe → control → pulse-width modulation (PWM) conversion → publish.
- **Two feedback levels:** outer loops govern motion; inner loops use angles and angular rates.
- **Coordinated turn:** bank tilts lift for centripetal force; sideslip stays near zero.

Figure 5.13 Fixed-Wing UAV Control
UAV: unmanned aerial vehicle

5.7.2 Low-Level Fixed-Wing Flight Control: Layered Control Structure

Pitch and Thrust Couple Altitude and Airspeed



(a) Small Fixed-Wing UAV Flight Control Structure Diagram

Figure 5.13(a) Outer loops set thrust and pitch targets; inner loops use angle and angular-rate feedback

Coupling: pitching up to climb reduces airspeed; more throttle increases speed and causes a climb. Coordinate both.

Layers: outer-loop integral action can remove altitude/airspeed offset; inner-loop angular-rate feedback adds damping.

5.7.2 Low-Level Fixed-Wing Flight Control: One-Step Pitch Calculation

Feedback Converts Pitch Error into a Limited Elevator Request

Independent instructional model: near trim, pitch channel only, radians internally.

θ: pitch angle; q: body pitch rate (rad/s). δ_e: normalized elevator increment from trim, limited to $[-0.25,0.25]$.

$$\dot{\theta} = q, \quad \dot{q} = -2q + 4\delta_e$$

$$\delta_e = \mathrm{sat}_{[-0.25,0.25]}[(\theta_r - \theta) - 0.3q]$$

$$\theta_r = 10^\circ = 0.17453\,\mathrm{rad}, \quad \theta(0) = q(0) = 0$$

Coefficients: -2 is aerodynamic pitch damping; 4 is elevator effectiveness; 0.3 is rate feedback adding damping.

One step: $\dot{q}(0) = 4 \times 0.17453 = 0.698\,\mathrm{rad/s^2}$;

Euler approximation $q(0.01) \approx 0.0070\,\mathrm{rad/s}$

RK45 integration to $t = 0.01\,\mathrm{s}$: $q = 0.006870\,\mathrm{rad/s}$, $\theta = 0.001979^\circ$

5.7.2 Low-Level Fixed-Wing Flight Control: Complete Response of the Same Model

Pitch Peaks at About 2.6 s, Then Converges to the Target

Result: target ±2% band at about 1.9 s; peak 10.152° at 2.62 s (1.52% overshoot).

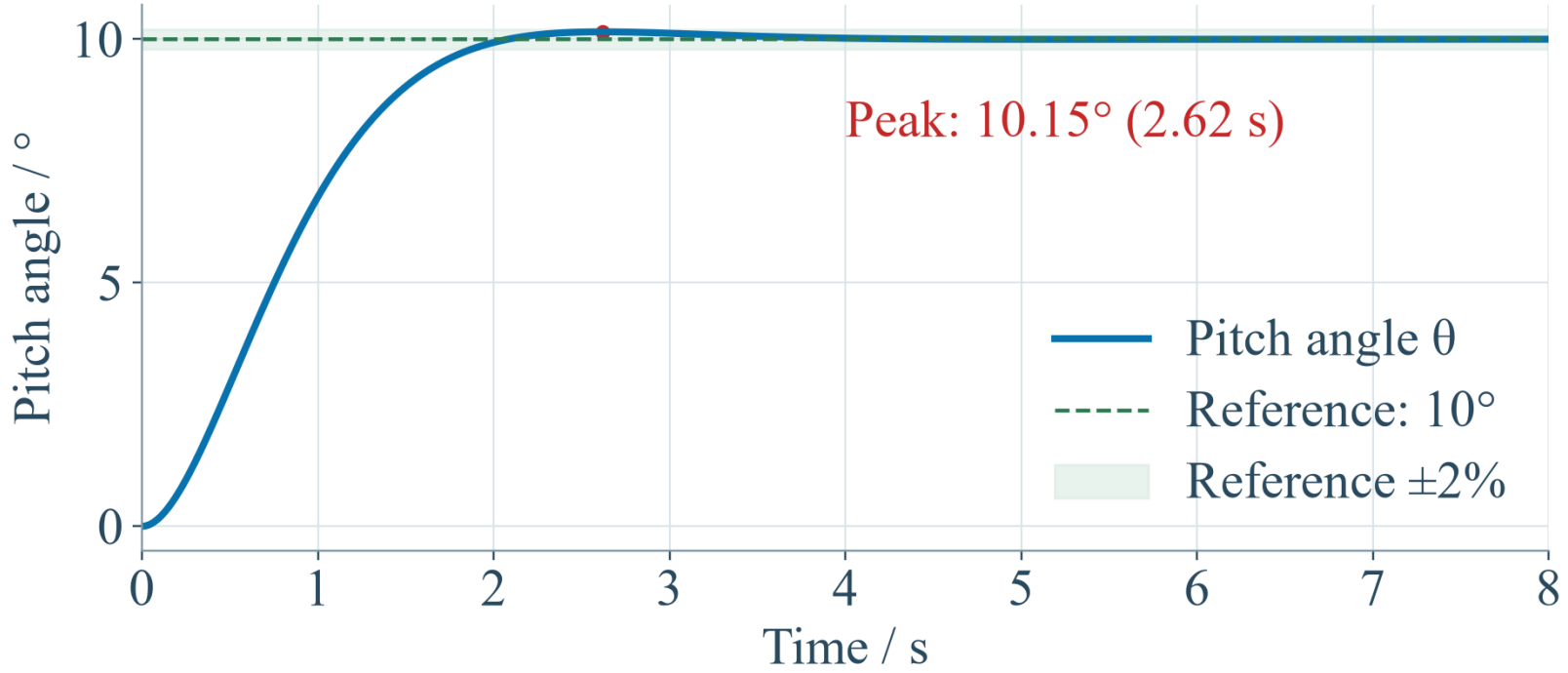


Figure: Local pitch-model response (instructional example)

5.7.2 Low-Level Fixed-Wing Flight Control: Interpreting the Complete Response

Check Surface Commands and Operating Point Alongside Pitch

Same independent instructional model; elevator is a normalized increment from trim.

Time / s	Pitch / °	Angular rate / rad·s^{-1}	Elevator increment
0	0.0000	0.00000	0.17453
0.5	2.9087	0.14760	0.07949
1	6.7594	0.10948	0.02372
2	9.9335	0.01602	-0.00364
5	9.9980	-0.00005	0.00005
10	10.0000	-0.00000	0.00000

Result: peak pitch 10.152°, overshoot 1.52%; peak absolute elevator request 0.17453, below the 0.25 limit.

5.7.2 Low-Level Fixed-Wing Flight Control: Airspeed and Control-Surface Effectiveness

Airspeed Changes the Control Effect of the Same Surface Deflection

q_∞: dynamic pressure; ρ: air density; V_a: airspeed (velocity relative to air).

$$q_\infty = \frac{1}{2}\rho V_a^2 \qquad M_\delta = q_\infty S\bar{c}\, C_{m\delta_e}\, \delta_e$$

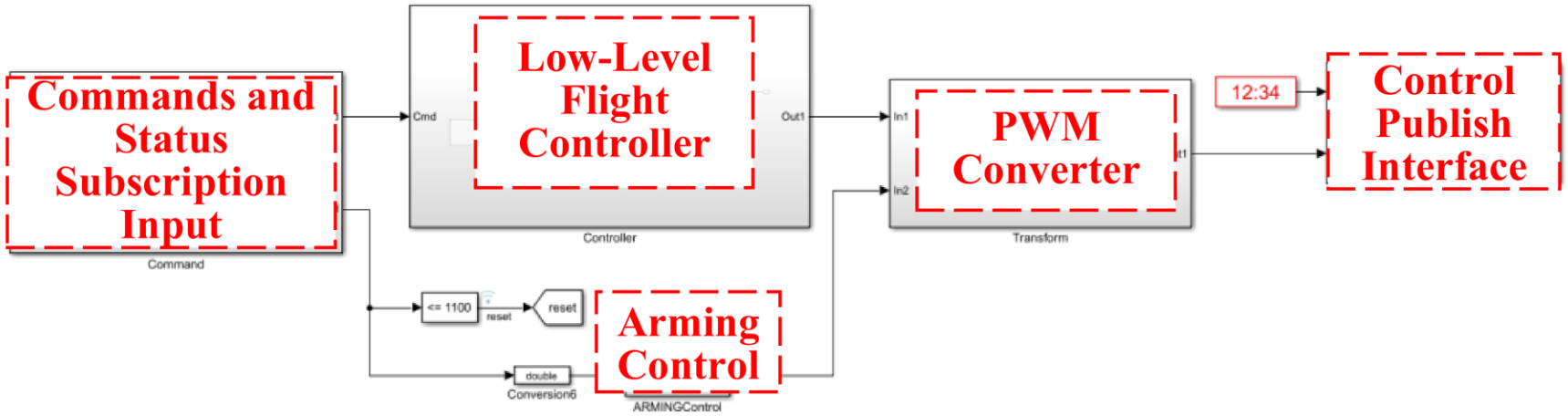
S: wing area; $\bar{c}$: mean aerodynamic chord; $C_{m\delta_e}$: elevator pitching-moment coefficient.

Example airspeed	Dynamic pressure (ρ=1.225 kg/m³)	Result
10 m/s	$0.5 \times 1.225 \times 10^2$	61.25 Pa
20 m/s	$0.5 \times 1.225 \times 20^2$	245.00 Pa

Result: doubling airspeed gives about 4 times dynamic pressure and same-deflection moment.
Gain scheduling: gains are often scheduled by airspeed as effectiveness and damping vary.
Stall protection: low speed and high angle of attack promote stall; use angle-of-attack limits and minimum-airspeed protection.

5.7.2 Low-Level Fixed-Wing Flight Control: Program and Publication Interfaces

Control Calculations and Arming Conditions Both Determine Output



(b) Fixed-Wing Low-Level Control Simulink Model Block Diagram

Figure 5.13(b) Implementation: subscribe to inputs → controller → PWM conversion → publish

- **Protocol conversion:** configure normalized ranges, PWM ranges, and channels separately for motors and servos.
- **Output permission:** arming, data validity, and a single active control source jointly govern publication.

5.7.2 Low-Level Fixed-Wing Flight Control: Pitch-Channel Example

Read the Pitch Chain, Then Compare Model and Hardware Inputs

Root directory: [安装目录]/RflySimAPIs/5.RflySimFlyCtrl/
1.BasicExps/e10-FixedWingCtrl/code_6/e4-1/
Software-in-the-loop (SITL; SIL in the textbook). Read Readme.pdf.
Run InitDatactrl.m; open SmallFixedWingUAVattitude.slx.
Pitch input: Command / PitchCH2 step block; step time 10, final value 100.

Open e4-1 Initialize → **Open attitude model Set pitch input** → **Run to completion Data Inspector** → **Compare tracking Surfaces and airspeed**

Hardware-in-the-loop (HITL; HIL in the textbook) comparison: code_6/e4-4/SmallFixedWingUAVattitudeHIL_pitch.slx

- **Acceptance:** pitch tracks the step target and converges; no sustained surface saturation.
- **Troubleshoot:** if divergent, check proportional–integral–derivative (PID) gains, initialization, and step amplitude. If traces are missing, enable simulation data logging.

Chapter Outline

Other Vehicle Types and Advanced Control Cases

Chapter 5 Filtering, State Estimation, and Low-Level Control

5.1 Filtering and Low-Level Control

5.2 Sensor Calibration and Estimation

5.3 Controller Design and Allocation

5.4 Control Performance and Tuning

5.5 PX4 Architecture and Interfaces

5.6 Control Development and Testing

5.7 Other Vehicles and Advanced Cases

5.8 Exercises and Classroom Discussion

4 Units in This Section

5.7.1 Other Vehicle Extensions

Textbook: Sec. 5.2.2.4

5.7.2 Low-Level Fixed-Wing Flight Control

Textbook: Sec. 5.2.2.4

5.7.3 Control Cases for Complex Configurations

Textbook: Secs. 5.3.1–5.3.2

5.7.4 Advanced Cases and Experiment Selection

Textbook: Secs. 5.3.1–5.3.2

5.7.3 Control Cases for Complex Configurations: Helicopter Control Inputs

Collective, Cyclic, and Tail-Rotor Inputs Have Different Roles

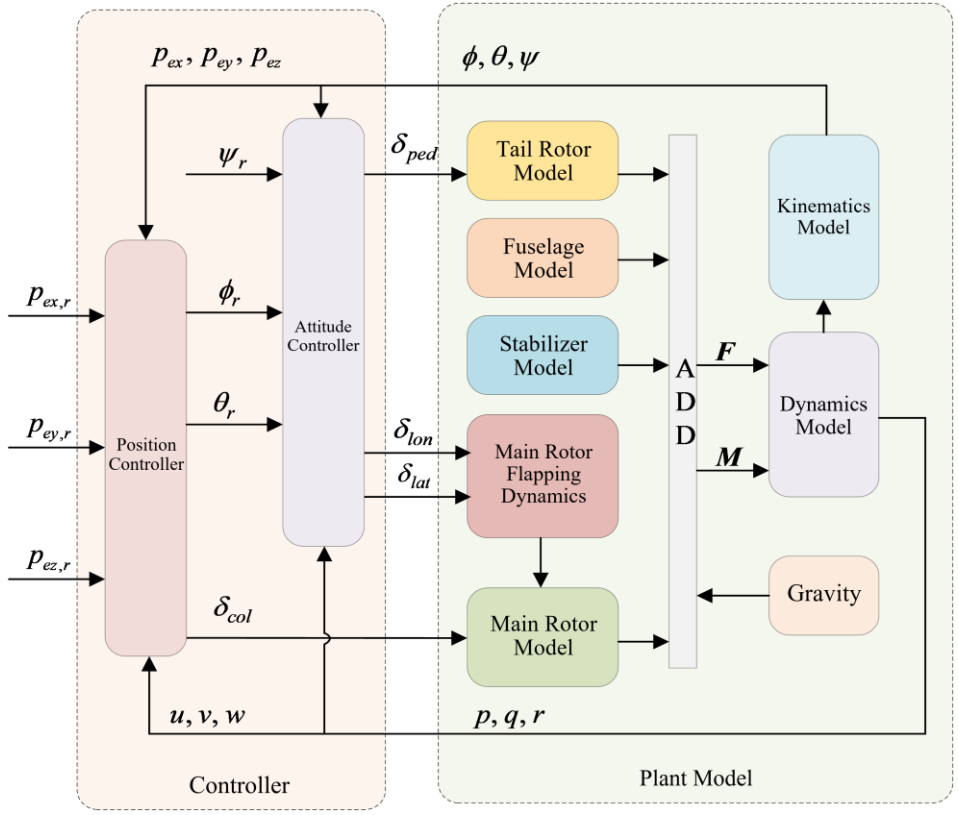


Figure 5.14(d) Enlarged helicopter-model detail

- Collective δ_{col} changes all blade pitches together to adjust total rotor thrust.
- Cyclic: δ_{lon} (longitudinal) and δ_{lat} (lateral) vary blade pitch with azimuth to control the rotor force direction.
- Tail rotor δ_{ped} contributes to yaw control and balances main-rotor reaction torque.
- Allocation: servos and the swashplate combine collective and cyclic; blade-pitch limits and mechanism response constrain motion.

5.7.3 Control Cases for Complex Configurations: Swashplate Mixing

Common Servo Motion Changes Collective; Differential Motion Changes Cyclic

Common motion shifts the plate; differential motion gives cyclic pitch

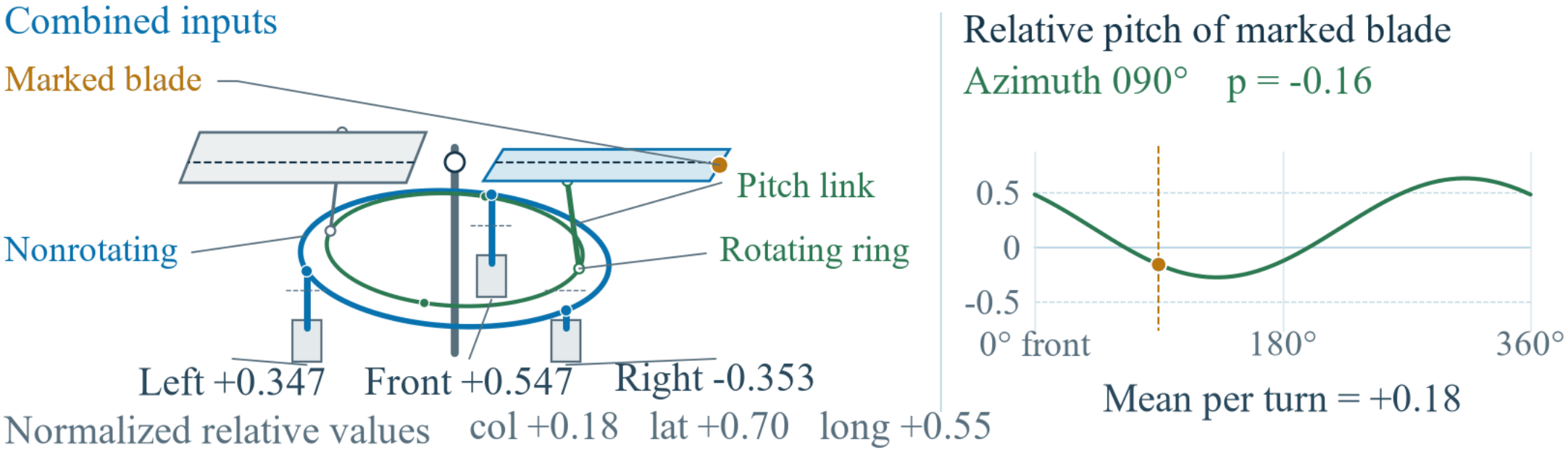


Figure: Three servos drive the swashplate and cyclic pitch.

Readings: servo values are normalized relative outputs from the e4 mixer. Follow the marked blade to see pitch vary with azimuth.

5.7.3 Control Cases for Complex Configurations: Helicopter Validation Chain

Stabilize the Attitude Inner Loop Before Adding Position Feedback

HITL requires Pixhawk 6X and a transmitter; recommended firmware 1.12.3, Blade 130X airframe.

2.AdvExps/e4_HELI-CtrlExp/1.AttitudeControl/
Readme.pdf · Init_AttControl.m · AttitudeControl.slx · HITLRun.bat

Run: build/upload → set airframe/enable HITL in QGroundControl (QGC) → HITLRun.bat → transmitter arm

$$\begin{bmatrix} s_{\mathrm{f}} \\ s_{\mathrm{r}} \\ s_{\mathrm{l}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0.667 \\ 1 & -0.5 & -0.333 \\ 1 & 0.5 & -0.333 \end{bmatrix} \begin{bmatrix} \delta_{\mathrm{col}} \\ \delta_{\mathrm{lat}} \\ \delta_{\mathrm{lon}} \end{bmatrix}$$

Swashplate mixing (u2servo): s_{f}, s_{r}, s_{l} are front, right, and left servos. Common motion adds collective; differential motion produces cyclic.

Position extension (full edition): 2.PositionControl/PosControl.slx adds position/velocity outer loops.

- **Acceptance:** arm only after CopterSim reports extended Kalman filter (EKF) initialization complete. Small single-stick inputs: matching attitude response, little cross-channel motion.
- **Troubleshoot:** Readme1.pdf FAQ covers firmware versions and build commands.

5.7.3 Control Cases for Complex Configurations: Two Tiltrotor Modes

Rotor and Fixed-Wing Modes Use Different Control Actions

Simplified two-mode example; firmware 1.13.2. In installer item 10, “是否屏蔽PX4自身控制器输出” (suppress the built-in controller outputs), select “是” (Yes).

Comparison	Rotor mode in this example	Fixed-wing mode in this example
Motion targets	Attitude stabilization, position or altitude hold	Constant speed, course, and coordinated turns
Key feedback	Attitude and angular rates; position and velocity	Also airspeed and course states
Output action	Three servos per side combine collective and longitudinal cyclic	Surfaces generate moments; common collective supplies propulsion.

Model locations: Helicopter Mode / FixedWing Mode

2.AdvExps/e6_Tiltrotor_CtrlExp/Readme.pdf

Controller/Init_AllMode_HIL.m · Tiltrotor_AllMode_HIL.slx

Run: upload → enable HITL → CopterSim → CH7 arm → CH5 position hold → CH6 mode switch

5.7.3 Control Cases for Complex Configurations: Tiltrotor Configuration and Checks

Both Nacelles Tilt; Three Servos per Side Set Blade Pitch

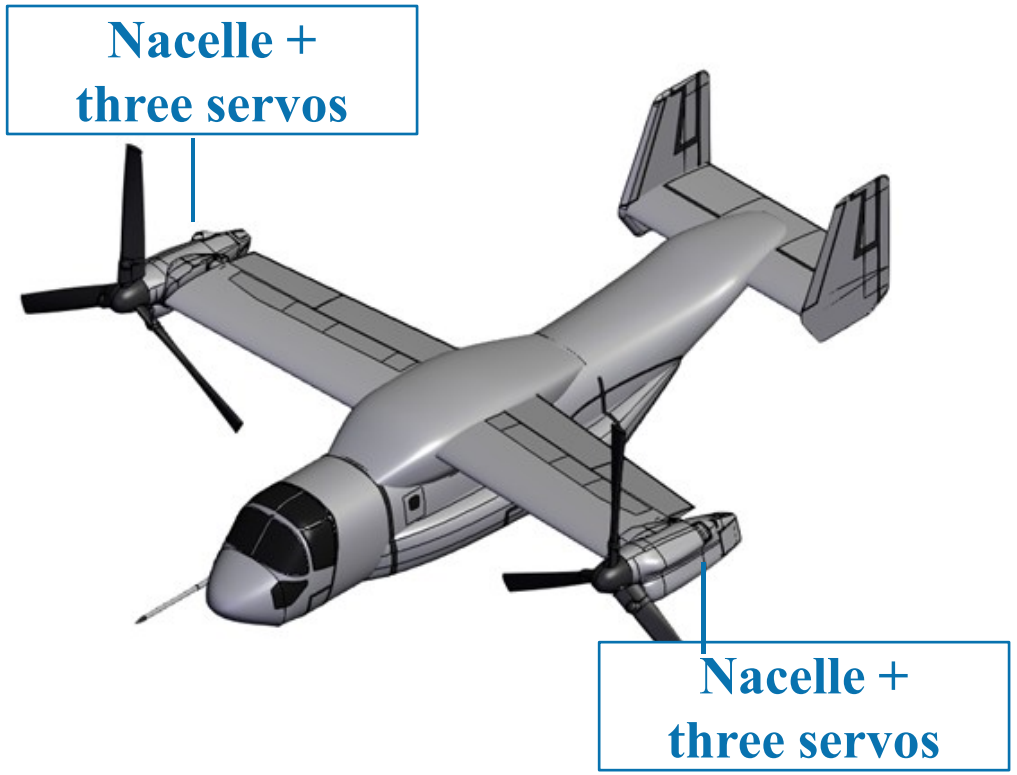


Figure: e6 tiltrotor configuration (example model block icon)

- **Configuration:** side-by-side twin rotors in tilting nacelles.
- **Rotor mode:** differential collective → roll; common longitudinal cyclic → pitch; differential longitudinal cyclic → yaw; common collective → vertical motion.
- **Fixed-wing mode:** surfaces generate moments; common collective provides propulsion.
- **Acceptance:** after EKF initialization and arming, CH5 selects position hold/constant speed; use CH6 to check the 3D transition.
- **Troubleshoot:** first copy QZXY and “倾转旋翼.xml” into the RflySim3D path.

5.7.3 Control Cases for Complex Configurations: Mode Transitions

Test Output Continuity During Mode Transitions Separately

Old mode stable → **States and targets** → **Bumpless handover** → **Validate new mode**

Normalized outputs for the same actuator channel: old controller 0.35, new controller 0.62.

$$\Delta u = 0.62 - 0.35 = 0.27 > 0.05$$

- **Instructional threshold:** allowable output jump 0.05; set separately from the tracking-error threshold.
- **Bumpless handover:** at switching, set the new controller’s integral contribution to “0.35−proportional contribution”, giving output 0.35. See “Control Source Switching Calculation” in slide deck 5.5.
- **Validity:** check airspeed, tilt angle, servo rates, data validity, and fallback after a failed transition.

Configuration changes alter control action; validate the handover as well.

Chapter Outline

Other Vehicle Types and Advanced Control Cases

Chapter 5 Filtering, State Estimation, and Low-Level Control

5.1 Filtering and Low-Level Control

5.2 Sensor Calibration and Estimation

5.3 Controller Design and Allocation

5.4 Control Performance and Tuning

5.5 PX4 Architecture and Interfaces

5.6 Control Development and Testing

5.7 Other Vehicles and Advanced Cases

5.8 Exercises and Classroom Discussion

4 Units in This Section

5.7.1 Other Vehicle Extensions

Textbook: Sec. 5.2.2.4

5.7.2 Low-Level Fixed-Wing Flight Control

Textbook: Sec. 5.2.2.4

5.7.3 Control Cases for Complex Configurations

Textbook: Secs. 5.3.1–5.3.2

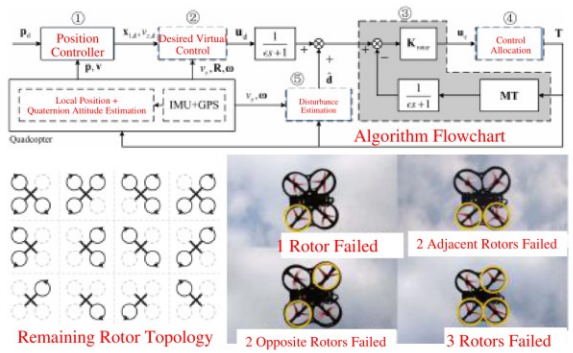
5.7.4 Advanced Cases and Experiment Selection

Textbook: Secs. 5.3.1–5.3.2

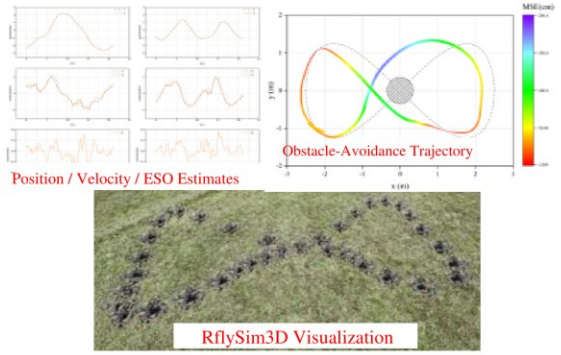
5.7 Other Vehicles and Advanced Cases

5.7.4 Advanced Cases and Experiment Selection: Overview of Advanced Cases

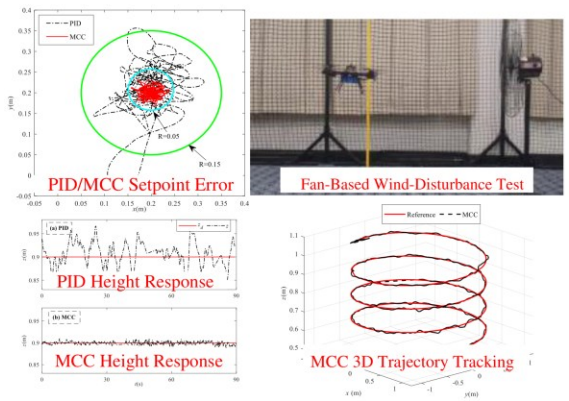
Identify the Research Question Before Reading Methods and Evidence



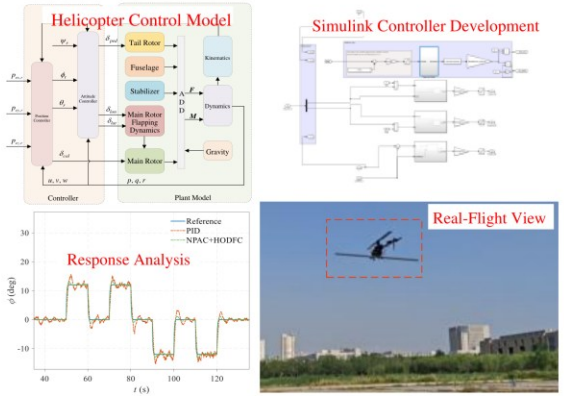
(a) Fault-Tolerant Control Case Study for a Quadcopter



(b) Optimal Flight Control Based on Reinforcement Learning



(c) Model-Compensation Control for a Quadcopter



(d) Control Development of Small Unmanned Helicopters

- **Figure 5.14(a) Fault tolerance:** what is controllable after one to three rotor failures?
- **Figure 5.14(b) Reinforcement learning:** approximate policy iteration for tracking and obstacle avoidance.
- **Figure 5.14(c) Model compensation:** counter model errors and external disturbances.
- **Figure 5.14(d) Helicopter:** decoupled attitude/position modeling and control.

Figure 5.14 Advanced Control Case Studies

5.7.4 Advanced Cases and Experiment Selection: Fault-Tolerant Control Mechanism

Disturbance Estimation and Dynamic Allocation Adapt to Actuator Authority

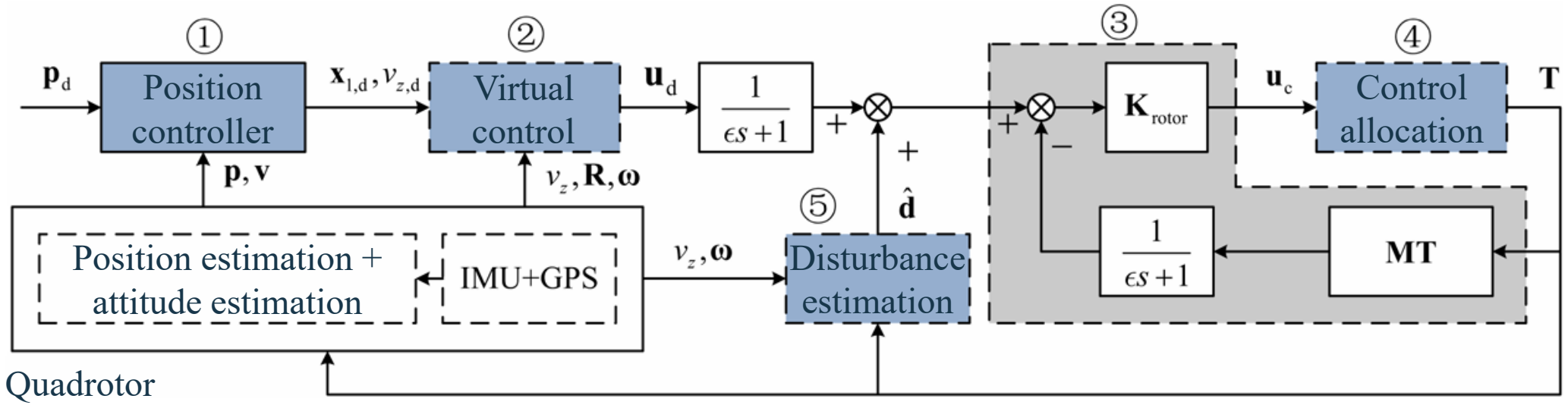
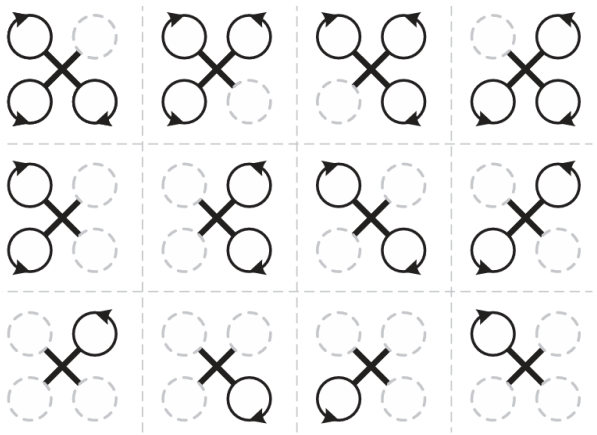


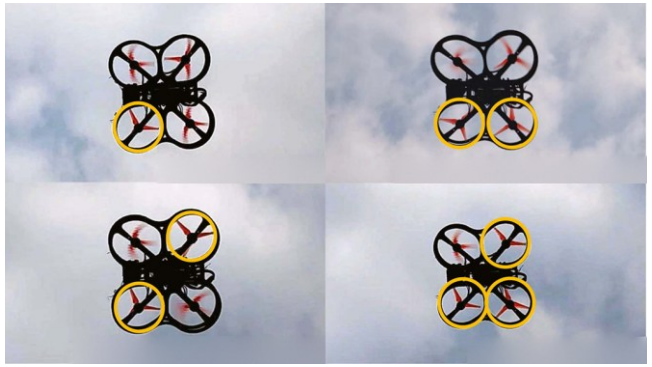
Figure 5.14(a) Enlarged fault-tolerant control block diagram

- Read: ① position control ② desired virtual control ③ K_{rotor} inner loop ④ control allocation ⑤ disturbance estimation.
- Disturbance estimation: add estimate $\hat{d}$ to the filtered desired virtual control.
- Failure cases: separately compare one, two same-side, two opposite-side, and three failed rotors.

5.7.4 Advanced Cases and Experiment Selection: Failure Conditions and Control Objectives

Remaining Rotor Geometry Determines Which Control Objectives Survive





One rotor | Two, same side
Two, opposite sides | Three rotors

- **Configuration:** distinguish positions and spin directions of remaining rotors.
- **Degradation:** give up independent yaw control under severe failures.
- **Evidence:** HITL and outdoor tests with specific parameters.

Left: rows show one-, two-, and three-rotor failures. Solid circles are remaining rotors (arrows show spin direction); dashed circles are failed rotors.

Figure 5.14(a) Remaining-rotor topology (left) and flight tests with four failure patterns (right)

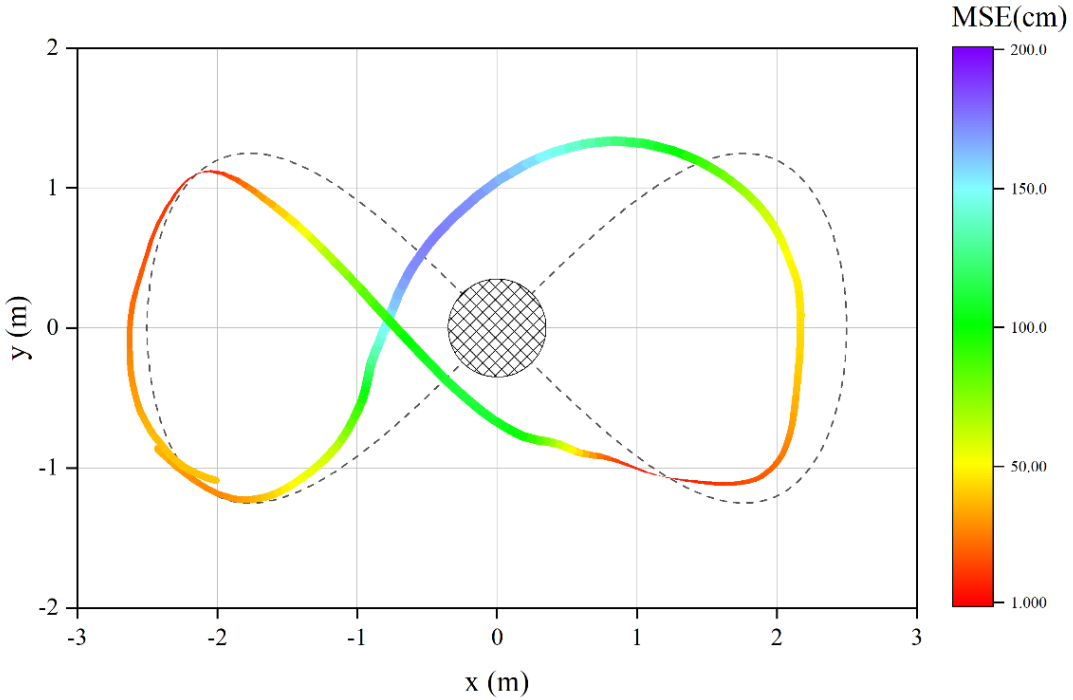
Paper: Ke C, Cai K-Y, Quan Q. Uniform passive fault-tolerant control of a quadcopter with one, two, or three rotor failure. IEEE T-RO, 2023, 39(6). https://doi.org/10.1109/TRO.2023.3297048

5.7 Other Vehicles and Advanced Cases

5.7.4 Advanced Cases and Experiment Selection: Reinforcement Learning Control

Policy Optimization Still Faces the Same Tracking and Constraint Tests

Dashed: commanded trajectory; colored: actual

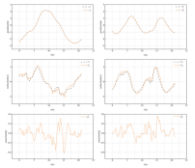


Coordinates: m; color scale MSE(cm): 1–200

Figure 5.14(b) Reinforcement learning control case

- **Extended state observer (ESO):** augment external disturbances and unmodeled dynamics as a state; estimate it with position and velocity.
- **Action:** horizontal acceleration correction.
- **Obstacle avoidance:** a control barrier function (CBF) adjusts action using the state and obstacles.

See "A Closer Look at Reinforcement Learning Evidence"





Experimental setting

5.7.4 Advanced Cases and Experiment Selection: A Closer Look at Reinforcement Learning Evidence

ESO Position and Velocity Estimates Stay Close to Actual Values

Dashed: actual values; orange: ESO estimates; horizontal axis: time (s), about 0–21 s

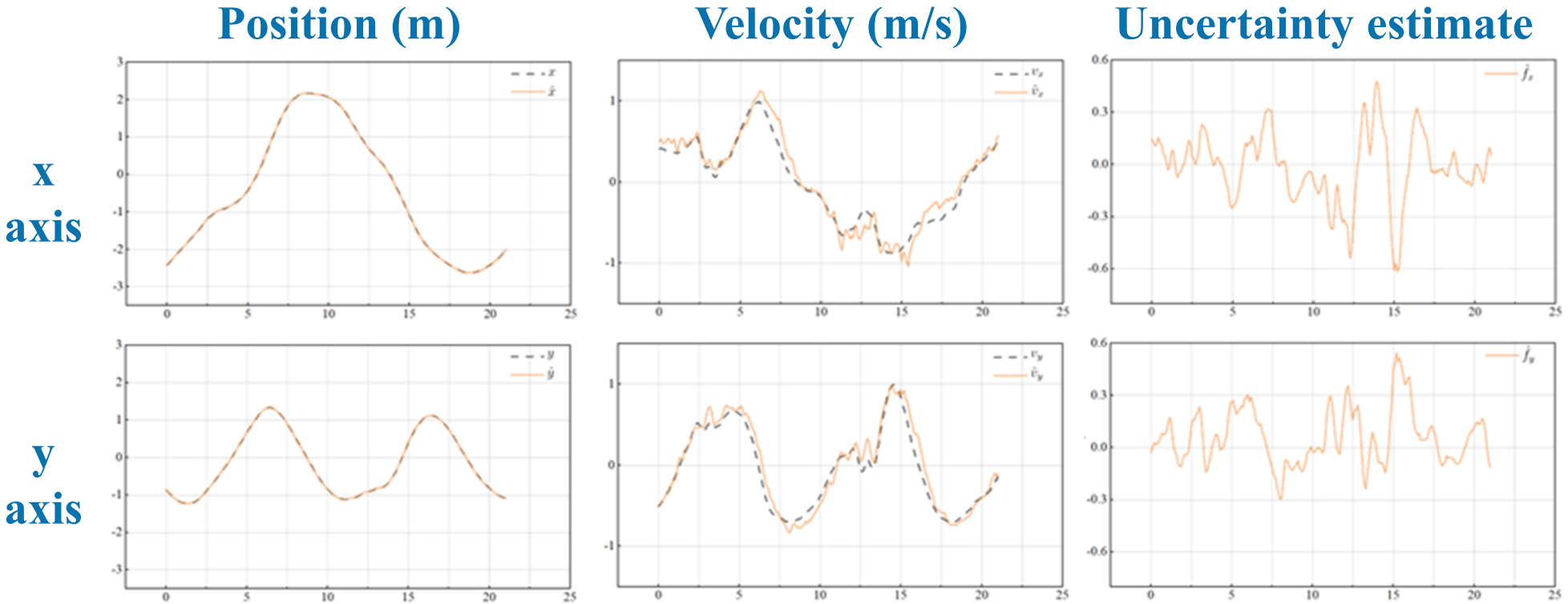


Figure 5.14(b) Position, velocity, and uncertainty estimates (rearranged into x and y rows)

5.7.4 Advanced Cases and Experiment Selection: States, Actions, and Cost in Reinforcement Learning

Tracking Cost Optimizes Actions; Safety Feedback Handles Obstacles Separately

Targets and states → **ESO estimates** → **Policy output** → **Safety correction** → **Limited acceleration**

x-axis states: position/velocity tracking errors; action u_x is a horizontal acceleration correction.

Error state

$$e_x = \begin{bmatrix} x - x_r \\ v_x - v_{x,r} \end{bmatrix}$$

Instantaneous cost: error and control effort

$$\ell_x = e_x^T \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} e_x + u_x^2$$

- **Policy evaluation:** the Critic estimates future accumulated cost under the current policy.
- **Policy improvement:** training updates the Actor using that evaluation; the Actor directly outputs actions at runtime.
- **Cost trade-off:** penalize tracking error and discourage excessive acceleration correction.
- **Safety correction:** CBF handles obstacle avoidance; apply limits before the inner loop.

5.7.4 Advanced Cases and Experiment Selection: Model-Compensation Control

Compensation Corrects Model Error and Disturbance Effects

Red solid: reference 0.9 m; black dash-dot: actual altitude
Time: 0–90 s; altitude ticks: 0.85, 0.90, 0.95 m

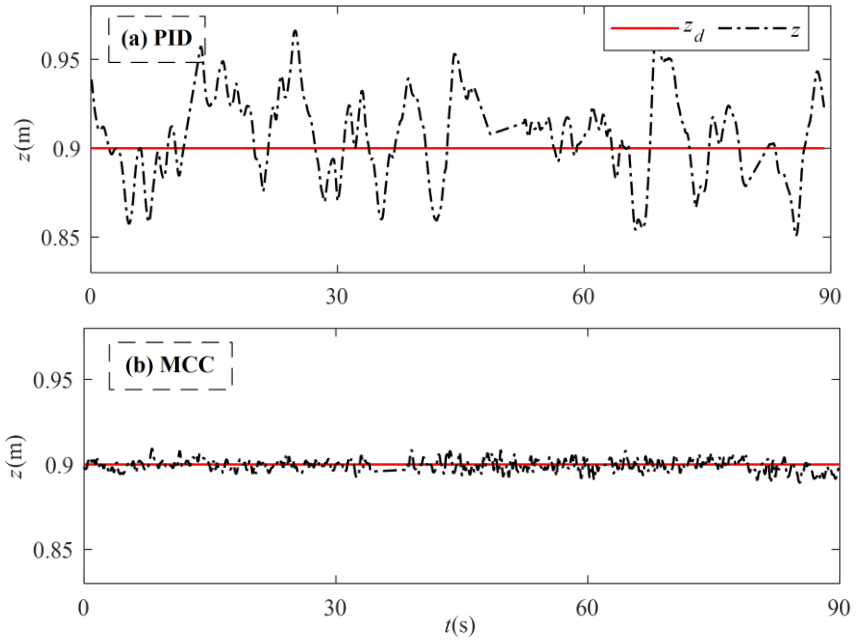


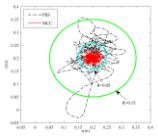
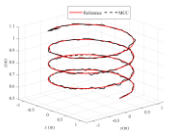
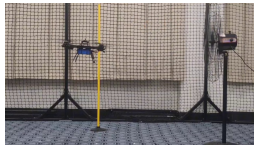
Figure 5.14(c) Altitude in wind: PID above, MCC below

Model-compensation control (MCC)
Instructional decomposition:

$$u = u_{\text{base}} + u_{\text{comp}}$$

u_{base}: baseline feedback control.
u_{comp}: from a nonlinear dynamic model and compensator, countering model error and disturbances.

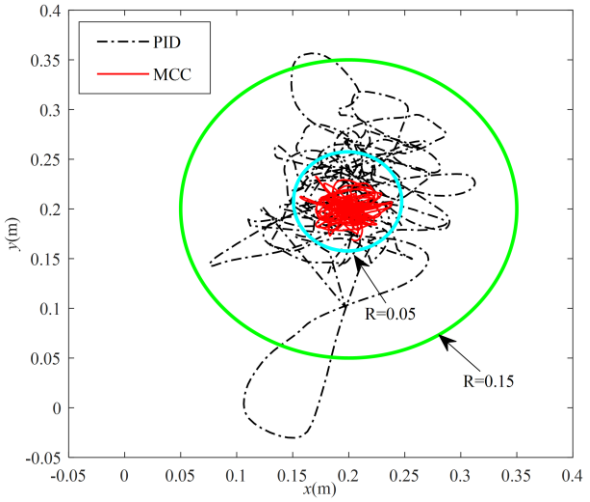
All three details: "A Closer Look at Model-Compensation Evidence"

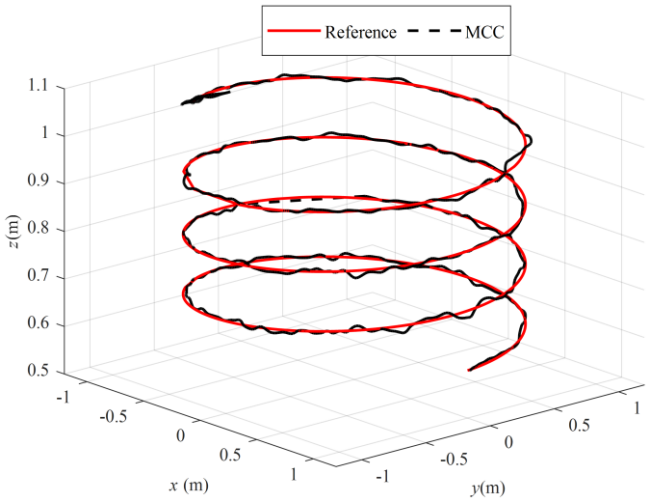




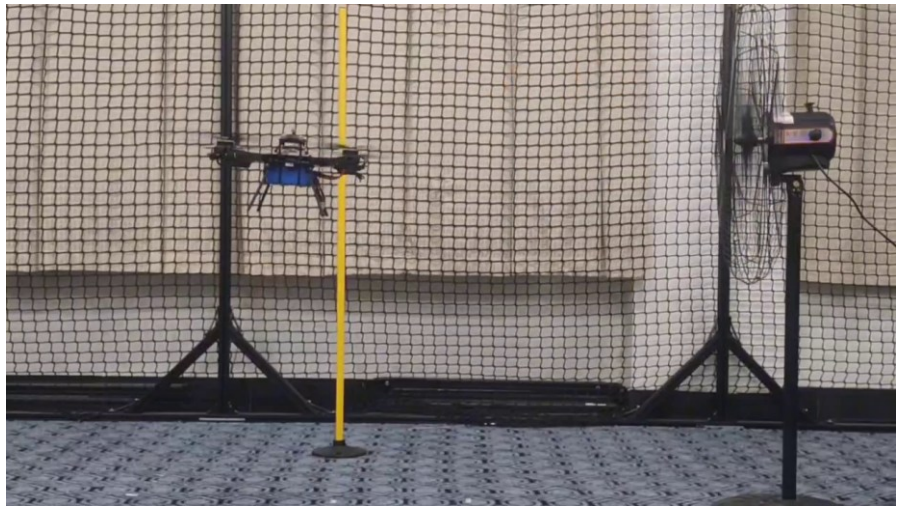
The sum remains subject to actuator limits.

5.7.4 Advanced Cases and Experiment Selection: A Closer Look at Model-Compensation Evidence

MCC Has a Tighter Position Spread and Tracks the Helix







Position hold: black dash-dot PID; red MCC

Helix: red solid reference, black dashed MCC

Wind: a fan blows sideways at the hovering multicopter

Figure 5.14(c) Model-compensation case: position hold, helical trajectory, and wind setup

Result under the plotted conditions: MCC position-hold traces stay inside the cyan circle (radius 0.05 m). PID repeatedly leaves it; a long lower segment and a short upper segment leave the green circle (0.15 m). MCC closely follows the helical reference.

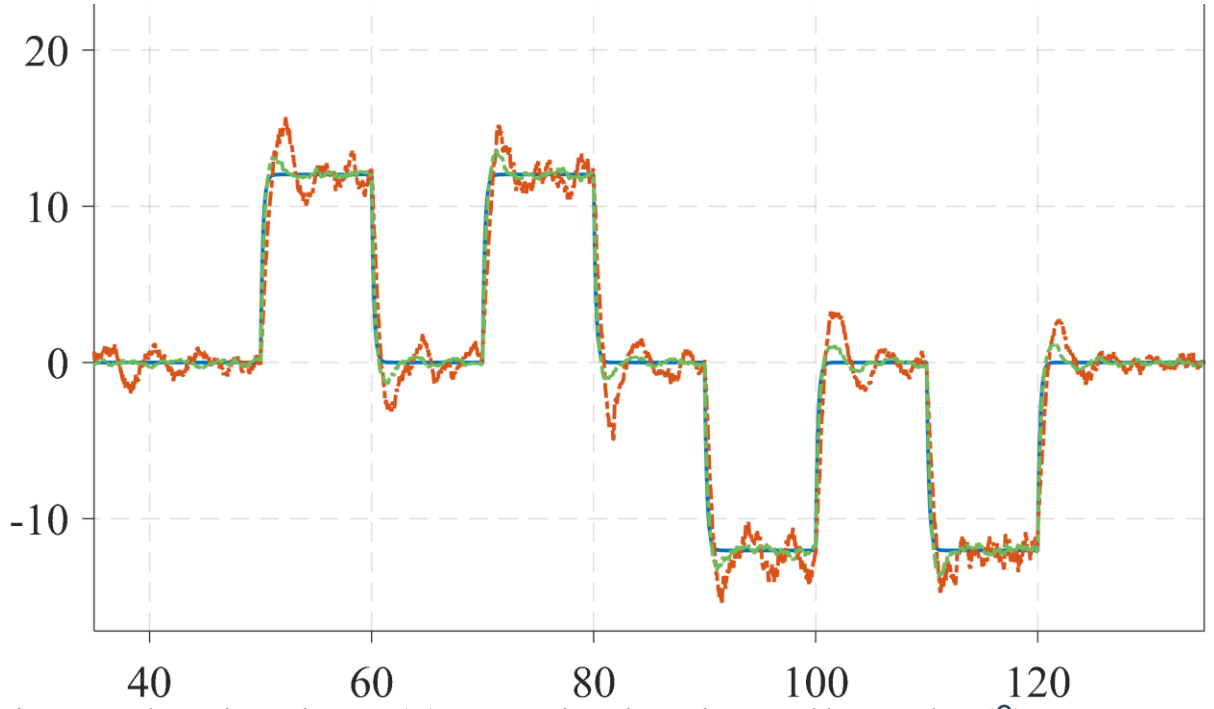
5.7 Other Vehicles and Advanced Cases

5.7.4 Advanced Cases and Experiment Selection: Helicopter Research Evidence

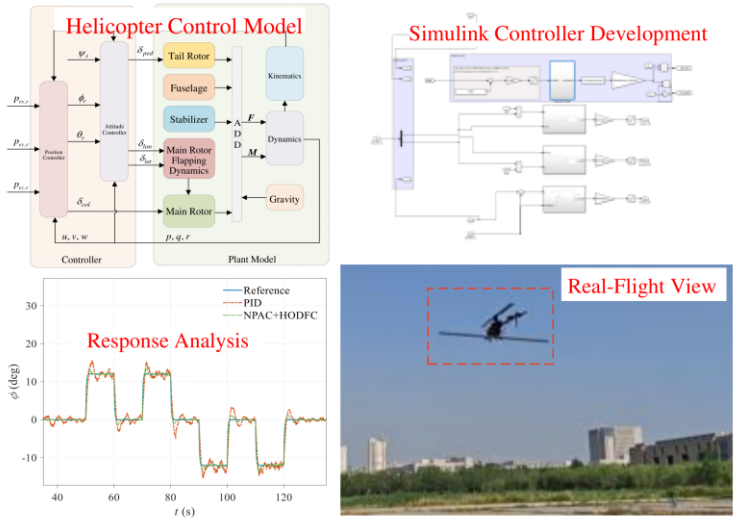
Models, Response Curves, and Flight Images Answer Different Questions

Blue solid: reference; orange dashed: PID

Green dash-dot: research controller, NPAC+HODFC



Horizontal axis: time (s); vertical axis: roll angle (°)



(d) Control Development of Small Unmanned Helicopters

Figure 5.14(d) Helicopter case

Read models/programs for states and coupling, responses for targets and errors, and flight images for context; use logs to assess performance.

5.7.4 Advanced Cases and Experiment Selection: Chapter Example Index: Interfaces

Find This Chapter's Interface Examples by Slide Section

Parent directory: [安装目录]/RflySimAPIs/5.RflySimFlyCtrl/; see Index.pdf for the complete experiment list.

Purpose / group	Subdirectory entry	Related slides and content
States and messages **0.ApiExps/**	11.StateDataGatAPI 6.uORB-Read-Write	5.5: raw and estimated quantities, timestamps
Control and logs **0.ApiExps/**	3.DesignExps · 5.Log-Write-Read 16.CtrlsSingalsAPI	5.1, 5.3, 5.5–5.6: control chain 5.4–5.6: log writing/reading
External targets **0.ApiExps/**	15.InputSourceAPI 17.OffboardCtrlsAPI	5.5: transmitter inputs; external targets in Offboard mode
Switching/tuning **0.ApiExps/**	21.SwitchPX4Simulink 10.QGC-Param-Tune	5.5–5.6: control-source switching 5.5: live parameter tuning

5.7.4 Advanced Cases and Experiment Selection: Chapter Example Index: Interfaces and Basic Functions

Communication, In-the-Loop Validation, and Basic Functions

Same parent directory; the code-generation experiment numbered 14 requires the full edition (see Index.pdf).

Purpose / group	Subdirectory entry	Related slides and content
Communication and validation 0.ApiExps/	8.Mavlink-Msg-Echo 14.SITLVeriGenCodeFirm	5.5: message echo 5.6: SITL validation
Design/estimation 1.BasicExps/	e1-FlightEval · e2-UavModeling e3-SensorCalib · e4-FilterDesign	Extensions: propulsion/modeling 5.1−5.2; 5.8, Exercise 7
Closed-loop control 1.BasicExps/	e5-AttitudeCtrl e6-PositionCtrl	Attitude: 5.1, 5.4; 5.8, Exercise 6 Position: 5.6
Modes/protection 1.BasicExps/	e7-SemiAutoCtrl e8-FailsafeLogic	5.6: target switching 5.8, Exercise 10: failsafe logic

5.7.4 Advanced Cases and Experiment Selection: Chapter Example Index: Advanced and Custom Experiments

Choose Examples by Problem; Read Version Conditions at Each Entry

3.CustExps and subexperiments marked in Index.pdf require the full edition. For firmware, see "Control Outputs Across Versions" in slide deck 5.5.

Topic / group	Subdirectory entry	Related slides and tasks
Modules and vehicle types • 1.BasicExps/	e9-ReplacePX4Module e10-FixedWingCtrl	5.5; 5.8, Exercise 9: single control source 5.7: airspeed/control surfaces
State feedback 2.AdvExps/	e5_LQR-CtrlExp/ 2.ModelID2LQRSimple	5.4: feedback from A/B distinguish place from lqr
Complex configurations 2.AdvExps/	e4_HELI-CtrlExp e6_Tiltrotor_CtrlExp	5.7: mechanism allocation two modes and handover
Research cases 3.CustExps/	e2_RL-CtrlExp e3_MCC-CtrlExp	5.7: reinforcement learning and model-compensation cases

5.7.4 Advanced Cases and Experiment Selection: Advanced Topics and Comparisons

One Main Change Needs a Comparable Set of Evidence

Branch	Main change	Common comparison and acceptance
Filtering	Time constant or observation weights	Same data segment; error, noise, and lag
Control law	Anti-windup / state feedback	Same model and limits; response and input
Allocation	Output limits / available actuators	Same request; residuals and limited channels
Vehicle transfer	Model, states, and control inputs	Same validation method; new conditions and coupling

5.7 Other Vehicles and Advanced Cases

Classroom Discussion

Can the Pitch Controller Be Reused as Airspeed Rises from 10 m/s to 20 m/s?

Scenario: the model on "One-Step Pitch Calculation" represents 10 m/s. Airspeed rises to 20 m/s; the control law, gains, and ±0.25 limits stay unchanged.

① Surface effectiveness

At the same deflection, how does the pitching moment change? What replaces effectiveness coefficient 4?

Observe: initial pitch acceleration

② Damping and response

How does damping term −2q change? What happens to closed-loop damping ratio and overshoot?

Observe: peak and overshoot

③ Scheduling and protection

What risks arise as airspeed falls? How should gains and protection be arranged?

Observe: angle of attack and surface saturation

Hint: use dynamic pressure from "Airspeed and Control-Surface Effectiveness", then compare with "Key Points".

Key Points

The Same Gains Remain Stable, but the Response Changes

Check	Derived result (instructional example)	Signals to observe
Surface effectiveness	Dynamic pressure 61.25→245 Pa; effectiveness coefficient 4→16 ($\propto V^2$)	Initial pitch acceleration
Damping and response	Damping coefficient −2→−4 ($\propto V$); damping ratio 0.8→1.1; overshoot disappears	Peak, overshoot, rise time
Scheduling and protection	Schedule gains by airspeed; retain angle-of-attack limits and minimum-airspeed protection at low speed	Airspeed, angle of attack, surface saturation

Characteristic equation: $s^2 + 3.2s + 4 \rightarrow s^2 + 8.8s + 16$; ω_n 2→4 rad/s; time to 90% is about 1.5→1.1 s.

Section Summary

Apply Transferable Methods to the New Plant and New Evidence

- **Vehicle transfer:** identify states and control inputs, then check directions, ranges, and coupled responses.
- **Fixed-wing:** airspeed ≠ ground speed. Airspeed changes both surface effectiveness and damping; redesign or schedule gains.
- **Complex configurations:** helicopter swashplates combine servo travel into collective and cyclic. Test output continuity separately during tiltrotor mode transitions.
- **Control and allocation:** match requests to actuator capability; check tracking, saturation, and allocation residuals.
- **Research cases:** compare against a common baseline; state conditions, data, costs, and failure limits.
- **Common errors:** common interface ≠ common algorithm; one operating-condition pass ≠ full-envelope validity.