INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 9 Model Predictive Control and Economic Optimization

CHAPTER 2

Hot-Air Balloon Simulator for Landing an Integrating Plant Behind a Dead Time

Pages 22–48

Edited by John D. Hedengren

How to cite this chapter. "Hot-Air Balloon Simulator for Landing an Integrating Plant Behind a Dead Time," in *Model Predictive Control and Economic Optimization*, John D. Hedengren, Ed., vol. 9 of *Interactive Control Engineering: Principles and Applications*. Advanced Process Solutions, LLC, 2026, ch. 2, pp. 22–48.

The page numbers in this file are those of the printed volume.

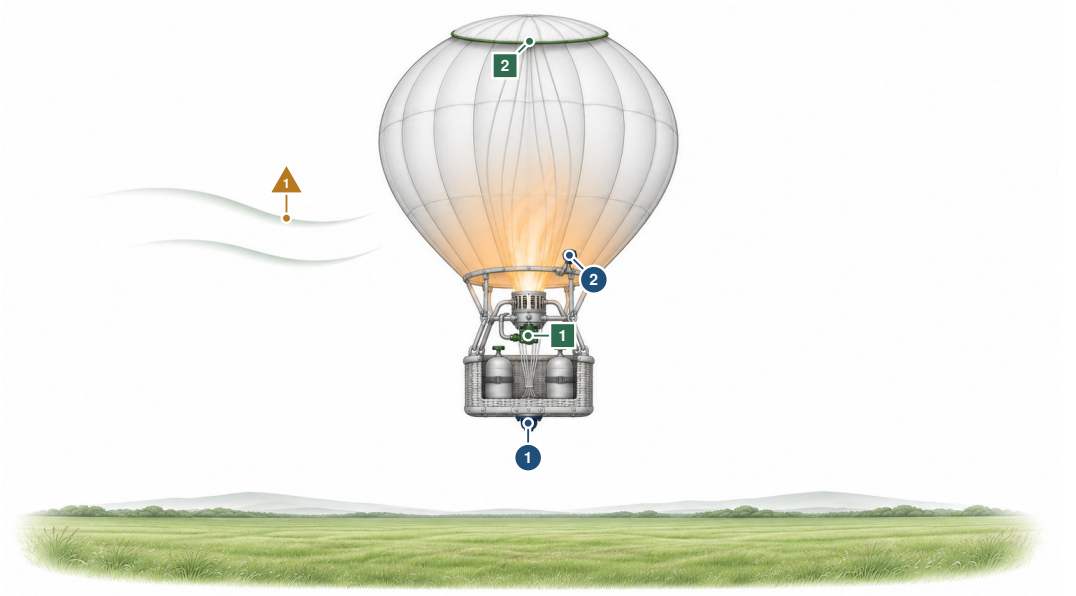
The app for this chapter
https://apmonitor.com/apps/hot-air-balloon/

The volume online: contents, other chapters and citation records
https://apmonitor.com/apps/books/v09-mpc-optimization/#hot-air-balloon

Copyright © 2026 Advanced Process Solutions, LLC. All rights reserved. Produced by Advanced Process Solutions, LLC.

CHAPTER 2

Hot-Air Balloon Simulator for Landing an Integrating Plant Behind a Dead Time



The controller moves 1 burner blast valve 2 parachute vent at the crown

The controller reads 1 altimeter and variometer 2 envelope temperature sensor

What disturbs it 1 wind

Notice that heat enters at the open mouth and is let out at the crown, and that the balloon answers its burner about half a minute late. Illustration, not to scale.

IN PLAIN TERMS Landing a Balloon That Answers Late

The main idea. A hot-air balloon rises when its burner heats the air inside it and sinks when that air cools or is let out through a vent at the top. A controller that looks ahead can plan the burns before the balloon needs them.

The challenge. The balloon starts to move only seconds after the flame lights, and then it keeps speeding up long after the burner stops. A gentle landing means heating while still high and falling fast, before the instruments show any need.

Why it matters. A pilot or controller that waits for the error to appear reaches the ground far too fast.

THREE WAYS TO READ THIS CHAPTER

Explore	12 min	the first page, the fact in Section 2.1, the history in Section 2.2, the first try-it in Section 2.4, the second try-it and the second fact in Section 2.6, and the key points
Study	55 min	the whole chapter with its three pause points and the twelve exercises at its end
Practice	20 min	the first page, the results (Section 2.6), Technology in Practice (Section 2.7), Limitations (Section 2.8) and the open challenge

The control question A hot-air balloon answers its burner four seconds late and then integrates twice on the way to the altitude, so there is no steady state from which to read a gain. What can one loop, a cascade and a step-response predictive controller do with such a plant, and what does it take to land it under a touchdown-speed limit?

Learning objectives

- Trace a blast-valve step through the dead time, the thermal integrator and the drag lag, and explain why no process gain can be read.
- Compute the float temperature, the lift per kelvin and the steady climb rate from the buoyancy and envelope energy balances.
- Explain from the phase of the plant why one PID loop on altitude hunts, and how a three-deep cascade gives each loop one slow element.
- Describe how the dynamic matrix controller identifies its step response, builds its gain matrix once and corrects a drift with a bias and a ramp.
- Choose a prediction horizon by where the identified model stops being true, and tell a model error from an aggressive optimizer.

Prerequisites Sections 1.6.2 (velocity form), 1.4.2 and 1.4.3 (step response and gain matrix), 1.4.4 (horizons and move suppression), 1.4.5 (the bias on an integrating plant) and 1.6.4 (how an exercise names a run).

The app https://apmonitor.com/apps/hot-air-balloon/

See also Volume 5, Chapter 2 for a burner whose fuel arrives behind a transport delay, and Volume 5, Chapter 3 for a predictive controller whose model omits part of the delay. Volume 3, Chapter 2 for matching the loop structure to where a disturbance enters, and Volume 3, Chapter 10 for another descent that must arrive slowly.

2.1 Introduction

The pilot of a hot-air balloon has one continuous way to change altitude, which is to change the temperature of the air held in the envelope. Opening the blast valve does not move the balloon and does not move the variometer. It lights a flame three meters tall inside an envelope twenty meters across, and the plume has to rise and mix before the mean temperature of the envelope changes at all. On the simulator of this chapter that takes 4 s. What follows the delay is worse than the delay. The envelope temperature is an integrator against a 646 s loss time constant, one kelvin of envelope temperature is worth 75.96 N of lift against 5356 kg of effective mass, quadratic drag turns that imbalance into a climb rate over about 33 s, and the altitude is the integral of the climb rate. A step on the blast valve moves the delivered heat at 4.0 s and the variometer past 0.5 m/s at 24.2 s, and two minutes later the balloon is 264 m higher and still accelerating.

> ✦ **DID YOU KNOW? Six to Fifteen Seconds Before the Balloon Answers**
>
> Flight training material warns that a pilot's input can take from 6 to 15 seconds to show as a reaction of the balloon, because the air inside the envelope is always moving [1]. In the simulator, the variometer passes 0.1 m/s at 13.0 s after a blast-valve step (Table 2.3), inside that range.

The shape of that plant is general. An output that integrates, an actuator that is both delayed and lagged, and a flight or a batch that ends on a hard limit rather than on a setpoint, appear together in ship steering, where rudder angle sets a yaw rate and heading is the integral [2], in building thermal control, where a heating plant with a long time constant drives a room whose temperature has no equilibrium of its own within the comfort band [3], and in every powered descent that must arrive at a fixed touchdown speed, such as the rocket of Volume 3, Chapter 10. An output that integrates behind a long delay is awkward enough that the standard dead-time compensator had to be reworked before it would hold such a plant at all [4], and each reach of the canal in Chapter 5 is such a plant, an integrator behind a wave travel time. The correct answers are structural rather than numerical.

Either the loop is broken into a cascade whose inner loop closes around the fast dynamics [5], or a predictive controller is given the delay and the integration explicitly [6, 7].

A block diagram states all of that and demonstrates none of it. Reading $Ke^{-\theta s}/s^2$ on a page does not produce the specific and counterintuitive experience of holding a burner open for five seconds from trim, seeing nothing happen, setting the valve back where it was, and then finding the balloon 586 m higher seven minutes after the burn began, 485 m higher than the same balloon left alone, because heat that has been bought cannot be given back. The misconception the plant corrects is that a manipulated variable set correctly holds the process where it is. Setting the burner to the heat input that exactly matches the fabric loss freezes the envelope temperature, and freezing the envelope temperature freezes whatever vertical speed the balloon already had. Trimming the actuator is not trimming the plant. Assessments of control education published over the past decade place experimental work with an unknown plant near the center of what a course should provide [8, 9], and a plant with no steady state to read a gain from is where that argument is strongest.

Pause. Try Exercise 2.1 before reading on.

Simulators for predictive control usually hide the controller behind a solver call, so the student sees a horizon and a tuning weight and never sees the free response being shifted, the bias being added or the constraint being enforced. This chapter works with a hot-air balloon simulator in which a step-response predictive controller is written out in full, from the step test that identifies it to the projected-gradient iteration that keeps its moves inside the actuator box, and it reports two results that the implementation forced. Section 2.2 traces balloon flight and its altitude control. Section 2.3 gives the nine-state flight model with its split-range actuator, its real transport delay and its terminal constraint on the arrival speed, with every equation and constant needed to rebuild it. Section 2.4 follows a blast-valve step through the chain of lags to the altitude, and Section 2.5 sets out the single loop, the cascade and the dynamic matrix controller that act on the same demand. Section 2.6 compares them on a level-off and on a graded landing that the single loop cannot pass at any gain, quantifies what a linear step-response model costs on a plant whose climb rate saturates as $\sqrt{\Delta T}$, and shows that a constant output bias is not offset-free when the output integrates. Section 2.7 describes the instruments, the burner and the platforms of real balloons and predictive controllers. Sections 2.8 to 2.10 state the limitations, the lessons that transfer and a summary. The general form of dynamic matrix control, its gain matrix and the failure of the bias on an integrating plant are set out in Sections 1.4.2 to 1.4.5, and this chapter keeps what is specific to the balloon.

2.2 Balloon Flight and the Control of Altitude

Manned flight began with a hot-air balloon [10], as the history segment below tells.

> **HISTORY 1783: Two Kinds of Balloon in One Year**
>
> The first recorded manned balloon flight was made on November 21, 1783, in a hot-air balloon built by the Montgolfier brothers of France. Pilâtre de Rozier and the Marquis d'Arlandes flew it for 23 minutes and 5.5 miles. Ten days later the physicist Jacques Charles flew the first gas balloon, a varnished silk envelope filled with hydrogen, for two hours and 27 miles, so the two kinds of balloon still flown today appeared in the same year. The modern hot-air balloon grew from a contract of the United States Navy in the 1950s for a small hot-air balloon for military purposes, a balloon the Navy never used [1]. Its propane burner, its fabric envelope and the vent in its crown are the actuators and the plant of this chapter.

What returned with it was an aircraft with two controls and no steering. The burner adds heat at a rate the pilot chooses, and the parachute vent, a disc in the crown of the envelope on a line to the basket, dumps hot air and lets cold air in. The two are never used together. Fuel is the binding resource, and the fuel a flight profile costs was already being computed from the envelope heat balance in the early lighter-than-air literature [11]. Gas balloons solved the same problem with a

different pair of actuators, a valve at the top that releases lifting gas and sand ballast over the side, and both of those are one-way and finite.

Altitude control on a manned balloon is still manual and still open loop in the strict sense. The instrument panel is an altimeter, a variometer, an envelope thermocouple at the crown with a fusible tell-tale, and a fuel gauge, and the technique taught in the certification literature is to anticipate, to burn in short bursts, and to fly the envelope temperature rather than the altitude [12]. The same source sets the standards the landing in this application is graded against, because an approach is flown to a target sink rate and a target ground speed rather than to a target position. The persistence of hand flying is not conservatism. It is a rational response to a plant on which a correction placed thirty seconds late is worse than no correction at all, made by an operator who can see the ground.

Automatic altitude control on balloons is real, current and unmanned. A superpressure balloon holds a fixed volume and trims its altitude with ballast, a pump moving air into and out of an internal ballonet, or a valve, and the loop that decides those actions runs for weeks without a pilot [13]. Stratospheric station-keeping goes further and uses altitude as the only steering authority available, choosing a height at which the wind blows toward the target, so a slow altitude loop becomes the inner loop of a navigation problem solved over days [14]. Both cases have the plant of this chapter, an integrating altitude behind a slow and asymmetric actuator, and both run the loop that the simulator asks the reader to design.

Physical modeling of balloon flight is a settled subject. Ascent and float codes integrate a vertical momentum balance with buoyancy, weight and quadratic drag against a standard atmosphere, and add an envelope energy balance carrying solar input, infrared exchange and the thermal mass of the film [15]. The added mass of the displaced air, which for a sphere in potential flow is half the displaced volume, is part of that balance and dominates the time it takes a climb rate to establish [16]. This application keeps the momentum and energy balances, replaces the radiation terms with a single lumped loss coefficient because a sport flight lasts an hour rather than a day, and adds the burner transport delay that the ascent codes have no reason to carry.

2.3 Flight Model

The vehicle is an AX-8 sport balloon of 3000 m^3 (106000 cubic feet) carrying 640 kg of envelope, basket, burner, tanks and four occupants, 77 kg of propane and 40 kg of sand ballast, for an all-up mass of 757 kg. Figure 2.1 shows the arrangement and labels every symbol used below. Nine states are integrated. Three of them carry the lesson, which are the altitude z, the climb rate v_z and the envelope temperature T_{env}, and the remaining six are the burner delivery position, the vent position, the propane and ballast masses and the horizontal position and speed.

The controlled, manipulated and disturbance variables are set out in Table 2.1. The clock is in seconds, altitudes in meters, temperatures in degrees Celsius except where the gas law needs kelvin, and both actuators in percent of full travel.

2.3.1 Vertical Momentum and the Added Mass

Altitude and climb rate follow Newton's second law with buoyancy, weight and quadratic drag, as shown in Equations (2.1) and (2.2).

$$\frac{\mathrm{d}z}{\mathrm{d}t} = v_z \tag{2.1}$$

$$m_{eff}\,\frac{\mathrm{d}v_z}{\mathrm{d}t} = (\rho_a V - m_{tot})\,g - \tfrac{1}{2}\rho_a C_D A\,(v_z - w)\,|v_z - w| \tag{2.2}$$

where $m_{tot} = m_{dry} + m_f + m_b + \rho_{env} V$ is everything the balloon carries including the air inside it, ρ_a is the ambient air density, V is the envelope volume, C_D is the drag coefficient on frontal area A, and w is the vertical motion of the air itself. Both sides carry newtons. The mass on the left is not m_{tot}, as shown in Equation (2.3).

$$m_{eff} = m_{tot} + k_a\,\rho_a V \tag{2.3}$$

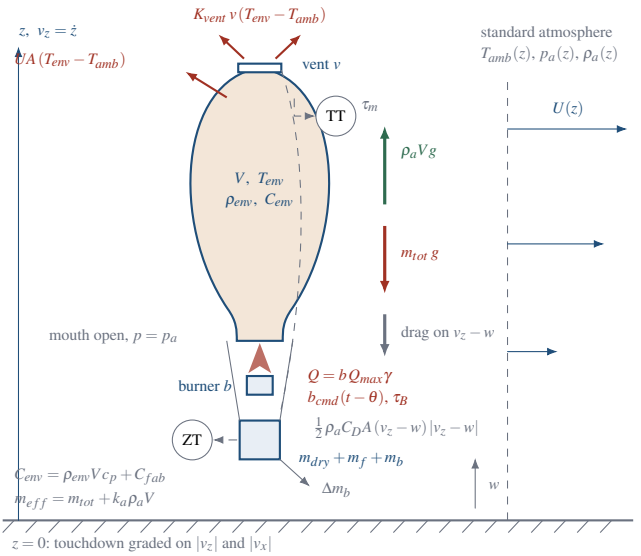


Figure 2.1 The AX-8 balloon of the simulator with the states, the two actuators, the forces and the heat flows. The envelope of volume V holds air at T_{env} and density ρ_{env} with total thermal capacity C_{env}, is open at the mouth and therefore at ambient pressure p_a, and loses heat through the fabric as $UA(T_{env}-T_{amb})$ and through the crown vent of opening v as $K_{vent}v(T_{env}-T_{amb})$. Burner fraction b follows the command delayed by θ with lag τ_B and delivers $bQ_{max}\gamma$. Net force is buoyancy $\rho_a V g$ less weight $m_{tot}g$ less quadratic drag on the velocity v_z-w relative to the air, accelerating the effective mass m_{eff} that includes the added mass $k_a\rho_a V$. TT is the crown thermocouple with lag τ_m and ZT is the altimeter and variometer.

Table 2.1 Classification of Variables. Nominal Values Are Those of the Design Point at 300 m Over a 15 °C Morning.

Symbol	Role	Nominal	Range
z	CV, measured	300 m	0–1500 m SP
v_z	inner CV, measured	0 m/s	±2.5 m/s envelope
T_{env}	inner CV, measured	90.06 °C	T_{amb}–120 °C
u (demand)	MV	12.65 %	−100–100 %
b (burner)	MV, split range	12.65 %	0–100 %
v (vent)	MV, split range	0 %	0–100 %
T_{sl}	DV, measured	15 °C	−10–38 °C
U_{10}	DV, measured	0 m/s	0–12 m/s
w	DV, measured	0 m/s	±3 m/s
ΔT_{inv}	DV, measured	0 K	0–12 K
Δm_b	DV, discrete	—	20 kg twice
m_f	state, measured	77 kg	0–77 kg
b_{pos}, v_{pos}	states	0.1265, 0	0–1
x, v_x	states, measured	0, $U(z)$	—

where k_a is the added-mass coefficient, taken as the potential-flow value of 0.5 for a sphere. At the design point $k_a\rho_a V = 1785$ kg against 3570 kg of balloon and envelope air, so a third of what has to be accelerated is the air being pushed out of the way [16]. Linearizing the drag term about a climb rate $|v_z|$ gives a first-order approach with time constant $m_{eff}/(2k_d|v_z|)$ where $k_d = \frac{1}{2}\rho_a C_D A$, which is 32.7 s at 1 m/s. The added mass supplies 10.9 s of that.

2.3.2 The Envelope Energy Balance

The envelope is open at the mouth and is therefore at ambient pressure, so its internal energy pVc_v/R does not change and every joule delivered leaves again as the enthalpy of air spilling out of the bottom. Writing that balance out and adding the thermal mass of the fabric gives Equation (2.4), which has the form of a closed vessel holding the air currently inside.

$$\begin{aligned} C_{env}\frac{\mathrm{d}T_{env}}{\mathrm{d}t} &= b\,Q_{max}\,\gamma-(UA+K_{vent}\,v)\,(T_{env}-T_{amb}) \\ C_{env} &= \rho_{env}Vc_p+C_{fab} \end{aligned} \tag{2.4}$$

where b and v are the delivered burner and vent fractions, Q_{max} is the burner rating, UA is the fabric loss coefficient, K_{vent} is the additional loss at a fully open vent, c_p is the specific heat of air and γ is the fuel-available factor of Equation (2.5). Both sides carry watts. At the design point the capacity is 2.972 MJ/K, of which the nylon supplies 145 kJ/K, or 4.9 %, and it is the part that keeps heating after the valve shuts. The ratio C_{env}/UA is 646 s, so over any maneuver Equation (2.4) behaves as an integrator. The two halves of the actuator are not gain matched. Full burner against the design-point loss gives $+0.82\,\mathrm{K/s}$ and a fully open vent gives $-1.41\,\mathrm{K/s}$, a ratio of 1.72, and that asymmetry is retained because it is what a parachute vent does and it is what makes one linear controller behave differently going up and coming down.

Propane is consumed in proportion to the delivered heat and the burner dies smoothly as the last half kilogram goes, as shown in Equation (2.5).

$$\frac{\mathrm{d}m_f}{\mathrm{d}t}=-\frac{b\,Q_{max}\,\gamma}{\eta_b\,\mathrm{LHV}},\qquad \gamma=\min\left(\max\left(\frac{m_f}{0.5\,\mathrm{kg}},0\right),1\right) \tag{2.5}$$

where η_b is the burner efficiency and LHV is the lower heating value of propane. The smooth gate keeps the four stages of the fixed-step integrator from straddling a discontinuity. At the design point Equation (2.5) runs at 29.9 kg/h, which is 8.3 g/s of load leaving the aircraft continuously and is the reason the float temperature drifts all flight.

2.3.3 Actuator Dynamics and the Transport Delay

Both actuators are first order and the burner carries a pure delay in front of its lag, as shown in Equation (2.6).

$$\tau_B\frac{\mathrm{d}b}{\mathrm{d}t}=b_{cmd}(t-\theta)-b,\qquad \tau_V\frac{\mathrm{d}v}{\mathrm{d}t}=v_{cmd}-v \tag{2.6}$$

where $\theta=4\,\mathrm{s}$ is the plume rise and mixing time, $\tau_B=0.8\,\mathrm{s}$ is the delivery lag and $\tau_V=1.0\,\mathrm{s}$ is the vent travel time. The delay is applied to the command through a ring buffer with linear interpolation, written at the controller period and read θ seconds back, so the right-hand side that the integrator evaluates four times per step stays a pure function of the state. The delay is set in steps of 0.5 s, a whole number of controller periods, so the delivered command is exactly the command issued θ earlier, held for one controller period. Delaying the command is equivalent to delaying the delivered heat and keeps the delay line out of the derivative evaluation.

2.3.4 Split Range and the Descent Envelope

Every architecture manipulates one number, the heat demand $u\in[-1,+1]$, which is split into the two actuators as shown in Equation (2.7).

$$b_{cmd}=\max(u,0),\qquad v_{cmd}=\max(-u,0) \tag{2.7}$$

so the burner and the vent can never be open at once. Using one manipulated variable is what makes the four architectures comparable. The vent also accepts a floor from the override described in Section 2.5. The climb rate the guidance is permitted to ask for is bounded by Equation (2.8).

$$|v_z|\le\min\left(v_{max},\,k_z z+v_{flare}\right) \tag{2.8}$$

where v_{max} is the cruise limit, k_z is the slope of the descent envelope and v_{flare} is what the envelope allows at the ground. Above $z = (v_{max} - v_{flare})/k_z$, which is 85 m at the default settings, Equation (2.8) is a flat cruise limit. Lower down the permitted speed falls with height, which converges the sink rate onto v_{flare} over a time scale of $1/k_z = 50\,\mathrm{s}$. The profile sets the requested terminal rate. Thermal delay and tracking error still affect the physical touchdown speed, so the terminal safety requirement must be checked against the measured arrival and not inferred from the guidance limit alone [17].

2.3.5 Atmosphere, Wind and Buoyancy Algebra

The column is a standard lapse rate applied to the day's sea-level temperature, as shown in Equation (2.9).

$$\begin{aligned} T_{amb} &= T_{sl} - Lz + \Delta T_{inv}\, \mathrm{e}^{-((z-z_{inv})/w_{inv})^2} \\ p_a &= p_0 \left(1 - \frac{Lz}{T_{sl}}\right)^{g/RL} \end{aligned} \tag{2.9}$$

with $\rho_a = p_a/(R\,T_{amb})$, where L is the lapse rate, R is the gas constant of air and ΔT_{inv} is an optional inversion of thickness w_{inv} centered at z_{inv}. The inversion is added to the temperature profile only and is not made hydrostatically consistent with the column above it. At 300 m over a 15 °C morning Equation (2.9) gives 13.05 °C, 97773 Pa and 1.19012 kg/m^3. Horizontal wind follows a logarithmic profile and vertical air motion is tapered to zero at the surface, as shown in Equation (2.10).

$$U(z) = U_{10}\,\frac{\ln((z+z_0)/z_0)}{\ln(z_{ref}/z_0)}, \qquad w = w_0\left(1 - \mathrm{e}^{-z/z_{th}}\right) \tag{2.10}$$

where z_0 is the surface roughness, z_{ref} is the reference height and z_{th} is the height over which a thermal dies out at the ground. The amplitude w_0 is the steady vertical air motion plus a thermal or a patch of sink of strength w_g that decays with time constant τ_g once the balloon has entered it. The horizontal channel is one quadratic drag equation on the same effective mass, $m_{eff}\dot{v}_x = \frac{1}{2}\rho_a C_D A(U(z) - v_x)|U(z) - v_x|$ and $\dot{x} = v_x$, and it feeds nothing back into the vertical channel. It exists because the touchdown verdict depends on ground speed as well as sink rate.

The envelope density is $\rho_{env} = p_a/(R\,T_{env})$ in kelvin, and the temperature that exactly floats a given load follows from neutral buoyancy without iteration, as shown in Equation (2.11).

$$T_{float} = \frac{p_a}{R\,(\rho_a - m_{load}/V)}, \qquad m_{load} = m_{dry} + m_f + m_b \tag{2.11}$$

Equation (2.11) is the reference every architecture in the application is really tracking, and the application draws it as a trace beside the envelope temperature. At the design point it returns 90.06 °C against a 120 °C fabric limit. Differentiating the buoyancy with respect to envelope temperature and equating the resulting force to quadratic drag gives the climb characteristic of Equation (2.12).

$$\frac{\partial F}{\partial T_{env}} = \frac{\rho_{env} V g}{T_{env}}, \qquad v_{z,ss} = \mathrm{sgn}(\Delta T)\sqrt{\frac{\left|\frac{\partial F}{\partial T_{env}}\Delta T\right|}{\frac{1}{2}\rho_a C_D A}} \tag{2.12}$$

where $\Delta T = T_{env} - T_{float}$. At the design point $\partial F/\partial T_{env} = 75.96\,\mathrm{N/K}$ against $k_d = 81.82\,\mathrm{N}$ per $(\mathrm{m/s})^2$, so one kelvin off float buys 0.96 m/s of steady climb, four kelvin buys 1.93 m/s and sixteen buys 3.85 m/s. The square root in Equation (2.12) is the source of most of the difficulty later, because it makes the balloon much easier to start than to stop and it is the term a linear model cannot carry.

2.3.6 Constants, Trimmed Start and Integration

Table 2.2 lists the constants. Eight are hidden and are never displayed, and they are the physical constants and the surface-layer constants that no scenario asks the reader to recover. Everything else is visible in the model panel and adjustable.

Table 2.2 Model Parameters. Hidden Constants Are Never Displayed. Every Visible Constant Is Also Adjustable From the Model Panel.

Symbol	Meaning	Value	Unit	Visibility
V	envelope volume	3000	m^3	visible
m_{dry}	dry all-up mass	640	kg	visible
m_{f0}	propane on board	77	kg	visible
m_{b0}	ballast on board	40	kg	visible
C_{fab}	fabric heat capacity	145	kJ/K	visible
C_D	drag coefficient	0.55	—	visible
A	frontal area	250	m^2	visible
k_a	added-mass coefficient	0.50	—	visible
Q_{max}	burner rating	2.80	MW	visible
θ	burner transport delay	4.0	s	visible
τ_B	heat delivery lag	0.8	s	visible
τ_V	vent travel time	1.0	s	visible
UA	envelope heat loss	4600	W/K	visible
K_{vent}	vent loss, full open	50000	W/K	visible
T_{fab}	fabric limit	120	°C	visible
τ_m	crown thermocouple lag	2.0	s	visible
L	lapse rate	0.0065	K/m	visible
z_{inv}	inversion altitude	400	m	visible
w_{inv}	inversion thickness	120	m	visible
z_{th}	thermal cut-off height	60	m	visible
w_g	thermal strength	2.0	m/s	visible
τ_g	thermal decay time	40	s	visible
v_{td}	stand-up touchdown limit	1.5	m/s	visible
v_{hard}	hard-landing speed	3.0	m/s	visible
u_{drag}	drag-out ground speed	4.0	m/s	visible
Δm_b	ballast release size	20	kg	visible
LHV	propane heating value	46.4	MJ/kg	hidden
η_b	burner efficiency	0.92	—	hidden
p_0	sea-level pressure	101325	Pa	hidden
R	gas constant of air	287.05	J/(kg K)	hidden
c_p	specific heat of air	1005	J/(kg K)	hidden
g	gravity	9.80665	m/s^2	hidden
z_0	surface roughness	0.10	m	hidden
z_{ref}	wind reference height	10	m	hidden

Every scenario opens trimmed. The altitude is set to the target, the envelope temperature to T_{float} of Equation (2.11) clipped to the fabric limit, the climb rate to zero, and the burner position to the opening that matches the fabric loss exactly, as shown in Equation (2.13).

$$b(0) = \frac{UA\left(T_{float} - T_{amb}\right)}{Q_{max}} \tag{2.13}$$

which returns 12.65 % at the design point, or 354 kW of burner against 354 kW of fabric loss. On a day too hot to float the load the state opens at the fabric limit and the balloon sits on the ground. The engine of Section 1.6.1 integrates with classical fourth-order Runge-Kutta on a fixed 0.05 s step, the controllers execute every 0.5 s, the predictive controller executes on its own 2 s subclock and the historian logs five samples per simulated second. The model draws no random numbers, so a scenario replays identically. Touchdown is detected when z reaches zero, the rigid body is then frozen, and the arrival is graded as a stand-up landing below 1.5 m/s, firm between 1.5 and 3.0 m/s, hard above that, and dragged when the ground speed of an arrival that is not hard exceeds 4 m/s.

2.3.7 Assumptions

Every simplification is given below beside the condition that defeats it.

The envelope is one lumped air mass at one temperature. A real envelope is stratified by twenty kelvin or more between crown and mouth, and it is the crown that reaches the fabric limit first, so the overtemperature warning here is optimistic about where the damage happens and approximately right about when.

The transport delay is a fixed 4 s. In a real balloon it depends on the burn rate, on the envelope size and on how much cold air the plume entrains, and it is shorter at full burn than at a trickle. Four seconds is a plausible figure for 3000 m^3, it is the number the whole application turns on, and it is the one an instrumented balloon should be used to measure.

Buoyancy is computed at a constant envelope volume. A real envelope breathes through its open mouth, but once inflated the shape is set by the load tapes and the volume error is under a percent.

Vertical drag is one quadratic term with textbook sphere constants. Neither $C_D = 0.55$ on 250 m^2 nor $k_a = 0.5$ has been measured on a balloon here, and k_a in particular sets how long a climb rate takes to establish.

Radiation is absorbed into UA. Solar gain on the envelope, infrared exchange with the sky and ground, and the diurnal cycle that dominates a long-duration flight are all absent [15], which is defensible for a one-hour sport flight and not for a stratospheric one.

The inversion is a temperature anomaly only. Making it hydrostatically consistent with the column above would change the density inside the layer by well under half a percent at 8 K and 400 m, and would not change the sign of anything.

Horizontal motion carries no direction, no turbulence and no navigation. The drag-out verdict at touchdown is therefore about speed alone, and the real steering mechanism of a balloon, which is choosing an altitude whose wind points somewhere useful, is absent.

The numerics repeat exactly. A fixed step and a fixed controller period reproduce one trajectory on every machine, which is convenient for grading and is not what a real flight computer does.

2.4 The Chain From Blast Valve to Altitude

Scenario 2 is a clean open-loop step of 20 % on the blast valve from the trimmed condition, held. Table 2.3 reports what the application measures, with crossing times read at the 0.05 s physics step. The delivered heat begins to move 4.0 s after the step, which is the transport delay alone, and then rises through the 0.8 s delivery lag. The envelope then heats at up to 0.187 K/s, close to the 0.188 K/s of 560 kW into 2.972 MJ/K, and does not slow measurably over two minutes because the 646 s loss constant has barely begun to act. The variometer passes 0.1 m/s at 13.0 s, 0.5 m/s at 24.2 s and 1.0 m/s at 33.2 s, and at 120 s the envelope is 20.5 K hotter, the balloon is 264.0 m higher, and it is still accelerating. The unstepped balloon climbs as well, by 6.5 m over the same 120 s, because it is burning 8.3 g/s of its own load, so 257.5 m of the gain belongs to the step.

> **TRY IT IN THE APP 2 · The Step Test the MPC Is Built From**
>
> Load the scenario and press Run. Watch the altitude. The blast valve steps up at 20 s, and for several seconds nothing moves. Two minutes after the step the balloon is 264.0 m higher and still accelerating, with no level at which a process gain could be read.

There is no steady state in Table 2.3 from which to read a process gain. Strictly the path from demand to altitude carries one pole at the origin, because altitude is the integral of climb rate, together with a delay of 4 s and three real poles at $1/\tau_B = 1.25/\text{s}$, at $UA/C_{env} = 1.55 \times 10^{-3}/\text{s}$ and at $2k_d|v_z|/m_{eff} = 3.1 \times 10^{-2}/\text{s}$ for a climb of 1 m/s. The thermal pole is so much slower than any maneuver that the plant behaves as a double integrator over the horizons that matter, which is what Table 2.3 shows. Evaluating that transfer function at $\omega = 0.01/\text{s}$, a period of about ten minutes, gives a phase of $-192°$, made up of $-90°$ from the integrator, $-81°$ from the thermal pole, $-18°$ from the

Table 2.3 Open-Loop Response to a 20 % Blast-Valve Step From the Trimmed Design Point, Measured on the Shipped Model. Times Are After the Step. Altitudes in Parentheses Subtract the Unstepped Run.

Quantity	Value
Delivered heat leaves its pre-step value	4.0 s
Initial envelope heating rate	0.187 K/s
Variometer reaches 0.1 m/s	13.0 s
Variometer reaches 0.5 m/s	24.2 s
Variometer reaches 1.0 m/s	33.2 s
Variometer reaches 2.0 m/s	49.8 s
Altitude gained at 30 s	7.2 m (6.9)
Altitude gained at 60 s	58.2 m (57.0)
Altitude gained at 120 s	264.0 m (257.5)
Envelope temperature rise at 120 s	20.5 K

drag lag and only -2.3° from the delay. The plant on its own is already past -180° there, and at a period of five minutes it is at -214°. A single proportional-integral-derivative loop from altitude error to burner position has to buy all of that phase back itself, and the derivative term is the only one that can see a climb rate while looking through the delay. Removing the derivative adds a second pole at the origin from the reset term, which takes the open loop to -282° at the same frequency before any lead is credited back, and the measured consequence is in Section 2.6.

The asymmetry compounds this. Descending is easy, because the vent removes heat at $-1.41\,\mathrm{K/s}$ and every kelvin removed is bought back at $+0.82\,\mathrm{K/s}$ through four seconds of dead time. The square root in Equation (2.12) then means that arresting a 3 m/s descent requires recovering 9.7 K while arresting a 1 m/s descent requires only 1.1 K. A controller that holds altitude well in cruise can still arrive at the ground much too fast, and that is what the challenge scenario grades.

2.5 Controllers

Four architectures manipulate the same demand u of Equation (2.7). The demand is slew limited at 50 %/s and clipped to $[-1, +1]$ after every architecture, and once the balloon is on the ground the demand is forced to -1 with the vent fully open, which is what a crew does and which keeps the logged actuators inside their declared limits.

2.5.1 Manual and the Single Loop

Manual mode exposes the blast valve and the vent as sliders, with the B key held for full burner, which is how the aircraft is actually flown. Every PID loop of the application is the velocity-form algorithm of Section 1.6.2, Equations (1.14) to (1.17), executed every $\Delta t_c = 0.5\,\mathrm{s}$ with the derivative taken on the measurement and back calculation against the loop's own output limits with tracking time $T_t = \tau_I$. On the single loop the measurement passes through a first-order filter of 3 s before the derivative sees it. The demand slew limit is applied after every architecture and is not fed back to any loop, so each back calculation acts only against its own clip. In manual the integral is set so that the output equals the operator's slider, and every loop not currently driving the actuator is tracked in the same way, so transfer is bumpless in both directions and changing the architecture in flight moves nothing.

The single loop drives altitude error straight onto the blast valve with $K_c = 0.06\,\%/\mathrm{m}$, $\tau_I = 1200\,\mathrm{s}$ and $\tau_D = 130\,\mathrm{s}$, which is as well as one loop can be tuned on this plant, and the vent is handled by the high-altitude override of Equation (2.14).

$$v_{ovr} = \mathrm{clip}\left(\frac{z - z_{\mathrm{SP}} - z_{db}}{z_{span}},\ 0,\ 1 \right) \qquad u = -v_{ovr} \quad \text{whenever } v_{ovr} > 0 \tag{2.14}$$

with a deadband $z_{db} = 40\,\mathrm{m}$ and a span $z_{span} = 60\,\mathrm{m}$. The override takes the burner away from the loop entirely while it acts, and the loop is not informed, so its back calculation still sees only its own clip to $[0, 1]$. That is the behavior that puts the balloon in a field on the graded landing of Section 2.6, where the override acts on 146 of the loop's 215 executions before touchdown.

2.5.2 Cascade

The cascade breaks the plant at each slow element. The altitude loop produces a climb-rate setpoint, the climb-rate loop produces an envelope-temperature setpoint, and the temperature loop drives the demand, as shown in Equation (2.15).

$$\begin{aligned} v_{z,\mathrm{SP}} &= \mathrm{clip}(\mathrm{PI}_z(z_{\mathrm{SP}} - z),\ -v_{lim},\ +v_{lim}) \\ T_{\mathrm{SP}} &= \mathrm{clip}\big(T_{float} + \mathrm{PI}_v(v_{z,\mathrm{SP}} - v_z),\ T_{amb} + 2,\ T_{fab} - 1\big) \\ u &= \mathrm{clip}\big(b_{trim} + \mathrm{PID}_T(T_{\mathrm{SP}} - \hat{T}_{env}),\ -1,\ +1\big) \end{aligned} \tag{2.15}$$

where v_{lim} is the descent envelope of Equation (2.8), $\hat{T}_{env}$ is the crown thermocouple through its 2 s filter, and the two bias terms are feedforwards. The middle loop is biased by the float temperature of Equation (2.11), so it begins from the answer the ideal-gas law already gives and corrects only what the model misses. The inner loop is biased by the trim burner of Equation (2.13) evaluated on the measured temperature. Default gains are $K_{c,z} = 1.6\,(\mathrm{m/s})$ per 100 m with $\tau_{I,z} = 1200\,\mathrm{s}$, $K_{c,v} = 3.0\,\mathrm{K}$ per m/s with $\tau_{I,v} = 90\,\mathrm{s}$, and $K_{c,T} = 12\,\%/\mathrm{K}$ with $\tau_{I,T} = 60\,\mathrm{s}$ and no derivative.

Three choices in Equation (2.15) deserve their reasons. The general rule, that a loop rejects the disturbances entering inside its own measurement path, is the subject of Volume 3, Chapter 2. The inner loop is fast because it sees a single 646 s lag with a steady-state gain of 609 K per unit of demand through only a 4 s delay. The middle loop is deliberately slow, because a thermal lifts the balloon without changing its buoyancy by a single newton and a climb-rate loop faster than the disturbance will spend real heat fighting air that was going to stop pushing on its own. The outer loop is almost purely proportional, because an integrating process is offset-free under proportional control alone and reset action there would add a fourth pole at the origin for nothing. For a nonzero altitude target the output is clipped to the climb-rate envelope. When the target is the field ($z_{\mathrm{SP}} = 0$), the cascade instead requests $-\min(v_{max}, k_z z + v_{flare})$ directly and tracks the inactive altitude PI to that request. This keeps an asymptotic altitude command from bypassing the flare setting near the ground.

2.5.3 Step-Response Predictive Control

The predictive controller is dynamic matrix control in the original step-response form [6, 7], arranged so that everything expensive happens once. Figure 2.2 separates the offline path from the online path.

Identification

When the application loads, the controller integrates a reduced five-state copy of the same model, with the declared parameters of Table 2.2, from the trimmed design point of Table 2.1, twice over the horizon, once with the demand held at its trim value and once with a $+0.08$ step, and differences the two altitude trajectories, as shown in Equation (2.16).

$$a_i = \frac{z_i^{step} - z_i^{base}}{\Delta u}, \qquad a_0 = 0 \tag{2.16}$$

The reduced model carries altitude, climb rate, envelope temperature and the two actuator positions, plus the burner delay as a ring of past commands prefilled with the pre-step command, and it is integrated with explicit Euler at 0.2 s. Prefilling the ring with the stepped command instead would remove the dead time from the identified response entirely. The load and the wind are frozen, because a 52 s run burns about 430 g of propane, or 0.06 % of the all-up mass. The result at the design point is $a_1 = a_2 = 0$, which is the four seconds of dead time at a 2 s sample, $a_3 = 0.004\,\mathrm{m}$,

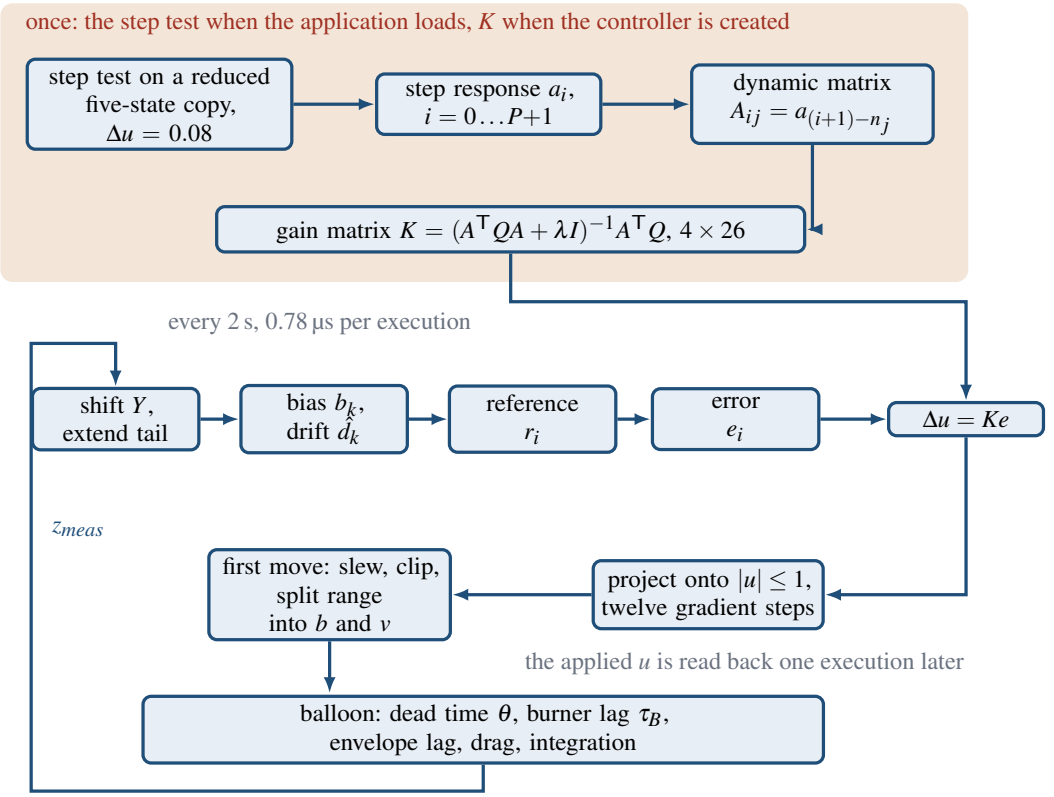


Figure 2.2 Structure of the step-response predictive controller. The shaded path runs once. The step test is taken on the declared design when the application loads, and the constant gain matrix K is produced when the controller is created. The unshaded path runs every 2 s and is one shift, one add, one 4×26 matrix-vector product and, on the small fraction of executions that violate the input box, twelve projected-gradient iterations. The prediction vector Y is advanced only by the demand the actuator actually received, which is where the anti-windup lives.

$a_{12} = 15.14$ m at 24 s and $a_{26} = 210.3$ m at 52 s. The increments still grow, $a_{26} - a_{25} = 23.9$ m against $a_{13} - a_{12} = 5.2$ m, which is what an integrating output looks like and what a self-regulating one never does. The identification is not repeated for a run, so a parameter changed from its default, another ground temperature or another starting altitude is model error that the bias of Equation (2.21) must absorb.

Dynamic Matrix and Gain Matrix

The prediction horizon is $P = 26$ points at 2 s, which is 52 s, and the move horizon is $M = 4$ after blocking at samples $n_j \in \{0, 1, 4, 9\}$, which is 0, 2, 8 and 18 s. Move blocking covers a horizon with a few decisions instead of one per sample and is standard practice [6]. The dynamic matrix and the gain matrix follow, as shown in Equation (2.17), which is Equations (1.3) and (1.5) written with A for G, P for N and λ for R.

$$A_{ij} = \begin{cases} a_{(i+1)-n_j}, & i+1 > n_j \\ 0, & \text{otherwise} \end{cases} \qquad K = \left(A^{\mathsf{T}} Q A + \lambda I\right)^{-1} A^{\mathsf{T}} Q \tag{2.17}$$

where A is 26×4, Q is diagonal with unit weight on the first eighteen points and terminal weight w_T on the last eight, and K is 4×26. The inverse is a 4×4 Gauss-Jordan elimination written out in the source. A unit step of demand is worth 210 m fifty-two seconds later, so the mean of the diagonal of $A^{\mathsf{T}}QA$ is 5.35×10^5 and a move suppression expressed in those units would need a slider spanning several decades. The application therefore exposes λ as a dimensionless multiple of the mean diagonal, as shown in Equation (2.18).

$$\lambda = \lambda_{rel}\,\frac{1}{M}\,\mathrm{tr}\!\left(A^{\mathsf{T}}QA\right) \tag{2.18}$$

so $\lambda_{rel} = 1$ always means move suppression comparable to the tracking term. The defaults are $\lambda_{rel} = 0.02$, which is $\lambda = 1.07 \times 10^4$, and $w_T = 6$. Equation (2.17) is rebuilt only when λ_{rel} or w_T moves, and the step response is never rebuilt at all.

The Online Execution

Each execution begins by accounting for what the actuator actually received, as shown in Equation (2.19).

$$Y_i \leftarrow Y_i + a_i\,(u_{now} - u_{prev}), \qquad i = 0 \ldots P \tag{2.19}$$

where u_{prev} is the demand at the previous execution. Nothing is ever added to the prediction speculatively. A move that the clip halved, that the slew limiter is still ramping toward, or that never happened because a different architecture was flying, enters at its true size, and Equation (2.19) is therefore the whole of this controller's anti-windup. It is also what keeps the predictive controller warm while another architecture flies, so selecting it in mid-flight is bumpless. The vector is then shifted one sample and the tail extended, as shown in Equation (2.20).

$$Y_i \leftarrow Y_{i+1} \quad (i < P), \qquad Y_P \leftarrow Y_P + d_1 + d_2 \tag{2.20}$$

with $d_1 = Y_P - Y_{P-1}$ and $d_2 = d_1 - (Y_{P-1} - Y_{P-2})$, which preserves both the slope and the curvature of the last two intervals and is exact for a double integrator and close for this plant. Extending the tail at a constant value would be correct for a self-regulating process and extending it at a constant slope would be correct for a single integrator, and neither describes this output over a 52 s horizon.

The correction to the plant follows, as shown in Equation (2.21).

$$\begin{aligned} b_k &= z_{meas} - Y_0, \qquad Y_i \leftarrow Y_i + b_k \\ \hat{d}_k &= \hat{d}_{k-1} + \alpha_d \left(\frac{b_k}{t_m} - \hat{d}_{k-1} \right) \end{aligned} \tag{2.21}$$

where $t_m = 2\,\mathrm{s}$ is the execution period and $\alpha_d = 0.33$ is a filter constant of about three executions. The constant b_k is the standard output-disturbance model that makes dynamic matrix control offset-free and is the predictive counterpart of integral action [18], as Section 1.4.5 shows on a self-regulating tank. On an integrating output a constant is not enough, which is the subject of Section 2.6.4. Because Y was re-anchored to the plant exactly one execution earlier and nothing else has touched it, b_k/t_m is a direct measurement of the unmodeled drift rate, clipped to $\pm 6\,\mathrm{m/s}$. The rate estimate is not written back into Y, so the estimator cannot chase its own output.

The reference trajectory carries the terminal constraint, as shown in Equation (2.22).

$$\begin{aligned} r_0 &= z_{meas} \\ \dot{r}_i &= \mathrm{clip}\!\left(\frac{z_{SP} - r_i}{\tau_{ref}},\; -v_{lim}(r_i),\; +v_{lim}(r_i) \right) \\ r_{i+1} &= \max(r_i + t_m\,\dot{r}_i,\; 0) \end{aligned} \tag{2.22}$$

Table 2.4 Scenario Sequence and the Mechanism Each Exercises

#	Scenario	What it shows
1	Hand flying from trim	the delay, discovered rather than told
2	Blast-valve step test	integrating response, no gain to read
3	Single loop on altitude	why one gain cannot close it
4	Same level-off, cascade	one slow element per loop
5	Level flight, fuel burning	a load that never stops moving
6	Up 300 m and back	the asymmetric actuator
7	Thermal, then inversion	measurement against load disturbance
8	Dawn against afternoon	density altitude as a hard constraint
9	Ballast against sink	a fast actuator that is a finite resource
10	Predictive control, plan drawn	the horizon, the bias, the plan
11	Graded landing	a terminal constraint on arrival speed

where $\tau_{ref} = 60\,\mathrm{s}$ and v_{lim} is Equation (2.8) evaluated at the reference altitude. For a landing ($z_{\mathrm{SP}} = 0$), the clipped proportional request in Equation (2.22) is replaced by $\dot{r}_i = -v_{lim}(r_i)$, with the zero floor on r_{i+1} retained. The flare setting then shapes the requested descent, while the actual velocity can still exceed the guidance profile because of tracking error. The tracking error is taken against the prediction with the drift ramp projected forward, as shown in Equation (2.23).

$$\begin{aligned} e_i &= r_{i+1} - \left[Y_{i+1} + \hat{d}_k \min((i+1)t_m,\ c_k) \right] \\ c_k &= P t_m \left(1 - \mathrm{e}^{-z_{meas}/z_{th}} \right) \end{aligned} \tag{2.23}$$

where the cap c_k tapers the projection to nothing at the surface using the same profile Equation (2.10) uses for vertical air motion. A thermal or a patch of sink dies at the ground, so extrapolating one all the way to touchdown would make the controller arrive hot and balloon back up.

The unconstrained solution is one matrix-vector product, $\Delta u = Ke$. If the cumulative sum of that plan would leave $[-1, +1]$ or exceed the per-execution rate limit, twelve projected-gradient iterations run on the same precomputed Hessian $H = A^{\mathsf{T}} Q A + \lambda I$, as shown in Equation (2.24).

$$\Delta u \leftarrow \Pi\left[\Delta u - \alpha \left(H \Delta u - A^{\mathsf{T}} Q e \right) \right], \qquad \alpha = \frac{1}{\max_j H_{jj}} \tag{2.24}$$

where Π clips each blocked move to the rate limit and then clips the running sum of the demand to the box, in order. Twelve iterations is a constant rather than a convergence test, which bounds the worst-case execution time. The first blocked move is then applied and nothing is added to the prediction, because the next execution reads back what the actuator really did through Equation (2.19). Over five independent runs of 100000 executions the online path costs 0.78 µs per execution on an Apple M-series laptop, with the individual run means between 0.757 and 0.811 µs. On the graded landing the projection was not needed on any of the 102 executions before touchdown.

2.6 Results

Table 2.4 gives the eleven scenarios in the order a flight is taught. Each one fixes the architecture, the tuning, the atmosphere and any timed events.

2.6.1 One Loop, Three Loops and a Predictive Loop

Scenarios 3 and 4 apply the same 150 m level-off, a setpoint move from 300 to 450 m at $t = 60\,\mathrm{s}$, to the same balloon through different architectures. Table 2.5 adds the predictive controller on the identical maneuver, which is scenario 4 with the architecture switched to predictive control before the run, and reports every figure over the same 2400 s run, with times measured from the setpoint change. Each retuned case below changes one setting of scenario 3 before the run.

Table 2.5 The Same 150 m Level-Off at $t = 60$ s Through Three Architectures, Read Over One 2400 s Run

	Single PID	Cascade	Predictive
Overshoot (m)	58.9	15.2	5.4
Last exit from ±15 m (s)	2340	221	220
Peak-to-peak, 300–900 s (m)	125.5	9.8	14.0
Peak-to-peak, 1800–2400 s (m)	201.1	0.8	6.8
Error at 2400 s (m)	−142.2	1.1	2.8
Demand travel (% of span)	1087	296	457

The single loop never settles and gets worse. Its swing grows from 125.5 m peak to peak over the first fifteen minutes to 201.1 m over the last ten, and the balloon is 142.2 m below its setpoint when the run ends. Neither direction of retuning helps. Raising the gain from 0.06 to 0.25 %/m widens the 300 to 900 s swing from 125.5 to 357.3 m and the 1800 to 2400 s swing from 201.1 to 372.6 m, and it still does not settle, and removing the derivative term ends the run on the ground at 4.06 m/s after 344 s, which is the proportional-integral case of Section 2.4. Reducing the gain to 0.02 %/m with the derivative removed lands the balloon as well, at 3.72 m/s after 414 s. The failure is structural rather than numerical, and scenario 3 lets the reader produce it before scenario 4 shows the alternative.

Both structured controllers settle. The cascade is the better of the two on this maneuver, with 0.8 m of residual swing against 6.8 m and 296 % of demand travel against 457 %, and the honest reading is that prediction earns nothing in the cruise. Its 15.2 m overshoot against the predictive controller's 5.4 m is the price of having to see an error before acting on it, and on a level-off that price is small.

2.6.2 The Landing

Scenario 11 is where the architectures separate. The balloon starts at 400 m on a 24 °C afternoon with 2.5 m/s of wind at 10 m and 0.4 m/s of standing sink, is cleared to land at $t = 30$ s, and meets a further patch of sink at $t = 150$ s. Table 2.6 reports the arrivals with the suggested configuration applied and the architecture then chosen before the run. The case as loaded, with its slack single loop at $K_c = 0.02\,\%/\mathrm{m}$ and no derivative, arrives in the same way as the first column, at 7.85 m/s after 108 s, more than five times the sink limit, and at 4.16 m/s across the ground, so it misses the ground-speed clause as well. The suggested configuration is predictive control with the cascade tuning and $v_{flare} = 0.70\,\mathrm{m/s}$. It lands standing up at 1.39 m/s and 3.53 m/s after 202 s, with 0.11 m/s of sink and 0.47 m/s of ground speed to spare and a 104.9 °C peak, 15.1 K under the fabric limit. The challenge is judged on the case as given, so changing a model parameter or the air voids the attempt.

> **TRY IT IN THE APP** **Challenge · Land It Under 1.5 m/s**
>
> Load the challenge and press Run as it is loaded. The slack single loop arrives at 7.85 m/s after 108 s, a hard landing. Press Reset, apply the suggested configuration, which selects predictive control, and run again: the balloon lands standing up at 1.39 m/s after 202 s.

The single loop arrives at 7.85 m/s after 108 s, which is a wreck, and it does so for a reason no gain addresses. Its vent override of Equation (2.14) opens on the way down, takes the burner away from the loop, and lets the envelope fall toward ambient. There is nowhere in a single loop to put a constraint on arrival speed. The cascade arrives at 1.27 m/s and the predictive controller at 1.39 m/s, and both meet the challenge limits. During the landing the cascade requests the descent profile directly and the predictive controller builds its reference from it. These are guidance limits, not hard constraints on the true velocity. With $v_{flare} = 0.70\,\mathrm{m/s}$, both arrivals are faster than the requested terminal rate because of tracking error. The predictive controller arrives 26.75 s sooner, while the cascade keeps more sink-rate margin, and both burn about 2.44 kg of propane. The cascade spends 48.1 s below 100 m, against 44.2 s for the predictive controller, and arrives with slightly less ground speed. A

Table 2.6 Graded Landing From 400 m on a 24 °C Afternoon. A Stand-Up Landing Requires Sink Below 1.5 m/s and Ground Speed Below 4 m/s.

	Single PID	Cascade	Predictive
Touchdown sink (m/s)	7.85	1.27	1.39
Ground speed (m/s)	4.16	3.49	3.53
Time to contact (s)	108	229	202
Propane burned (kg)	0.32	2.44	2.44
Peak envelope (°C)	104.8	104.9	104.9
Verdict	hard	stand-up	stand-up

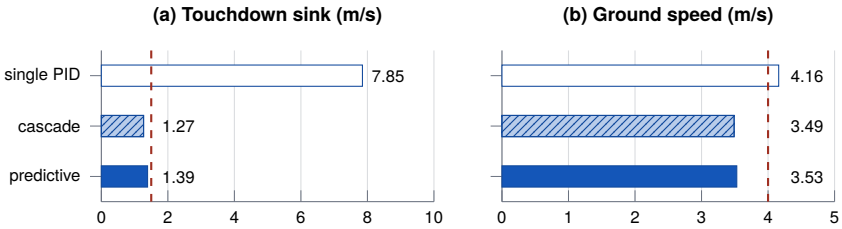


Figure 2.3 The arrivals of Table 2.6: (a) the touchdown sink rate and (b) the ground speed, white bars for the single PID loop, hatched bars for the cascade and dark bars for the predictive controller, with the stand-up limits of 1.5 m/s and 4 m/s as dashed lines and the values of panel (b) printed beyond its limit. Notice that the single loop fails both limits, while the predictive controller passes the sink limit with only 0.11 m/s to spare.

Table 2.7 Altitude Gain in m per Unit of Demand at Four Horizons and Four Step Sizes. A Linear Plant Would Give Identical Rows.

Δu	26 s	52 s	90 s	120 s
0.02	20.5	222.3	1182.8	2557.9
0.08	20.4	210.3	950.5	1808.4
0.20	20.1	191.4	735.4	1300.8
0.50	19.6	160.7	525.6	884.3

flare sweep from 0.3 to 2.5 m/s changes the cascade touchdown from 0.92 to 2.08 and the predictive touchdown from 1.19 to 1.58 m/s. The measured verdict, and not the requested profile alone, decides whether a landing passes. Figure 2.3 draws the three arrivals against the stand-up limits.

Table 2.8 Settled Altitude Hold at 300 m, Peak to Peak and Final Error Over 400 to 2000 s, for Two Prediction Horizons Across a Sweep of Move Suppression

	$P = 26$ (52 s)		$P = 60$ (120 s)	
λ_{rel}	p-p (m)	error (m)	p-p (m)	error (m)
0.004	1.4	2.4	170.2	−42.4
0.02	2.9	3.9	214.4	−99.3
0.1	17.0	12.0	229.9	47.3
0.5	24.2	−5.3	234.4	−107.5
2	28.7	−9.7	252.7	−82.5
5	35.2	−12.0	294.3	118.0

✦ DID YOU KNOW? The Hard Landing Burned the Least Fuel

From the simulator: on the graded landing the single loop burns 0.32 kg of propane, against 2.44 kg for the cascade and for the predictive controller, and it is the only one of the three graded hard (Table 2.6). The heat it never bought is the heat that would have slowed the descent.

2.6.3 Choosing the Horizon by Where the Model Stops Being True

The step response of Equation (2.16) is identified at $\Delta u = 0.08$ and is then used as though the plant were linear. It is not. Quadratic drag makes the true climb rate saturate as $\sqrt{\Delta T}$ while the identified a_i keeps growing, and rescaling the reduced model to other step sizes measures the disagreement directly. Table 2.7 normalizes the altitude gain by the step size, so a linear plant would return identical columns.

At 26 s the four rows of Table 2.7 agree to 4.3 %. At 52 s the model identified at $\Delta u = 0.08$ over-predicts a $\Delta u = 0.20$ move by 9.9 %, which is 3.8 m on a 38.3 m response. At 90 s the same comparison is 29 % and 43.0 m, and at 120 s it is 39 % and 101.5 m. The reason is visible in the climb rate. A $\Delta u = 0.20$ move has the balloon at 2.09 m/s at 52 s and 3.45 m/s at 90 s, and by then the square root in Equation (2.12) has taken hold while the linear model is still accelerating like t^3.

Weighting prediction points the model cannot predict is expensive. The general trade between horizon, move suppression and model error is set out in Section 1.4.4. The balloon adds a case the tank there does not show, a horizon too long for the model. Table 2.8 holds the balloon at 300 m under the predictive controller with no setpoint change and no disturbance, which is scenario 5 with the architecture switched to predictive control before the run, and reports the peak-to-peak altitude over 400 to 2000 s and the error $z - z_{SP}$ at 2000 s for two horizons and a sweep of move suppression. The application fixes $P = 26$, so the 120 s column is a counterfactual run of a copy of the source with only the horizon constant changed.

The 52 s horizon holds the balloon inside 2.9 m at the default weight, and its worst case anywhere in the sweep is 35.2 m. The 120 s horizon puts the loop into a 214.4 m limit cycle, and no value of λ_{rel} over three decades removes it, because raising move suppression from 0.004 to 5 moves the swing from 170.2 to 294.3 m and never below. A limit cycle that does not respond to move suppression is the signature of a model error rather than an aggressive optimizer, and the remedy is to stop weighting the part of the horizon the model cannot predict. The cycle is not an artifact of the terminal weight sitting on the least trustworthy points either. Setting $w_T = 1$ at $P = 60$, which makes Q the identity, leaves the swing between 142.8 and 293.7 m over the same move-suppression sweep, while at $P = 26$ raising w_T from 1 to 40 moves the settled swing only from 1.6 to 4.4 m. Shortening the horizon to 52 s shortens it well below the settling time of the maneuvers the controller performs, which contradicts the usual advice, and it is correct here. The rule that survives is to choose the prediction horizon by where the identified model stops being true rather than by the settling time, and no tuning table contains it.

Table 2.9 Effect of the Two Disturbance-Model Terms. Hold Is at 300 m With No Applied Disturbance, Read Over 400 to 2000 s. Landing Is Scenario 11 With the Suggested Configuration.

Disturbance model	Hold p-p (m)	Hold error (m)	Touchdown (m/s)
Bias and drift ramp	2.9	3.9	1.39
Bias only	117.0	132.9	2.18
Neither	996.7	605.6	2.51

🕮 **Pause.** Try Exercise 2.3 before reading on.

2.6.4 A Constant Bias Is Not Offset-Free on an Integrating Output

The output-disturbance model of Equation (2.21) is the standard route to offset-free predictive control, and the conditions under which it works are known [18, 19]. Adding a constant to every point of the horizon is correct when a sustained unmodeled load appears as a fixed output offset. On an integrating output it appears as a fixed offset *per sample*. The balloon supplies a permanent example without any disturbance being applied at all, because Equation (2.5) removes 8.3 g/s of load and the float temperature of Equation (2.11) therefore falls continuously.

Table 2.9 measures all three cases on copies of the shipped controller with one constant or one line changed, counterfactual runs that the application itself does not offer. With the bias alone the controller holds a 132.9 m steady offset and swings 117.0 m, because a constant added to every point of a horizon whose far end is already wrong by the drift rate multiplied by the horizon leaves the far end wrong by exactly that amount, the mechanism of Equation (1.6). Adding the two lines of Equation (2.21) that estimate the drift rate and the one term of Equation (2.23) that projects it forward takes the offset to 3.9 m and the swing to 2.9 m. The same three cases land the balloon at 1.39, 2.18 and 2.51 m/s against a 1.5 m/s stand-up limit, so only the full disturbance model passes. Removing the bias entirely leaves the controller dead reckoning off a step test taken on the declared design, and Table 2.9 prices that at 605.6 m of offset.

🕮 **Pause.** Try Exercise 2.2 before reading on.

2.6.5 Disturbances, Density Altitude and Ballast

Scenario 7 puts two disturbances through the same instrument. A thermal lifts the air and the balloon with it, so the variometer reads a climb while the buoyancy has not changed by a single newton, and the climb-rate loop drives the envelope 2.16 K below float chasing air that stops pushing on its own. An inversion is the opposite case. Eight kelvin of warm air at 400 m is air 2.7 % less dense, the float temperature at that height moves from 90.18 to 103.24 °C, and the controller is right to burn. The same reading on the same instrument means do nothing in one case and burn in the other, and the only defense the cascade has is a middle loop slower than the disturbance it should ignore.

Density altitude is a constraint no controller can relieve, and Table 2.10 gives its size. The same balloon and the same 757 kg floats at 300 m on 69.5 °C at dawn and needs 119.3 °C on a 33 °C afternoon, which is seven tenths of a degree from the fabric limit, and the ceiling collapses from 6743 m to 618 m. Every degree of ground temperature costs roughly a degree and a half of envelope temperature and two hundred meters of ceiling. A controller sitting against a limit it is not allowed to cross is behaving correctly, and the only handles are load and time of day.

Scenario 9 puts the envelope against that limit deliberately. On a 28 °C afternoon with 3 m/s of sink the cascade drives the envelope to 119.5 °C and the balloon is still sinking. One 20 kg sandbag is 196 N, which is 2.58 K of envelope temperature that is not available and 1.55 m/s of steady climb, and it is delivered with no dead time, no lag and no fuel. Releasing it turns the sink into a 0.19 m/s climb, and at 460 s it is at 262.3 m against 174.6 m for a counterfactual run with the scripted release removed, 88 m apart. There are exactly two sandbags, so the question stops being how to tune a loop

Table 2.10 Density Altitude. Float Temperature at 300 m and Ceiling at the 120 °C Fabric Limit, for 757 kg All Up.

Ground temperature	Float temperature	Ceiling
2 °C	69.5 °C	6743 m
15 °C	90.1 °C	5325 m
24 °C	104.6 °C	3779 m
28 °C	111.1 °C	2764 m
33 °C	119.3 °C	618 m

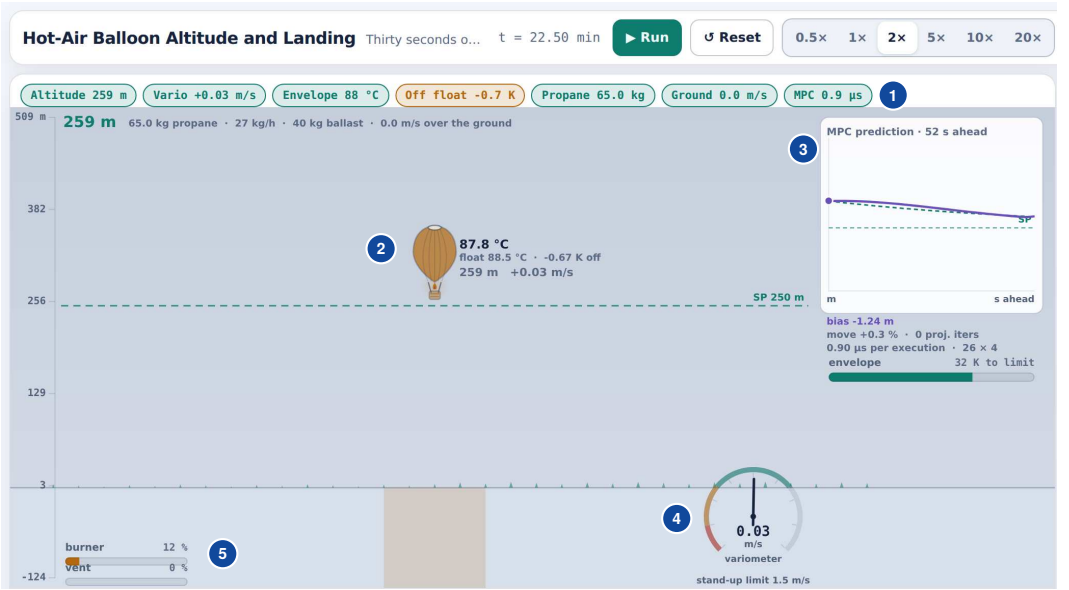


Figure 2.4 Scenario 10 as shipped at $t = 22.5$ min, after the descent from 550 m to a 250 m setpoint, cropped to the header and the plant view. (1) The badges and (2) the balloon's label give the balloon at 259 m with the variometer at $+0.03$ m/s and an envelope of 87.8 °C that is 0.67 K below float. (3) The inset draws the controller's own predicted altitude over the next 52 s against the reference it is tracking, with the live bias of -1.24 m, the last move of $+0.3$ %, zero projection iterations and the execution time the page measures on the machine running it. (4) The variometer dial reads the climb rate against the stand-up limit, and (5) the actuator bars show the blast valve, commanded at 12.0 % and printed rounded, with the vent shut.

and becomes when to spend a finite resource, which is a supervisory decision rather than a regulatory one.

Scenario 5 makes the standing load visible without any disturbance at all. Over five minutes of level flight the cascade walks its temperature setpoint down by 0.32 K and holds the altitude within 0.06 m, while the hand-flown leg that follows, switched to manual at 300 s with the blast valve trimmed to 12.60 %, climbs 48.2 m in the next five minutes and is still going up at 0.38 m/s when the clock stops. Figure 2.4 shows the application under predictive control.

2.7 Technology in Practice

What Measures It

The airworthiness standard for manned free balloons, 14 CFR Part 31, requires an altimeter and a rate of climb indicator on every balloon, and a fuel quantity gauge and an envelope temperature indicator on a hot-air balloon. The temperature indicator must show the maximum envelope skin temperature in operation and be marked with the limiting safe temperature of the envelope material [20]. In practice a cable plugged into the instrument pod runs to a sensor in the top of the envelope. Wireless links that carry the temperature signal to the basket remove the cable, but radio interference or the burner's thermal plume can degrade them, and the mechanical fuel gauges on the tanks are sometimes inaccurate and do not read from 0 to 100 % [1]. The chapter's TT is that crown sensor with a 2 s lag, and its ZT is the pod's altimeter and variometer.

> **TECHNOLOGY Sensor: Altimeter and Variometer**
>
> A pressure altimeter is an aneroid barometer. A stack of sealed, evacuated wafers expands as the static pressure falls with height, and a linkage turns that motion into an altitude. The vertical speed indicator, which balloonists call a variometer, feeds static pressure straight into a diaphragm and into its case through a calibrated leak, so a climb or descent leaves a pressure difference across the diaphragm. Its needle shows the trend at once but takes 6 to 9 seconds to settle on a new rate, a delay called the lag, and rough handling or turbulence lengthens the lag and makes the rate erratic. An altimeter that reads more than 75 feet off a surveyed field elevation goes for recalibration [21]. The chapter's ZT reads altitude and climb rate without a lag of its own.

What Moves It

The standard also requires a means for the controlled release of hot air in flight [20]. In a balloon with a parachute top, opening the parachute valve partly is the normal way to cool the envelope, and the opening closes and seals itself under the upward pressure when the line is released. Manufacturers limit how long a parachute vent may stay open, and pilots time each opening [1]. The chapter's vent follows its command with a 1.0 s travel time and is never open with the burner, by the split range of Equation (2.7).

> **TECHNOLOGY Final Element: The Blast Valve of a Vaporizer Burner**
>
> Liquid propane is drawn from the bottom of the tanks in the basket. Opening the blast valve lets it into a heat exchange coil in the flame, where it vaporizes partly or completely, and the pilot light ignites it as it leaves the coil. The valve is meant to be opened and closed briskly and fully: a partly opened valve gives a soft yellow flame of unknown output, and four one-second burns give less heat than one four-second burn. Output falls by about four percent per thousand feet of altitude, propane that is too cold cannot supply enough pressure, and a butane mixture may go out when the valve is pulled [1]. The chapter's burner delivers a continuous fraction b of a 2.80 MW rating, which a pilot can only approach by timing full burns.

Where the Controller Runs

In a manned balloon the controller is the pilot. Training starts from a standard burn of four seconds, one every 25 to 40 seconds to hold level flight, and plans each maneuver in numbers of burns [1]. In a process plant a predictive controller sits above the basic loops: a survey of industrial practice draws it executing about every minute over regulatory PID control executing every second, and notes that a controller that must execute in a few milliseconds may not be able to solve a quadratic program each time, so a good suboptimal solution may be the only option [22]. The simulator's PID loops execute every 0.5 s and the predictive controller every 2 s, and its fixed twelve projected-gradient iterations are such a bounded suboptimal solution. Figure 2.5 draws the loops on balloon hardware.

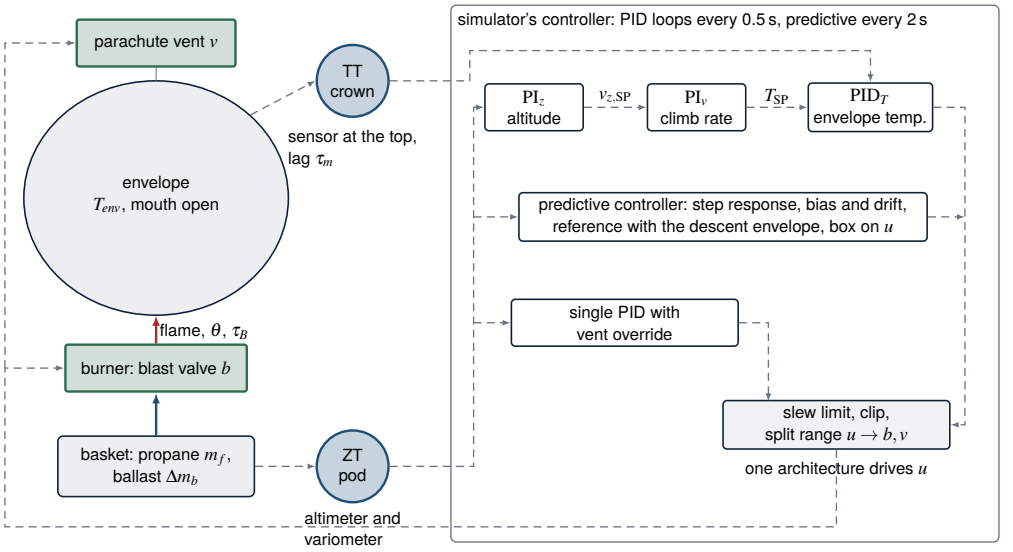


Figure 2.5 The chapter's loops on balloon hardware. ZT, the pod's altimeter and variometer, feeds the altitude loop, the single loop and the predictive controller. TT, the crown sensor, feeds the inner temperature loop of the cascade. Whichever architecture flies sets one demand u, which is slew limited, clipped and split between the blast valve and the parachute vent. Notice that the burner's delay θ and lag τ_B sit between the split range and the envelope, outside every measurement.

> **IN PRACTICE** **Flying the Flare by the Numbers**
>
> Balloon training teaches a descent that slows as the ground comes up. A new pilot matches the vertical speed indicator to the altimeter, descending at 500 feet per minute from 500 feet above the ground and at 400 from 400, slows to 200 and then 100 feet per minute, and levels off with one or two standard burns of four seconds each. Below 200 feet the pilot stops reading instruments and looks down for obstacles, power lines above all. A climb follows the same rule: the rate of climb should not exceed the distance still to go [1]. The descent envelope of Equation (2.8) has the same shape, a permitted sink that falls with height.

What Is Modernizing It

Electronic instrument systems are replacing the pressure-driven analog instruments of older balloons [1], and an instantaneous vertical speed indicator adds accelerometers to remove the variometer's lag [21]. Unmanned superpressure balloons in the stratosphere are already flown by a computer: a controller trained by reinforcement learning, which weighs cues such as wind speed and solar elevation against forecast errors and sparse wind measurements, has held balloons on station, including a 39-day controlled experiment over the Pacific Ocean [14]. The 2003 survey expected infinite prediction horizons to become a standard feature of predictive control for their theoretical properties [22]. The balloon shows that such a horizon needs a model that stays true that far ahead.

2.8 Limitations

The model is not a balloon flight model and cannot support a claim about a real balloon's climb performance, fuel burn or landing behavior. A single lumped envelope temperature, a fixed transport delay, textbook sphere aerodynamics and no radiation put it well short of the ascent codes used for mission planning [15], and nothing in it should be read as flight guidance.

The predictive controller carries three limitations worth naming beside its advantages. It identifies its step response by integrating a reduced copy of the same model at the declared design point, so it begins with a nearly perfect model of that point and is then flown against a different integrator, a

changing load, other altitudes and ground temperatures, and real disturbances. On a real plant the step test is where most of the model error comes from, and this application cannot show that. It has no state estimator and sees altitude alone, so neither the climb rate nor the envelope temperature reaches it and everything the linear model misses is absorbed into one bias and one drift rate. A state-space formulation with an observer and an augmented disturbance model would do better [19] and would not run in 0.78 µs. Volume 5, Chapter 2 carries the fuel in transit through its delay in the Kalman filter of such a controller. Output constraints are handled by the reference trajectory of Equation (2.22) and by the bounded projection of Equation (2.24) rather than as hard constraints in the optimization, so the descent envelope limits what the controller is asked to do and not what the balloon does, unlike the hard and soft output limits of Section 1.4.6. In the middle of a descent with unmodeled sink in the air the balloon exceeds Equation (2.8) by 0.35 m/s at 252 m on the graded landing, against 0.33 m/s for the cascade. A formulation with explicit output constraints and a terminal region would close that gap and would need a general quadratic-program solver [17].

Every upset is a commanded step or a timed event rather than the slow wandering of real air, and a trajectory that repeats to the last digit is itself unlike a flight. A thermal that arrives as a step of 2.0 m/s and decays with a 40 s time constant is tidier than the air over a plowed field, but it puts a measurement disturbance and a load disturbance on the same instrument within one run. A ground temperature that jumps from 2 to 33 °C between two legs of scenario 8 cannot occur in one flight, but it puts the two density altitudes side by side on one trend. Five features were kept because no controller for this plant can avoid them, which are the pure delay in front of the actuator lag, the double integration, the square-root climb characteristic, the split range that is not gain matched and the constraint on arrival speed. Section 2.3.7 states where each individual assumption breaks.

> ⚠ **OPEN CHALLENGE** **A Disturbance Model for an Output That Integrates**
>
> The constant output bias that makes a step-response controller offset-free fails on a purely integrating output, and industrial packages patch it ad hoc. One typical fix assumes a constant integrating disturbance and assigns a fraction of each prediction error to it through a rotation factor. A survey of industrial predictive control in 2003 found that factor hard to choose in practice, found the control of noisy integrating systems often poor, and called the design of sensible disturbance models a ripe area for research [22]. The simulator's altitude carries no noise, so its drift ramp has never been tested against noise. Progress would be a disturbance model that follows a drift without amplifying noise.

2.9 Discussion

Two results of this chapter transfer beyond balloons. The first is that a prediction horizon can be too long for a reason unrelated to computing cost. Table 2.8 shows a loop driven into a 214.4 m limit cycle by a horizon of 120 s on a plant whose maneuvers take three minutes, and shows that move suppression does not remove it. The diagnostic is useful in the field, because a cycle that persists across three decades of move suppression is a model problem and not a tuning problem, and shortening the horizon costs nothing that the terminal weight cannot restore. The second is the disturbance model. Dynamic matrix control is offset-free on self-regulating plants because the constant output bias captures a sustained load exactly [18], and the correct generalization for an integrating output is a ramp. Table 2.9 prices it at 129 m of steady offset and the difference between a stand-up landing and a firm one, obtained from two lines of arithmetic that add nothing measurable to the execution time.

The same two questions arise wherever an integrating output sits behind a delayed actuator that must arrive somewhere at a bounded speed. Ship steering integrates a yaw rate produced by a rudder whose hydraulics lag [2], building thermal control drives a room through a plant whose transport time is measured in minutes [3], and unmanned balloons steer entirely by choosing an altitude, which makes the loop in this application the inner loop of a navigation problem [14]. Each has an

actuator that is slow in one direction and fast in the other, each has a resource that is spent rather than modulated, and each ends against a constraint on a rate rather than on a position.

The equations of this chapter also support methods beyond the four architectures implemented. Nine states with two constrained inputs, a hard fabric-temperature limit, a finite fuel inventory and a discrete ballast release form a mixed-integer optimal control problem in which the decision of when to spend a sandbag is made by the optimizer rather than by the pilot. Moving horizon estimation on the same equations would recover the vertical air motion and the envelope temperature from altitude alone, which would replace the single bias and drift pair with a real disturbance state.

2.10 Conclusion

This chapter studied a hot-air balloon as an integrating output behind a delayed and lagged actuator that must arrive at the ground under a speed limit. The simulator integrates nine states with buoyancy, weight, quadratic drag and a potential-flow added mass of 1785 kg, an open-system envelope energy balance of 2.972 MJ/K against 4.6 kW/K of fabric loss, a 4 s burner transport delay, a split-range vent that removes heat 1.72 times faster than the burner supplies it at the design point, fuel burn at 29.9 kg/h and a standard atmosphere. An open-loop 20 % blast-valve step moves the delivered heat at 4.0 s, the variometer past 1.0 m/s at 33.2 s, and the balloon 264.0 m higher at 120 s with no steady state to read a gain from. A single loop on altitude swings 125.5 m peak to peak and grows to 201.1 m, while a three-deep cascade holds 0.8 m and a step-response predictive controller holds 6.8 m on the same maneuver. On the graded landing the three arrive at 7.85, 1.27 and 1.39 m/s against a 1.5 m/s stand-up limit. The cascade and the predictive controller use an explicit descent profile, and the actual arrival rate also depends on tuning and tracking error. Two implementation results were measured rather than assumed. Extending the prediction horizon from 52 to 120 s puts the loop into a 214.4 m limit cycle that move suppression from 0.004 to 5 does not remove, because the linear step response over-predicts a $\Delta u = 0.20$ move by 29 % at 90 s. Replacing the constant output bias with a filtered drift ramp takes the settled offset from 132.9 to 3.9 m and the touchdown from 2.18 to 1.39 m/s, at 0.78 µs per execution measured over 100000 executions. A fuller model would add envelope stratification, a burn-rate-dependent transport delay, solar and infrared terms in the energy balance, and a state-space predictive controller with an observer for comparison against the step-response form. The balloon is a plant on which to learn to predict an output that answers half a minute late.

KEY POINTS

- A balloon answers its burner late, and a burner set back to its trim holds whatever climb rate the balloon already has.
- One PID loop on altitude never settles, while a three-deep cascade gives each loop one slow element and settles.
- On the graded landing the cascade and the predictive controller arrive at 1.27 and 1.39 m/s, the single loop at 7.85 m/s.
- A 120 s horizon cycles at every move suppression, because the step response is wrong that far ahead.
- On an integrating output a filtered drift ramp takes the offset of a constant bias from 132.9 to 3.9 m.

Where else this applies. An output that integrates behind a wave travel time is also the plant of each reach of the canal in Chapter 5, where the delay sits in the water rather than in a burner plume.

References

[1] Federal Aviation Administration, “Balloon flying handbook,” FAA-H-8083-11B, Chapters 2, 7 and Glossary, https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/balloon_flying_handbook, 2024.

[2] J. Van Amerongen, “Adaptive steering of ships—A model reference approach,” *Automatica*, vol. 20, no. 1, pp. 3–14, 1984.

[3] F. Oldewurtel, A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari, "Use of model predictive control and weather forecasts for energy efficient building climate control," *Energy and Buildings*, vol. 45, pp. 15–27, 2012.

[4] K. J. Åström, C. C. Hang, and B. C. Lim, "A new Smith predictor for controlling a process with an integrator and long dead-time," *IEEE Transactions on Automatic Control*, vol. 39, no. 2, pp. 343–345, 1994.

[5] K. J. Åström and T. Hägglund, *Advanced PID Control*. Research Triangle Park, NC: ISA – The Instrumentation, Systems, and Automation Society, 2006.

[6] E. F. Camacho and C. Bordons, *Model Predictive Control*, 2nd ed., ser. Advanced Textbooks in Control and Signal Processing. London: Springer, 2007.

[7] C. E. Garcia and A. M. Morshedi, "Quadratic programming solution of dynamic matrix control (QDMC)," *Chemical Engineering Communications*, vol. 46, no. 1-3, pp. 73–87, 1986.

[8] B. W. Bequette, "Process control practice and education: Past, present and future," *Computers & Chemical Engineering*, vol. 128, pp. 538–556, 2019.

[9] J. Park, R. A. Martin, J. D. Kelly, and J. D. Hedengren, "Benchmark temperature microcontroller for process dynamics and control," *Computers & Chemical Engineering*, vol. 135, p. 106736, 2020.

[10] C. C. Gillispie, *The Montgolfier Brothers and the Invention of Aviation, 1783–1784*. Princeton, NJ: Princeton University Press, 1983.

[11] W. Marion, "Fuel requirements of a hot air balloon," in *7th Aerodynamic Decelerator and Balloon Technology Conference*. Reston, VA: American Institute of Aeronautics and Astronautics, 1981.

[12] *Balloon Flying Handbook*, U.S. Federal Aviation Administration, Washington, DC, 2022, FAA-H-8083-11B.

[13] M. S. Smith, "Demonstration of fine altitude control on stratospheric balloons to achieve a desired ground track," in *AIAA Balloon Systems Conference*. Reston, VA: American Institute of Aeronautics and Astronautics, 2017.

[14] M. G. Bellemare, S. Candido, P. S. Castro, J. Gong, M. C. Machado, S. Moitra, S. S. Ponda, and Z. Wang, "Autonomous navigation of stratospheric balloons using reinforcement learning," *Nature*, vol. 588, no. 7836, pp. 77–82, 2020.

[15] R. Farley, "BalloonAscent: 3-D simulation tool for the ascent and float of high-altitude balloons," in *AIAA 5th ATIO and 16th Lighter-Than-Air Systems Technology and Balloon Systems Conferences*. Reston, VA: American Institute of Aeronautics and Astronautics, 2005.

[16] S. K. Karanfilian and T. J. Kotas, "Drag on a sphere in unsteady motion in a liquid at rest," *Journal of Fluid Mechanics*, vol. 87, no. 1, pp. 85–96, 1978.

[17] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[18] K. R. Muske and T. A. Badgwell, "Disturbance modeling for offset-free linear model predictive control," *Journal of Process Control*, vol. 12, no. 5, pp. 617–632, 2002.

[19] G. Pannocchia and J. B. Rawlings, "Disturbance models for offset-free model-predictive control," *AIChE Journal*, vol. 49, no. 2, pp. 426–437, 2003.

[20] Federal Aviation Administration, "Airworthiness standards: Manned free balloons," Code of Federal Regulations, Title 14, Part 31, 2024 edition, U.S. Government Publishing Office, https://www.govinfo.gov/content/pkg/CFR-2024-title14-vol1/pdf/CFR-2024-title14-vol1-part31.pdf, 2024.

[21] ——, "Pilot's handbook of aeronautical knowledge," FAA-H-8083-25B, Chapter 8, Flight Instruments, https://www.faa.gov/sites/faa.gov/files/regulations_policies/handbooks_manuals/aviation/phak/10_phak_ch8.pdf, 2016.

[22] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

2.11 Exercises

2.1 Concept check · 1 min

In Scenario 1 the balloon floats trimmed at 300 m. Full burner is held for five seconds and the blast valve is then set straight back to its 12.65 % trim. What does the balloon do over the next seven minutes?

(a) It keeps climbing, because the trim holds the extra envelope temperature and so the climb rate

(b) It climbs for about a minute and returns to 300 m, because the valve is back at its trim

(c) It climbs about fifty meters and floats there, because the thinner air balances the hotter envelope

(d) It sinks back below 300 m, because the fabric loss removes the extra heat within a minute

2.2 Concept check · 2 min

The predictive controller holds the balloon at 300 m with no applied disturbance. With the constant output bias alone, and the drift ramp removed, it settles 132.9 m from its setpoint. Which explanation is right?

(a) The bias is filtered too slowly to follow the fuel burn, so it lags the plant by a fixed amount

(b) The burning fuel makes the model fall behind the plant by a fixed amount each sample, which a constant cannot represent

(c) The step response was identified with full tanks, so its gain is wrong by the share of fuel since burned

(d) The projection clips the planned moves to the actuator box, so the correction is never applied

2.3 Concept check · 2 min

With the prediction horizon lengthened from 52 to 120 s, the held balloon cycles by more than 140 m peak to peak at every move suppression from 0.004 to 5, and also with the terminal weight set to 1. What does that pattern point to?

(a) The optimizer is too aggressive, so a move suppression above 5 would remove the cycle

(b) The terminal weight pulls the plan toward the last eight points, where the predictions are least reliable

(c) The linear step response is wrong at the far end of the horizon, so every plan rests on a model error

(d) The longer horizon takes longer to compute, so the controller misses executions and acts late

2.4 Theory and calculation · 6 min

At the design point the air at 300 m has $p_a = 97773\,\mathrm{Pa}$, $T_{amb} = 13.05\,^{\circ}\mathrm{C}$ and $\rho_a = 1.19012\,\mathrm{kg/m^3}$. The balloon has $V = 3000\,\mathrm{m^3}$, a load of 757 kg, $C_D = 0.55$ and $A = 250\,\mathrm{m^2}$, with $R = 287.05\,\mathrm{J/(kg\,K)}$ and $g = 9.80665\,\mathrm{m/s^2}$.

(a) Find the float temperature from Equation (2.11). *Checkpoint:* 90.06 °C.

(b) Find the lift per kelvin and the drag constant $k_d = \frac{1}{2}\rho_a C_D A$ at float, and the steady climb rate 1 K and 4 K above float. *Checkpoint:* 75.96 N/K, 81.82 N per $(\mathrm{m/s})^2$; 0.96 and 1.93 m/s.

(c) Find how far above float the envelope must be to arrest a steady descent of 3 m/s, and of 1 m/s. *Checkpoint:* 9.7 K and 1.1 K.

2.5 Theory and calculation · 6 min

At float the envelope air has a density of $0.93778\,\mathrm{kg/m^3}$ at 90.06 °C, against 13.05 °C outside. Use $V = 3000\,\mathrm{m^3}$, $c_p = 1005\,\mathrm{J/(kg\,K)}$, $C_{fab} = 145\,\mathrm{kJ/K}$, $UA = 4600\,\mathrm{W/K}$, $K_{vent} = 50000\,\mathrm{W/K}$, $Q_{max} = 2.80\,\mathrm{MW}$, $\eta_b = 0.92$ and $\mathrm{LHV} = 46.4\,\mathrm{MJ/kg}$.

(a) Find the thermal capacity C_{env} and the time constant C_{env}/UA. *Checkpoint:* 2.972 MJ/K and 646 s.

(b) Find the trim burner opening and the propane it burns. *Checkpoint:* 12.65 %; 8.30 g/s, or 29.9 kg/h.

(c) Find the rate of change of the envelope temperature from float at full burner and at a fully open vent, and their ratio. *Checkpoint:* $+0.82\,\mathrm{K/s}$, $-1.41\,\mathrm{K/s}$, 1.72.

2.6 Theory and calculation · 7 min

Take the path from demand to altitude as one pole at the origin, a 4 s delay and real poles at 1.25/s, 1.548×10^{-3}/s (the envelope loss) and 3.056×10^{-2}/s (the drag lag at 1 m/s).

(a) Find the phase of the plant at $\omega = 0.01$/s and at 0.02/s. *Checkpoint:* $-192.1°$ and $-214.3°$.

(b) The single loop uses the ideal PID $K_c\,(1 + 1/(\tau_I s) + \tau_D s)$ with $\tau_I = 1200$ s and $\tau_D = 130$ s. Find the phase it adds at $\omega = 0.01$/s and the phase of the loop there. *Checkpoint:* $+50.6°$; $-141.5°$.

(c) Repeat (b) with $\tau_D = 0$, and explain the measured result that the loop without derivative puts the balloon on the ground. *Checkpoint:* $-4.8°$; $-196.8°$.

2.7 App challenge · 8 min

Load Scenario 2, *The step test the MPC is built from*, at the default seed and change nothing: the balloon is in Manual and the Burner blast valve steps from 12.65 to 32.65 % at $t = 20$ s. Run to $t = 150$ s at the default speed 2 (about 75 s), pause and export the CSV file. Its time column is in minutes for a run this long, so convert it to seconds before reading.

(a) From `burner_cmd_pct` and `burner_delivered_pct`, find how long after the commanded step the delivered heat leaves 12.65 %. *Checkpoint:* 4.0 s (rows at 20.2 and 24.2 s).

(b) From `vz_mps`, find how long after $t = 20$ s the climb rate first reaches 0.5 and 1.0 m/s, to 0.3 s. *Checkpoint:* 24.2 s and 33.4 s.

(c) From `alt_m`, find the altitude gained between $t = 20$ and 72 s, and compare it with what the controller's identified step response, $a_{26} = 210.3$ m per unit of demand at 52 s, predicts for this step. Give the gain to 0.3 m. *Checkpoint:* 39.6 m measured against 42.1 m predicted.

2.8 App challenge · 8 min

Load Scenario 3, *One PID loop on altitude, hunting*, at the default seed and change nothing: the Target altitude moves from 300 to 450 m at $t = 60$ s. Run to $t = 2400$ s at speed 20 (about 2 min) and export the CSV file. Repeat with Scenario 4, *The same level-off on the cascade*, as loaded. Convert the time column from minutes to seconds and read `alt_m`.

(a) For Scenario 3, give the peak-to-peak altitude over 300 to 900 s and over 1800 to 2400 s, and the altitude at 2400 s. *Checkpoint:* 125.5 m, 201.1 m; 307.8 m.

(b) For Scenario 4, give the highest altitude after the setpoint change and the last time the altitude is outside 435 to 465 m. *Checkpoint:* 465.2 m; 280.6 s, which is 220.6 s after the change.

(c) Explain why the cascade settles where the single loop does not.

2.9 App challenge · 9 min

Load the challenge, *Land it under 1.5 m/s*, at the default seed and run it as given at speed 5 until it pauses at touchdown (about 22 s). Read the Touchdown tile and its hover text. Press Reveal suggested tuning, press Reset, run to touchdown again (about 40 s) and export the CSV file. Press Reset, set Control architecture to Cascade before Run (the suggested settings stay), run to touchdown and export again.

(a) Give the touchdown sink rate, the verdict and the ground speed as given, and say which clauses of the target fail. *Checkpoint:* 7.85 m/s, hard, 4.2 m/s; both.

(b) Give the sink rate, the ground speed and the time of touchdown under the suggested predictive controller, and the propane burned (77 kg less `fuel_kg` in the last row). *Checkpoint:* 1.39 m/s, 3.5 m/s, 202 s; 2.44 kg.

(c) Give the same for the cascade. Compare both sink rates with the suggested flare speed of 0.70 m/s, and say what the comparison means for how a landing is validated. *Checkpoint:* 1.27 m/s, 3.5 m/s, 229 s; 2.44 kg.

2.10 Application area · 4 min

An operator plans a flight that must hold 300 m with the 757 kg load of Table 2.10. On a 33 °C afternoon the air at 300 m has $p_a = 97979\,\mathrm{Pa}$ and $\rho_a = 1.12206\,\mathrm{kg/m^3}$.

(a) From Table 2.10, give the margin between the float temperature and the 120 °C fabric limit at 28 and at 33 °C. *Checkpoint:* 8.9 K and 0.7 K.

(b) Find the load that floats at 300 m on 110 °C that afternoon, a 10 K margin, and how much must come off. *Checkpoint:* About 694 kg, so 63 kg less.

(c) Name the handles the operator has, and say why none of them is a controller setting.

2.11 Application area · 3 min

The model holds the envelope at one temperature, while a real envelope is stratified and its crown reaches the fabric limit first (Section 2.3.7). Suppose the crown runs 15 K above the lumped mean.

(a) Find the mean temperature at which the crown reaches 120 °C, and say which ground temperatures of Table 2.10 still let 757 kg float at 300 m. *Checkpoint:* 105 °C; 2, 15 and 24 °C, the last with 0.4 K to spare.

(b) Say what a pilot should read in flight and what the lumped model gets right.

2.12 Review the agent · 5 min

An AI assistant reviews Scenario 3 and writes: “The single loop hunts because a derivative time of 130 s is far too long. On a plant this slow the derivative mostly amplifies measurement noise. Set PID derivative time to 0 and lower PID gain to 0.02 % per m, and the loop becomes a gentle PI controller that settles without overshoot.” Find the flaw and say how you would check the proposal.