INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 2 Practical PID Control

CHAPTER 2

Automobile Cruise Control Simulator for Velocity-Form PID Under Quadratic Drag

Pages 24–54

Edited by John D. Hedengren

How to cite this chapter. "Automobile Cruise Control Simulator for Velocity-Form PID Under Quadratic Drag," in *Practical PID Control*, John D. Hedengren, Ed., vol. 2 of *Interactive Control Engineering: Principles and Applications*. Advanced Process Solutions, LLC, 2026, ch. 2, pp. 24–54.

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

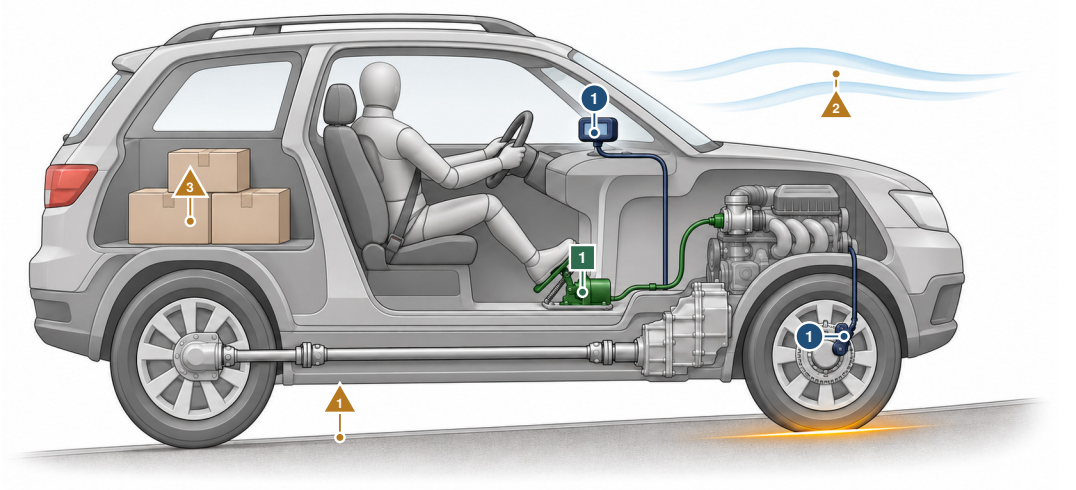
The app for this chapter
https://apmonitor.com/apps/speed/

The volume online: contents, other chapters and citation records
https://apmonitor.com/apps/books/v02-practical-pid/#speed

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

CHAPTER 2

Automobile Cruise Control Simulator for Velocity-Form PID Under Quadratic Drag



The controller moves 1 a drive that moves the gas pedal

The controller reads 1 the speedometer: a pickup at the wheel and its readout

What disturbs it 1 the slope of the road 2 a headwind 3 passengers and cargo

Notice that the controller never measures the hill, the wind or the cargo, and learns of each only from the speed the car loses. Illustration, not to scale.

IN PLAIN TERMS Holding a Car's Speed as the Road Changes

The main idea. A cruise controller holds a car at a chosen speed by moving the gas pedal. It reads the speed and presses harder when a hill, a headwind or a heavy load slows the car.

The challenge. Air pushes back harder the faster the car goes, so one push on the pedal buys less speed at high speed than at low speed. When the pedal is already on the floor, the controller must not keep asking for more, or the car shoots past its target once the load eases.

Why it matters. A test made at one speed tells only part of the story at another speed.

THREE WAYS TO READ THIS CHAPTER

Explore	12 min	the first page, the fact on how common PID control is (Section 2.1), the history of governors (Section 2.2), the history of tuning by experiment (Section 2.5.6), the fact on the offset and the try-it segments of cases 4 and 6 (Section 2.6) and the key points
Study	70 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 2.9), Limitations (Section 2.10) and the open challenges

The control question Aerodynamic drag grows with the square of road speed, so the process gain and the time constant of a cruise control loop both fall as the car goes faster. What does a step test at one cruise speed predict about the loop at another, and what does a velocity-form PID do when the pedal reaches its stop?

Learning objectives

- Derive the steady pedal map and the linearized gain and time constant from the momentum balance.
- Compute the natural frequency and damping ratio of the PI loop, and explain why only the damping moves with speed.
- Identify a first-order model from a pedal step test and tell its secant gain from the tangent gain.
- Predict the offset of a proportional loop and the integrated error of a PI loop for a road grade.
- Decide whether a setpoint is reachable, and describe how back-calculation bounds windup when it is not.

Prerequisites Sections 1.4.2 and 1.6.2 (velocity form), 1.4.4 (step test), 1.4.9 (windup) and 1.4.13 (a gain that moves with the operating point). Sections 1.4.7 and 1.6.3 are needed only for the relay test and the predictive controller.

The app https://apmonitor.com/apps/speed/

See also Volume 1, Chapter 2 and Volume 1, Chapter 3 for step-test identification on tanks whose model moves with the level or the lineup. Volume 7, Chapter 3 for a plant that the setpoint chooses, Volume 7, Chapter 4 for braking at the friction peak and Volume 7, Chapter 14 for cruise loops in a string.

2.1 Introduction

Speed regulation on a road vehicle is longitudinal vehicle control in its simplest form, a single loop closed around a road speed measurement with the throttle as its actuator [1]. A driver holding a fixed throttle operates open loop against a load that changes with every rise in the pavement and every gust of wind, and the correction is applied by a foot with no calibration and a reaction time of several hundred milliseconds. What separates this loop from other single-input single-output regulators is the form of that load.

Aerodynamic drag on a road vehicle is proportional to the square of the air speed relative to the body and is the dominant resistance at highway speeds [2]. A quadratic load makes the plant nonlinear in a form compact enough to treat in closed form. The steady pedal position required to hold a speed grows as the square of that speed, which means the incremental speed produced by one extra percent of pedal falls in inverse proportion to the speed already being held. A step test performed at 10 m/s and a step test performed at 35 m/s on the same vehicle therefore return process gains that differ by a factor of three. A process gain that moves with the operating point is the standard motivation for gain scheduling [3], which Section 1.4.13 introduces. The inertia term of this plant carries the same factor, so the identified time constant falls by the same three, and the two changes very nearly cancel inside the loop.

Simulators used to teach cruise control usually linearize the drag term or present the vehicle as a first-order lag with fixed constants [4]. However, a fixed first-order lag removes the property that makes the loop worth studying, because every step test then returns the same answer and no evidence remains that the operating point matters. This chapter works with a browser cruise control simulation that retains the quadratic drag term in full. It quantifies what a step test at one cruise speed does and does not predict about the loop at another.

The experiments built on this model exercise an identifiable set of process dynamics and control principles [4, 5]. The pedal ladder and the step tests show first-order self-regulating dynamics and the difference between a secant gain and a tangent gain on a nonlinear plant. The three disturbances separate a load change that moves the steady state from one that moves only the dynamics. Switching integral action off shows the offset that proportional action alone leaves and its dependence on the open-loop gain, and integral action removes that offset at a fixed cost in integrated error. The saturated

runs show actuator limits and anti-windup in a sampled controller. The physical fundamentals are equally compact, because the model is one momentum balance on a rigid body, the driving forces are a tractive force set by the actuator and two resistances set by the operating point, and the single time scale is the ratio of the mass to the slope of the drag curve. Gain variation with throughput is a routine commissioning problem in engine, compressor and pump control, where a loop tuned at one operating point oscillates at another, so the question this chapter poses is one a practicing engineer answers repeatedly [6].

> **DID YOU KNOW? PID Runs Most Loops**
>
> More than 95 % of all industrial control problems are solved by PID control, and many of those controllers are in fact PI controllers, because derivative action is often not included [6]. The derivative time of this chapter's cruise loop also defaults to zero.

The chapter is organized as follows. Section 2.2 places the speed loop inside adaptive cruise control and recalls the longitudinal force balance. Section 2.3 states the one-state vehicle model, with the force balance, pedal limits, instrument chain and integration scheme in enough detail to be rebuilt from scratch. Section 2.4 treats the operating-point dependence of this plant in closed form. It shows that the ratio of process gain to time constant is the thrust-to-mass ratio and carries no speed at all. As a consequence, a proportional-integral (PI) loop on this vehicle runs at a natural frequency fixed by the integral gain, the thrust parameter and the mass, with the whole speed dependence pushed into the damping ratio. Section 2.5 describes the velocity-form PID, a linear model predictive controller offered beside it, a bounded moving horizon estimator that can replace that controller's filter, and a relay experiment that writes PI constants into the PID. Section 2.6 reports runs in which one disturbance is applied at three cruise speeds and at two vehicle masses under a single set of gains, followed by saturation, setpoint and noise experiments and the shipped cases, each read out of the exported record. Section 2.7 compares the predictive controller with the PID on the shipped cases and separates what a disturbance estimate and a map of the road ahead each contribute. Section 2.8 measures the estimator and the relay against the shipped choices. Section 2.9 sets the loop in its real sensor, final element and controller platform, Sections 2.10 and 2.11 state the limitations of the model and its economic and safety context, Section 2.12 summarizes the results, and the exercises follow in Section 2.13.

2.2 Background

Speed regulation on a road vehicle started as a mechanical limit rather than as a setpoint. A governor acting on the throttle linkage caps the engine speed, and therefore the road speed in a given gear on a given grade, but it holds nothing constant, because the road load never enters the mechanism. Electronic cruise control replaced the cap with a regulator by closing a loop around a road speed measurement and positioning the throttle to hold a driver-selected value. The structure of that loop is a single-input single-output regulator on a plant whose two principal disturbances, grade and wind, are not measured, so integral action is the only route to zero offset [1].

The limitations of a speed regulator that cannot see the road ahead drove the next thirty years of work. Conventional cruise control holds a speed regardless of what is in front of the vehicle, and a comparison of that behavior against a controller that also regulates the spacing to a lead vehicle set out the requirements for an autonomous intelligent cruise control [7]. Adaptive cruise control, which switches between a speed loop and a spacing loop depending on which constraint is active, has since become the most widely deployed longitudinal automation on production vehicles, and the development of its sensing, its control laws and its switching logic has been reviewed in detail [8]. Chaining such vehicles together raises a question that a single loop does not, because a string of controllers each following its predecessor can amplify a disturbance as it propagates backward. The PATH program built and instrumented platoons to study exactly that, together with the lateral

Table 2.1 Classification of Variables. Nominal Values Are Those of the 25 m/s Cruise Point on a Flat Road With the Default 200 kg Load.

Symbol	Role	Nominal	Range
v	state, true road speed	25.00 m/s	≥ 0
$\hat{v}$ (SI)	CV, measured speed	25.00 m/s	≥ 0
v_{SP}	setpoint	25.0 m/s	0 to 45, step 0.5
u	MV, pedal position	15.31 %	−50 to 100, step 1
m_L	DV, passengers and cargo	200 kg	0 to 800, step 10
s	DV, road grade	0 %	−10 to +10, step 0.5
w	DV, headwind	0 m/s	−20 to +20, step 1
σ_n	DV, speedometer noise	0.08 m/s	0 to 0.50
v_r	relative air speed	25.00 m/s	—
x	output, distance traveled	0 m	≥ 0
saturated	output, pedal-at-limit flag	false	—

automation needed to make them practical [9]. Volume 7, Chapter 14 treats the string stability of such a platoon. In every one of these architectures the speed loop of this chapter is the inner loop, and its tuning sets what the outer loop is allowed to ask for.

The modeling side of the problem was settled earlier and has changed less. The longitudinal force balance on a vehicle is a momentum balance with three resistances, aerodynamic drag, rolling resistance and the component of weight along the road, and the tractive force available to oppose them is set by the powertrain and limited by tire adhesion [1]. Drag is written as one half the air density times a drag coefficient times a frontal area times the square of the relative air speed, and the measurement of that coefficient in wind tunnels and by coastdown testing is a mature discipline [2]. Combining the road load with a powertrain model gives the fuel consumption of a driving cycle, which is the form in which the same balance appears in propulsion system design [10]. The simulator of this chapter uses this balance with the rolling resistance term deliberately removed, for the reason given in Section 2.3.4.

> ⧖ HISTORY **Moderators and Governors, 1868**
>
> In 1868 James Clerk Maxwell set out the dynamical theory of the governor, the part of a machine that keeps its speed nearly uniform against changes in the driving power or the resistance. He separated two kinds. In the first, the regulator adds a resistance that depends on the speed itself, and more driving power leaves the speed permanently higher. Such regulators, he wrote, should be called moderators rather than governors. In the second kind, the speed error moves an independent piece that keeps adding resistance for as long as the speed stays above its normal value, and the machine returns to the same normal speed whatever the change in power or resistance [11]. The moderator acts like the proportional loop of this chapter, the governor like the loop with integral action.

2.3 Vehicle Model

The simulated vehicle is a single rigid mass traveling along a road of adjustable grade. The controlled variable is the road speed reported by the speedometer and the manipulated variable is the gas pedal position, which spans full braking to full throttle on one continuous scale. Cargo mass, road grade and headwind stay under operator control whether the loop is open or closed, and none of the three is measured, so all three are unmeasured load disturbances, although the predictive controller of Section 2.5.4 can be given the grade changes a case schedules, as a map would supply them. The loop has a single state, a single actuator and no cascade. Figure 2.1 shows each symbol at the point on the vehicle where the equations apply it.

Each signal is assigned its role in Table 2.1. Speed is carried in m/s, both pedal position and grade in percent, and time in seconds, all of them SI. A single differential equation is integrated and a single

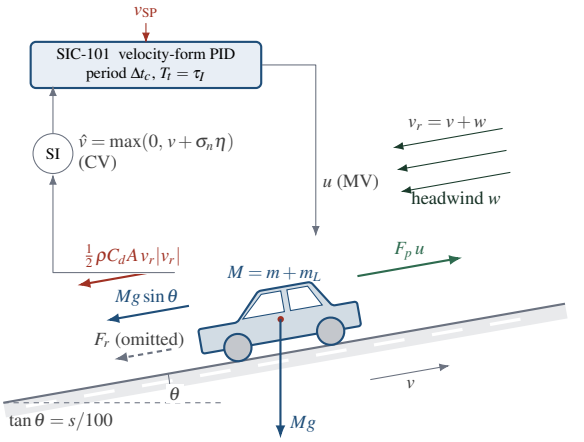


Figure 2.1 Longitudinal force balance on the simulated vehicle. The pedal produces a tractive force proportional to its position through the thrust parameter F_p, aerodynamic drag acts on the relative air speed v_r rather than on the road speed v, and a grade of s percent adds the component $Mg\sin\theta$ of the total weight along the road. Rolling resistance F_r is drawn for completeness and is not part of the model. Controller SIC-101 reads the noisy speedometer output $\hat{v}$ and positions the pedal.

algebraic bound is imposed on its solution, because clipping the road speed at zero after every step is what keeps the vehicle from rolling backward.

2.3.1 Momentum Balance

The vehicle is a point mass driven by a tractive force proportional to pedal position and resisted by aerodynamic drag and by the along-road component of its own weight, as shown in Equation (2.1).

$$M\frac{\mathrm{d}v}{\mathrm{d}t} = F_p\,u \;-\; \tfrac{1}{2}\rho\, C_d\, A\, v_r\,|v_r| \;-\; M\,g\sin\theta, \qquad v \geq 0 \tag{2.1}$$

where $M = m + m_L$ is the total moving mass in kg, m is the empty vehicle mass, m_L is the passenger and cargo load, F_p is the thrust parameter in N per percent of pedal, u is the pedal position in percent, ρ is the air density in kg/m^3, C_d is the drag coefficient, A is the frontal area in m^2, g is gravitational acceleration and θ is the road inclination. Both sides carry units of N. The relative air speed and the road inclination are the two algebraic definitions of Equation (2.2).

$$v_r = v + w, \qquad \theta = \arctan\!\left(\frac{s}{100}\right) \tag{2.2}$$

where w is the headwind in m/s, positive against the vehicle, and s is the road grade in percent rise, positive uphill. Writing the drag term as $v_r|v_r|$ rather than v_r^2 keeps the sign of the relative air speed, so a tailwind stronger than the road speed pushes the vehicle along instead of holding it back. Grade is entered as a percentage rather than as an angle because that is how road signs and survey data report it, and the arctan makes the two definitions agree exactly rather than to first order.

Grouping the three drag constants into a single coefficient $k = \frac{1}{2}\rho C_d A$ shortens every expression that follows. With the default values $k = 0.735\,\mathrm{kg/m}$, so the drag on the flat at 25 m/s is 459.4 N and the propulsive power needed to hold that speed is 11.5 kW. Setting the derivative in (2.1) to zero on a flat road with no wind gives the two steady-state maps of Equation (2.3).

$$\bar{u} = \frac{k\,\bar{v}^2}{F_p}, \qquad \bar{v} = \sqrt{\frac{F_p\,\bar{u}}{k}} \tag{2.3}$$

Equation (2.3) is the whole source of the nonlinearity. Neither expression contains the mass, so cargo changes how quickly the vehicle reaches a speed and not which speed it reaches. Holding 25 m/s on the flat requires $\bar{u} = 15.31\,\%$, holding 35 m/s requires 30.01 %, and the pedal at 100 % would settle at 63.9 m/s, which is well above the 45 m/s top of the setpoint range.

Pause. Try Exercise 2.2 before reading on.

2.3.2 Actuator and Measurement

The pedal is one signed variable rather than two, with negative values representing the brake. The applied position is clipped to $u \in [-50, 100]$ percent, so the tractive force is bounded by 3000 N at full throttle and −1500 N at full brake. On the flat with the default mass this gives 4.29 m/s^2 of acceleration from rest and −2.80 m/s^2 of deceleration from 25 m/s, the second of which is about 28.5 % of gravity and is therefore a weak brake by the standards of a real vehicle. Braking is modeled as a force independent of speed and independent of load transfer, and the clip at $v \geq 0$ is what stops the vehicle rather than the brake releasing.

What the controller sees is never the state. Equation (2.4) is the speedometer output that reaches it.

$$\hat{v} = \max(0,\ v + \sigma_n\,\eta) \tag{2.4}$$

with σ_n the standard deviation in m/s and η a fresh standard normal deviate, generated by a Box-Muller pair from the application's seeded random source once for each execution of the controller. The default $\sigma_n = 0.08\,\mathrm{m/s}$ is 0.29 km/h and is white at the 10 Hz execution rate. The instrument chain has no lag, no quantization, no bias and no dependence on tire radius. The vehicle animation and its speed readout are driven by the true state v while the speed trend and the controller are driven by $\hat{v}$, so the contribution of the instrument is visible against the physics during a run.

Table 2.2 carries every constant of the plant and the PID that a reimplementation needs, and Section 2.5.4 gives the remaining ones of the predictive controller. Parameter entries are bounded before they reach the model, because m is floored at 1 kg, F_p at machine epsilon and C_d, A, ρ and σ_n at zero, and the cargo load is floored at zero before the two masses are added. The default vehicle is light and has a large drag area, because $m = 500\,\mathrm{kg}$ and $C_dA = 1.2\,\mathrm{m}^2$ together compress the whole speed range into a response that can be observed in real time, and all six vehicle constants are operator adjustable so that a more representative vehicle can be entered. The top of the integral-time range, 120 s, switches integral action off. Every case starts at steady operation, so $v(0)$ is the steady speed of the setpoint in force in automatic, or of the pedal in force in manual, and $u(0)$ is the pedal that holds it.

2.3.3 Numerics and Clocks

The app supplies its own integration step in place of the Runge–Kutta default of the framework (Section 1.6.1). Explicit Euler on a fixed 0.01 s step advances Equation (2.1). A higher-order integrator returns nothing on this right-hand side, because the pedal is held constant between controller executions and the speed is clipped at zero, and those two discontinuities set the error floor rather than the smooth drag term. Five Euler steps make one simulation step of 0.05 s, and the odometer that drives the animation accumulates $\Delta t\, v$ inside the same loop so that the road markings and the wheels scroll at the true road speed. Two simulation steps make one controller execution of 0.1 s, and one row joins the historian every 0.1 s, holding the true speed at its own time stamp, the measured speed that the controller execution at that stamp acts on, and the pedal held over the 0.1 s just ended. The engine keeps these clocks apart from the drawing of the screen, as Section 1.6.1 describes, so the speed multiplier changes the pace of a session and not its result. The noise generator restarts from the same seed whenever a case is loaded or Reset or a setting is changed before Run, so a session at 10× repeats a session at 1× sample for sample with the speedometer noise on or off, and the exported record advances by one control period per row.

Table 2.2 Model and Controller Parameters. The First Block Is Operator Adjustable, the Second Is Fixed in the Source.

Symbol	Meaning	Default	Range	Unit
Operator adjustable				
m	empty vehicle mass	500	300–2000	kg
C_d	drag coefficient	0.24	0.10–0.60	—
A	frontal area	5.0	2.0–8.0	m^2
ρ	air density	1.225	0.80–1.50	kg/m^3
F_p	thrust parameter	30	10–60	N/%
σ_n	speedometer noise	0.08	0–0.50	m/s
m_L	passengers and cargo	200	0–800	kg
s	road grade	0	−10–+10	%
w	headwind	0	−20–+20	m/s
v_{SP}	speed setpoint	25	0–45	m/s
u	manual pedal	15.31	−50–+100	%
K_c	proportional gain	6	0–30	% s/m
τ_I	integral time	12	0.05–120	s
τ_D	derivative time	0	0–10	s
T_p	MPC prediction horizon	80	10–160	s
κ	MPC move suppression	0.3	0–3	—
—	MPC estimator	Kalman	Kalman, MHE	—
—	speed multiplier	1	0.5–10	—
Fixed in the source				
g	gravitational acceleration	9.80665	—	m/s^2
Δt	integration step (Euler)	0.01	—	s
—	simulation step	0.05	—	s
Δt_c	controller sample period	0.1	—	s
u_b	controller bias	$u(0)$	—	%
T_t	anti-windup tracking time	τ_I	—	s
T_s	MPC decision period	0.5	—	s
N_e	MHE window	30	—	executions
h	relay amplitude	7.5	—	%
ε	relay switching band	0.24	—	m/s
—	historian rate	10	—	samples/s
$v(0)$	initial road speed	$\bar{v}$	—	m/s

2.3.4 Assumptions

Only the mechanisms that alter a tuning decision survive into the model. Each item below names what has been left out and the operating region in which the omission starts to matter.

1. *Rolling resistance is absent.* The only speed-dependent resistance is aerodynamic. A rolling resistance coefficient of 0.01 on the default vehicle would contribute $f_r Mg = 68.6\,\text{N}$, which equals the aerodynamic drag at 9.7 m/s and is 15 % of it at 25 m/s. The omission therefore matters most at low speed, where it makes the simulated vehicle coast much further than a real one, and it removes the one road-load term that does not vanish as the vehicle stops.

2. *Thrust is linear in pedal position and instantaneous.* There is no engine map, no gearbox, no torque converter, no turbocharger lag and no traction limit, so F_p is the same at 5 % pedal and at 100 % pedal and at every speed. A real powertrain delivers a tractive force that falls with speed in each gear and steps at every shift, which adds a second operating-point dependence on top of the drag one and introduces dead time of the order of a few hundred milliseconds [10].

3. *Braking is negative thrust.* Negative pedal produces a force of $F_p u$ with no dependence on speed, no separation between engine braking and friction braking, no wheel lock and no anti-lock intervention. The floor at $v \geq 0$ is what brings the vehicle to rest.

4. *Constant drag coefficient.* C_d does not vary with Reynolds number, with yaw angle or with the presence of the headwind. A crosswind component changes both the effective frontal area and the drag coefficient of a real body, and measured coefficients move appreciably with yaw angle [2]. The headwind input therefore models a purely head-on or tail-on wind, which is the only case in which the relative air speed is a scalar sum.

5. *Point mass on a rigid road.* There is no pitch, no suspension, no load transfer, no tire slip and no rotational inertia of the wheels and driveline. Wheel and driveline inertia typically add several percent to the effective mass in the lowest gears and would appear in the model as an increase in M that depends on gear.

6. *Grade and wind are step-changeable and spatially uniform.* The road has one slope at a time, applied everywhere at once, and the wind has one speed. A real grade arrives progressively as the vehicle enters it, which is a ramp rather than a step and is easier for the loop to reject.

7. *Ideal speedometer apart from white noise.* The sensor has no lag, no bias, no quantization and no dependence on tire radius or wheel slip, and the noise is white and Gaussian. A wheel-speed measurement on a real vehicle is biased by tire wear and inflation and is quantized by the tooth count of the encoder, and neither error is represented.

8. *Omitted entirely.* Fuel consumption, engine speed, gear selection, road curvature, a lead vehicle, spacing control and every lateral degree of freedom are missing. None of them alters the drag-driven gain variation this chapter is built around, which is why each was left out.

2.4 Dynamic Analysis

2.4.1 Linearization About a Cruise Point

A first-order Taylor expansion of Equation (2.1) around a steady cruise point $(\bar{v}, \bar{u}, \bar{w}, \bar{s} = 0)$ produces the deviation model of Equation (2.5).

$$\tau \frac{\mathrm{d}v'}{\mathrm{d}t} + v' = K_u u' + K_w w' + K_s s' \tag{2.5}$$

where the primes denote deviations from the cruise point and the four constants are given by Equation (2.6).

$$\tau = \frac{M}{2k\bar{v}_r}, \quad K_u = \frac{F_p}{2k\bar{v}_r}, \quad K_w = -1, \quad K_s = -\frac{g\,\tau}{100} \tag{2.6}$$

with τ the process time constant in s, K_u the process gain in m/s per percent of pedal, K_w dimensionless and K_s in m/s per percent of grade. The three gains are not interchangeable and each carries a different piece of the physics. K_u and τ both carry the factor $1/(2k\bar{v}_r)$, so both shrink as the cruise speed rises. K_w is exactly -1 and is not an approximation at all, because the drag term depends on $v + w$ alone, so at a fixed pedal position the sum $v + w$ is fixed and a headwind removes its own magnitude from the road speed whatever the speed, the mass or the drag parameters happen to be. K_s inherits the time constant, so a hill costs more speed on a heavy vehicle and on a slow one.

Evaluated at 25 m/s with the default 700 kg, the linearized vehicle carries $K_u = 0.816\,\mathrm{m/(s\,\%)}$ against $\tau = 19.05\,\mathrm{s}$, and evaluated at 35 m/s the same vehicle carries $K_u = 0.583\,\mathrm{m/(s\,\%)}$ against $\tau = 13.61\,\mathrm{s}$. The two descriptions differ only in the speed at which each was taken.

🕮 **Pause.** Try Exercise 2.1 before reading on.

2.4.2 Invariance of the Gain to Time Constant Ratio

The drag coefficient, the frontal area, the air density and the cruise speed all cancel when the gain in (2.6) is divided by the time constant, as shown in Equation (2.7).

$$\frac{K_u}{\tau} = \frac{F_p}{M} = 0.0429\,\mathrm{m/(s^2\,\%)} \tag{2.7}$$

The right-hand side of Equation (2.7) is Newton's second law with the drag term deleted, and it is the initial slope of the speed response to a pedal step. Every step test on this vehicle therefore leaves the starting point at the same acceleration per percent of pedal, and what separates one test from another is the final speed and the time taken to arrive at it. Measuring that initial slope returns the thrust-to-mass ratio alone, and waiting for the response to settle returns the drag.

2.4.3 Closed-Loop Dynamics and Steady-State Error

Substituting the deviation model (2.5) into a proportional-integral control law and clearing the denominator gives the characteristic equation of the loop, as shown in Equation (2.8).

$$M\lambda^2 + (2k\bar{v}_r + F_pK_c)\,\lambda + F_pK_I = 0 \tag{2.8}$$

where λ is the complex frequency in 1/s and $K_I = K_c/\tau_I$ is the integral gain of the parallel form (Section 1.4.2), 0.5 %/m at the default $K_c = 6\,\%\,\mathrm{s/m}$ and $\tau_I = 12\,\mathrm{s}$. Reading the natural frequency and the damping ratio off Equation (2.8) gives Equation (2.9).

$$\omega_n = \sqrt{\frac{F_pK_I}{M}}, \qquad \zeta = \frac{2k\bar{v}_r + F_pK_c}{2\sqrt{F_pK_IM}} \tag{2.9}$$

Equation (2.9) is the central result for tuning this loop. The natural frequency does not contain the cruise speed at all, so the closed loop runs at $\omega_n = 0.146\,\mathrm{rad/s}$ with the default gains whether the vehicle is doing 10 or 35 m/s. The damping ratio does contain the cruise speed, and it rises linearly with it, from 0.950 at 10 m/s through 1.058 at 25 m/s to 1.129 at 35 m/s. A threefold change in process gain would ordinarily leave a controller commissioned at one end of the range unusable at the other, which does not happen on this plant. What transfers between operating points is the speed of the response, and what does not transfer is the damping and, as shown next, the offset that proportional-only control leaves behind.

Two closed-loop invariants follow directly and are worth stating separately because both are exact rather than approximate. A load force d acting along the road is rejected in steady state by a pedal increment d/F_p, whatever the speed and whatever the mass, because thrust is linear in pedal position. The integral of the error over the whole rejection is then fixed by the integral gain alone, as shown in Equation (2.10).

$$\int_0^\infty e\,\mathrm{d}t = \frac{d}{F_p\,K_I} \tag{2.10}$$

where $e = v_{\mathrm{SP}} - \hat{v}$ in m/s and the integral carries units of m. Equation (2.10) holds because the only term in the velocity-form recursion that can supply a permanent output change is the integral increment, so the accumulated error is whatever it takes to produce d/F_p of pedal. The corresponding offset when integral action is switched off is given by Equation (2.11).

$$e_\infty \simeq \frac{d}{2k\bar{v}_r + F_pK_c} \tag{2.11}$$

Equation (2.11) is the only one of the four that is approximate, because the drag term has been linearized, and it shares its denominator with Equation (2.8). A hill therefore costs a proportional loop less speed at high cruise speed than at low, which is the opposite of what the falling process gain alone would suggest, and is again a consequence of the drag itself providing part of the restoring action.

Table 2.3 Step Tests With a 2 % Pedal Increment From 2, 15 and 30 % and $\sigma_n = 0$. The Predicted Columns Come From Equation (2.6) Evaluated at the Starting Speed.

$\bar{v}$ (m/s)	m_L (kg)	$\hat{K}_u$ (m/(s %))	K_u	$\hat{\tau}$ (s)	τ	R^2
9.04	200	1.871	2.259	41.09	52.70	0.99968
24.74	200	0.799	0.825	18.42	19.25	0.99999
34.99	200	0.574	0.583	13.30	13.61	1.00000
24.74	0	0.799	0.825	13.16	13.75	0.99999
24.74	600	0.799	0.825	28.95	30.24	0.99999

2.4.4 Identification From Step Tests

Pedal step tests run by hand in manual mode, with the speedometer noise turned off, are collected in Table 2.3. Section 1.4.4 gives the step test and its fit in general. The application accepts the pedal in whole percent, so each test sets the pedal before Run at the whole percent nearest the value that (2.3) predicts for 10, 25 and 35 m/s (2, 15 and 30 %), which starts the case at the steady speed of that pedal, adds 2 % 10 s into the run and waits 600 s. The gain is the observed speed change divided by the pedal change and the time constant is the time to 63.2 % of that change, both taken from the true speed in the exported comma-separated file, and R^2 belongs to a two-parameter first-order fit to the same record. An unstepped run from the same start does not move over the window, so no drift correction is needed.

The identified gain falls by a factor of 3.26 between 9.04 and 34.99 m/s against the factor of 3.87 predicted by the ratio of starting speeds, and the identified time constant falls by 3.09 against the same prediction. Every fitted value falls short of the linearized prediction, and the shortfall widens as the vehicle slows, from 1.6 % of the gain at 34.99 m/s to 17 % of it at 9.04 m/s. The mechanism is the same one the table exists to demonstrate. A 2 % pedal step at 9.04 m/s raises the speed by 3.74 m/s, which is 41 % of the starting value, so the plant the response is traveling through has a lower gain and a shorter time constant than the plant the linearization describes, and the secant necessarily lands below the tangent. At 34.99 m/s the same step is a 3 % excursion and the two agree to within 2.3 %. The residual behaves the same way, with R^2 rising from 0.99968 to 1.00000 as the fractional excursion shrinks.

Pause. Try Exercise 2.3 before reading on.

The last two rows isolate the mass. Loading 600 kg of cargo instead of none more than doubles the identified time constant, from 13.16 s to 28.95 s, a ratio of 2.201 against the mass ratio of 2.200, while the identified gain is unchanged at 0.799 m/(s %) in all three rows. That result is Equation (2.3) restated as a measurement, because the steady-state map contains no mass and the dynamic term contains nothing else. The gravity-drained tank of Volume 1, Chapter 2 has the same structure, a gain and a time constant that move together with the operating point at a fixed ratio.

Keeping three descriptions of the same vehicle apart matters more here than it usually does. Equation (2.1) is the simulation model, nonlinear and exact inside its own assumptions. The pair $(\hat{K}_u, \hat{\tau})$ from Table 2.3 is the identified control model, good near one cruise speed and one mass and nowhere else. The PID of Section 2.5 holds no model whatever, only three tuning constants and a sample period, so the operating point has to be tracked by whoever chose those numbers. The predictive controller of Section 2.5.4 carries such a pair inside the loop, fitted once at 25 m/s and never refitted.

The guide supplied with the application quotes $K_u \approx 0.76\,\mathrm{m/(s\,\%)}$ and $\tau \approx 17.3\,\mathrm{s}$ for a 5 % pedal step at 25 m/s, beside the tangent values 0.82 and 19 s of Equation (2.6). Raising the pedal of the first case from 15.31 to 20 %, the nearest step the whole-percent pedal allows, reproduces them at 0.762 and 17.34 s, and the 2 % step of Table 2.3 does not. Both pairs are correct measurements of

the same vehicle. The difference between them is the size of the excursion, and it is the cheapest available demonstration that a secant gain on a nonlinear plant is a property of the test as much as of the process.

2.5 Controller Design

2.5.1 Velocity Form

The PID is the velocity-form algorithm of the framework, which Section 1.6.2 gives in general, and this subsection writes it in the symbols of the speed loop. The algorithm produces a change in pedal position each cycle rather than a position, which is the velocity or incremental form [12], with the proportional term acting on the error and the derivative term on the measurement. Equation (2.12) defines the error as setpoint minus measurement, and Equations (2.13) to (2.15) give the three increments.

$$e_k = v_{\mathrm{SP}} - \hat{v}_k \tag{2.12}$$

$$\Delta u_k^P = K_c\,(e_k - e_{k-1}) \tag{2.13}$$

$$\Delta u_k^I = \frac{K_c}{\tau_I}\, e_k\, \Delta t_c \tag{2.14}$$

$$\Delta u_k^D = -\,K_c \tau_D\, \frac{\hat{v}_k - 2\hat{v}_{k-1} + \hat{v}_{k-2}}{\Delta t_c} \tag{2.15}$$

with K_c measured in % s/m, τ_I and τ_D in seconds, and a sample period Δt_c of 0.1 s. The derivative acts on the unfiltered measurement. Combining the increments, clipping the demand to the pedal travel and correcting the integral gives the new output, as shown in Equation (2.16).

$$\begin{aligned} s_k &= s_{k-1} + \Delta u_k^P + \Delta u_k^D \\ c_k &= u_b + s_k + I_{k-1} \\ u_k &= \min(100,\ \max(-50,\ c_k)) \\ I_k &= I_{k-1} + \Delta u_k^I + \frac{\Delta t_c}{T_t}\,(u_k - c_k) \end{aligned} \tag{2.16}$$

Here s_k accumulates the proportional and derivative increments without limit, c_k is the output the algorithm demands, u_b is the bias, and I_k carries the integral contribution, so the integral used at scan k was accumulated through scan $k-1$. The last line is back-calculation with the tracking time $T_t = \tau_I$, and it is the only place the clip enters the recursion. Away from both limits $u_k = c_k$, the correction vanishes, and the recursion is algebraically identical to the positional law with the same three constants. On a limit it is not, which is the purpose of the correction. Keeping s_k unclipped leaves the proportional expression telescoping exactly, and Section 1.4.9 shows what clipping the running sum would cost. In the notation of Equations (1.23) to (1.26), s_k is the running sum S_k and c_k is the output before limiting v_k. The code forms the proportional increment as the difference of the whole terms $K_c e$ at consecutive scans, so a gain change moves the pedal by the change in gain times the current error and is bumpless only on a settled loop (Section 1.4.10). The loop is reverse acting in the ISA sense, because pressing the pedal raises the speed, so a speed above setpoint has to back the pedal off (Section 1.4.3). No action switch is offered.

2.5.2 Initialization and Transfer Between Modes

A velocity-form recursion has no bias term of its own, so where the recursion starts decides the operating point it holds. Whenever a case is loaded or Reset, or a setting is changed before Run, the application puts the vehicle at steady operation, at the steady speed of the setpoint in automatic or of the manual pedal in manual, sets u_b to the pedal that holds it, and sets s and the integral to zero. A case that starts in automatic therefore applies the whole proportional term $K_c e$ at its first

execution, which is the positional law with the steady pedal as its bias, and on a settled start that error is the noise of the first reading alone. Transfer between manual and automatic is bumpless in both directions by the tracking of Section 1.6.2. In manual mode the controller keeps executing and sets $I_k = u_{\text{man}} - u_b - s_k$ after every scan, so its demanded output equals the pedal the driver holds. The first automatic execution therefore moves the pedal only by the change in the proportional and derivative terms since the last manual scan. Telescoping Equation (2.16) from the transfer gives Equation (2.17) for constant gains and an output off both limits.

$$u_k = b + K_c\, e_k + \frac{K_c}{\tau_I}\,\Delta t_c \sum_{j=k_0}^{k-1} e_j - K_c\,\tau_D\,\frac{\hat{v}_k - \hat{v}_{k-1}}{\Delta t_c} \tag{2.17}$$

where k_0 is the first automatic execution and $b = u_0 - K_c e_0 + K_c \tau_D (\hat{v}_0 - \hat{v}_{-1})/\Delta t_c$ is fixed by the pedal u_0, the error e_0 and the speed change at the last manual scan. Equation (2.17) is the positional law with an effective bias b set by the moment of transfer rather than by an operator entry. Setting τ_I to the 120 s top of its range switches integral action off and freezes I_k, which leaves a proportional law whose bias is whatever $u_b + I_k$ held at that moment, and whose offset the integral no longer removes. With integral action on, $u_b + I_k$ settles on the flat at 25 m/s at the 15.31 % of Equation (2.3), which is the bias that a positional law needs at this operating point [4].

2.5.3 Saturation and Windup

Anti-windup is the back-calculation term of Equation (2.16) [13], one of the remedies that Section 1.4.9 compares. While the pedal sits on a stop $u_{\lim}$, the correction drives $u_b + I_k$ toward the value that reproduces the applied pedal. Because $T_t = \tau_I$, the integral increment $(K_c/\tau_I) e_k \Delta t_c$ cancels the part of the correction that comes from the proportional term when $\tau_D = 0$, and Equation (2.18) is what remains, the sampled counterpart of Equation (1.18).

$$u_b + I_k - u_{\lim} = \left(1 - \frac{\Delta t_c}{\tau_I}\right)\left(u_b + I_{k-1} - u_{\lim}\right) \tag{2.18}$$

The integral contribution therefore relaxes to the stop itself with time constant τ_I, removing $1/120$ of the excess each scan at the default $\tau_I = 12$ s, and the demanded output settles at $u_{\lim} + K_c e_k$. The pedal leaves the stop only when the error changes sign, which is when the speed crosses the setpoint, and the integral then has to shed the difference between the stop and the pedal the new operating point needs. Windup is bounded at the stop rather than removed, and the application offers no switch that turns the correction off. Measurements on a pedal held at 100 % for 290 s are reported in Section 2.6.

Once the loop has settled, the integral term is the only place the operating point lives, which is why a setpoint change leaves the accumulated disturbance rejection in place. The new setpoint reaches the recursion as a step in the error, the proportional increment moves the pedal by K_c times that step at the next execution, and the derivative increment, which acts on the measurement, adds nothing. Both behaviors are measured in Section 2.6.

2.5.4 Model Predictive Control

A linear model predictive controller (MPC) can drive the pedal in automatic mode instead of the PID, which stays the default in every case except case 9. It follows the general arrangement of Section 1.6.3, and this subsection gives the model, the estimator and the settings it carries for the vehicle. At each decision, $T_s = 0.5$ s or five controller executions apart, the MPC computes a pedal trajectory over a horizon and applies only its first move, keeping it to the next decision [14]. The prediction model is Equation (2.19), fitted with the speedometer noise off to the averaged responses to pedal steps of ± 3 % about the design point of the shipped cases, $\bar{v}_m = 25$ m/s on the flat with the default load and $\bar{u}_m = 15.31$ %.

$$\tau_m \frac{\mathrm{d}\xi}{\mathrm{d}t} = -\xi + K_m\,(u - \bar{u}_m + \delta + p) \tag{2.19}$$

Here ξ is the speed deviation of the model in m/s, whose predicted speed is $v_m = \bar{v}_m + \xi$, $K_m = 0.820\,\mathrm{m/(s\,\%)}$ and $\tau_m = 19.23\,\mathrm{s}$ are the fitted gain and time constant, and δ and p are an unmeasured disturbance and the grade read from a map, both in percent of pedal. The fit finds no dead time and a root-mean-square residual of 0.0011 m/s, and its pair lies 0.5 and 1.0 % above the tangent values of Equation (2.6).

The estimate $\hat{\delta}$ is supplied by a steady-state Kalman filter built on Equation (2.19), sampled with a zero-order hold at Δt_c and extended with δ as a random walk, the input-disturbance arrangement that removes offset from a predictive controller [15]. At every execution, whichever controller holds the pedal, the filter runs the two steps of Equation (2.20).

$$\begin{aligned} \bar{\xi}_k &= \phi\,\hat{\xi}_{k-1} + \gamma\left(u_{k-1} - \bar{u}_m + \hat{\delta}_{k-1} + p_{k-1}\right) \\ \begin{bmatrix}\hat{\xi}_k \\ \hat{\delta}_k\end{bmatrix} &= \begin{bmatrix}\bar{\xi}_k \\ \hat{\delta}_{k-1}\end{bmatrix} + \begin{bmatrix}\ell_\xi \\ \ell_\delta\end{bmatrix}\left(\hat{v}_k - \bar{v}_m - \bar{\xi}_k\right) \end{aligned} \tag{2.20}$$

In Equation (2.20) $\phi = e^{-\Delta t_c/\tau_m} = 0.99481$ and $\gamma = K_m(1-\phi) = 0.004\,254\,\mathrm{m/(s\,\%)}$, u_{k-1} and p_{k-1} are the pedal and the map over the period just ended, $\hat{v}_k$ is the speedometer reading shared with the PID, and $\ell_\xi = 0.162$ and $\ell_\delta = 3.53\,\%\,\mathrm{s/m}$ are the steady-state gains obtained with a measurement variance $r = \sigma_n^2$, a state noise variance of $10^{-6}r$ and a disturbance noise variance $q_\delta = 10\,r/K_m^2 = 0.0952\,\%^2$. An unmapped grade, headwind or cargo appears in $\hat{\delta}$ as an equivalent pedal deficit, which every plan carries to the end of its horizon, and a steady load therefore leaves no offset.

With the map on, p is the pedal that the grade of the coming road is worth at the design mass and thrust, as shown in Equation (2.21).

$$p(t) = -\frac{\bar{M}g}{\bar{F}_p}\,\sin\!\left(\arctan\frac{s_{\mathrm{map}}(t)}{100}\right) \tag{2.21}$$

Here $\bar{M} = 700\,\mathrm{kg}$, $\bar{F}_p = 30\,\mathrm{N/\%}$ and $s_{\mathrm{map}}(t)$ is the grade the case schedules for time t, never a grade the operator sets during a run, so a 5 % grade gives $p = -11.43\,\%$. The filter applies p as a known input, which leaves $\hat{\delta}$ only what the map omits, and each plan reads p over its whole horizon, as look-ahead controllers for heavy trucks read the slope of the coming road from map data [16]. The map is exact and is converted at the mass of the default vehicle, so case 9 is a teaching bound on what a preview can deliver.

At each decision a quadratic program is solved on Equation (2.19) sampled at T_s, over a horizon $T_p = 80\,\mathrm{s}$ of $N = 160$ steps, in which twelve pedal positions μ_j, held over move blocks of $(1,1,1,1,2,2,3,4,6,9,15,115)$ steps, minimize Equation (2.22) subject to the hard limits $-50 \le \mu_j \le 100$.

$$J = \frac{T_s}{\tau_n}\sum_{k=1}^{N}\left(\frac{v_{m,k} - v_{\mathrm{SP}}}{y_s}\right)^2 + \kappa\sum_{j=1}^{12}\left(\frac{\mu_j - \mu_{j-1}}{u_s}\right)^2 \tag{2.22}$$

Here $v_{m,k}$ is the speed predicted k steps ahead, μ_0 the pedal now, $\tau_n = 19.2\,\mathrm{s}$, $y_s = 1\,\mathrm{m/s}$ the settling band of the scorecard, $u_s = 150\,\%$ the pedal span and $\kappa = 0.3$ a dimensionless move suppression, named λ in the application and renamed here because λ is the complex frequency of Equation (2.8). No speed band, rate limit or terminal weight enters. The decisions are departures from $\bar{u}_m$, the Hessian carries a ridge of 10^{-9} relative to the two weights, and a dual active-set method [17] returns the optimum. One rule, common to the seven single-loop studio apps, set κ and q_δ before any comparison, as the largest κ within 10 % of the best true-value IAE with no more pedal travel than the PID on cases 3 and 4.

The plan is discarded if the solver finds no optimum or if the move it would apply is not a finite number inside the pedal travel. The PID is the fallback of Section 1.6.3. It follows the applied pedal at every execution, so it takes the pedal without a bump until a plan is accepted again. Forced at

20.5 s of case 9 with the noise off, while the plans are adding pedal for the grade, a rejection moves the pedal by 0.092 %, the size of each of the next three PID moves, and the plan that returns at 21.1 s moves it by 3.8 %. The MPC reads the speedometer, the pedal it applied, the setpoint, the pedal limits and, with the map on, the grade schedule of the case, and nothing of the plant, so the vehicle of Equation (2.1), the instrument of Equation (2.4) and the model of Equation (2.19) stay distinct.

2.5.5 Moving Horizon Estimation

The state from which each plan starts has a second possible source. Under the MPC, a moving horizon estimator (MHE) can replace the filter of Equation (2.20) as that source, fitting the same sampled model to the latest readings while holding the states inside declared bounds [18]. The filter itself runs on in either case. With the augmented state $x_j = [\xi_j \;\; \delta_j]^{\mathsf{T}}$ and the window $j = s, \dots, k$, $s = k - N_e$, every execution solves Equation (2.23) for x_s and the process noise $\nu_s, \dots, \nu_{k-1}$.

$$\begin{aligned} \min \; & \|x_s - \bar{x}_s\|^2_{\bar{P}^{-1}} + \sum_{j=s}^{k-1} \|\nu_j\|^2_{Q^{-1}} + \frac{1}{r} \sum_{j=s}^{k} (\hat{v}_j - \bar{v}_m - \xi_j)^2 \\ \text{s.t. } & x_{j+1} = \begin{bmatrix} \phi & \gamma \\ 0 & 1 \end{bmatrix} x_j + \begin{bmatrix} \gamma \\ 0 \end{bmatrix} (u_j - \bar{u}_m + p_j) + \nu_j \\ & \bar{v}_m + \xi_j \geq 0 \end{aligned} \tag{2.23}$$

In Equation (2.23), $Q = \operatorname{diag}(10^{-6} r, q_\delta)$ and r are the variances of the filter, u_j and p_j are the pedal and the map applied from execution j to $j+1$, $\bar{x}_s$ is the estimate the MHE formed at $s-1$ advanced one step through the model, and $\bar{P}$ is the steady a-priori covariance of the filter, held fixed as the arrival cost. With $N_e = 30$ the window covers 3 s and 31 readings, so a solve has 62 decisions and 31 bound rows. A dual active-set method [17] returns each solution. The only bound is the $v \geq 0$ of Equation (2.1), the one physical limit that the declared values place on the speed, while δ stays unbounded because nothing declares its range in pedal units. A window that ends without an optimum, with a non-finite estimate or with one outside the bound is rejected, and its estimate is replaced by the Kalman update of the one-step MHE prediction, clipped to the bound and marked in the log. A vehicle constant changed by the operator reaches the MHE only through the speedometer, exactly as it reaches the filter.

2.5.6 Relay Autotuning

The PID constants can also be measured on the running loop instead of derived from a step test, by the relay feedback experiment of [19], which Section 1.4.7 introduces. The application offers it as an action taken from the present state in automatic, never as a mode in which a case runs. The relay takes the pedal from the PID and holds it $h = 7.5\,\%$, a twentieth of its travel or 0.9 of the distance to the nearer stop if that is less, above or below the pedal averaged over the preceding 6 s, one tenth of the trend window, switching each time the speedometer reading crosses the setpoint by more than $\varepsilon = 3\sigma_n = 0.24\,\mathrm{m/s}$, where σ_n is the declared noise and not the value set for the run. A cycle counts only when each half lasts four executions or more. When two counted cycles agree within 5 % in period and in swing $\pm a$, the describing function of a relay with hysteresis and the Tyreus–Luyben rule of Table 1.2 set the constants of Equation (2.24) and $\tau_D = 0$, where K_{cu} is the ultimate gain in % s/m, D the share of each period during which the pedal sits high and P_u the measured period.

$$K_{cu} = \frac{4h \sin(\pi D)}{\pi \sqrt{a^2 - \varepsilon^2}}, \qquad K_c = \frac{K_{cu}}{3.2}, \qquad \tau_I = 2.2\, P_u \tag{2.24}$$

The first relation is Equation (1.16), in which the relay amplitude and the swing are written δ and α, with the factor $\sin(\pi D)$ added for a cycle whose two halves are unequal. The experiment gives up after 30 cycles or 120 s. The PID tracks the applied pedal throughout and resumes from it with

whatever constants the experiment left. Where the swing is small against the band, the band and not the loop fixes the cycle, and the application flags the result once $\psi = \arcsin(\varepsilon/a)$, the angle separating the observed cycle from the -180° point, reaches 10°.

> ⧖ HISTORY **Tuning by Experiment, 1942 to 1984**
>
> The first tuning rules for PID controllers were presented by Ziegler and Nichols in 1942, after extensive experiments with pneumatic simulators and a differential analyzer. One method reads a bump test of the open loop, and the other raises the proportional gain until the loop starts to oscillate. The rules had a huge impact, and controller makers adopted them for routine use [6]. In 1984 it was still common experience that many industrial regulators were poorly tuned, partly because simple and robust tuning methods had not been available. A relay placed in the loop then generated its own test signal and found the critical gain and frequency, a method simple to run on a microprocessor [19]. Relay autotuning, started by pushing a button, is now used in many commercial PID controllers [6].

2.6 Results

The application is used as Section 1.6.4 describes for every framework app. Manual mode positions the pedal directly, automatic mode closes the loop of Section 2.5, and each run exports as a comma-separated file with the measured and the true road speed, the pedal, the setpoint and the three disturbances at ten rows per second. Ten cases ship with the application, each starting at steady operation, and Table 2.4 lists them in selector order with the requirement for completing each. Every experiment below, up to Section 2.6.6, starts from the first of them, in which the vehicle cruises at 25 m/s in manual with the pedal at 15.31 %, and is set up from the operator inputs of Tables 2.1 and 2.2 before the run starts, with every setting on a step the application accepts. A setting chosen before Run moves the starting steady state rather than acting as a step, so each step or upset below is applied after the run has started, and the application gives the same run whether a setting is entered before Run or carried through Reset. The scorecard reports the integrated absolute error (IAE), overshoot, settling time, pedal travel and share of time on a limit, but it integrates the whole run, so every number quoted here is computed from the exported record over the window named, or at the 0.05 s simulation step where stated. The sequence appears in Table 2.5. Runs used for identification and for the deterministic offsets set the speedometer noise to zero, which the application allows.

2.6.1 Manual Operation and the Steady-State Map

Equal increments of the pedal produce speed increments that shrink as the vehicle goes faster. With the pedal set to zero before Run, so that the first case starts at rest, manual settings of 5, 10 and 15 % applied in turn from $t = 1$ s, each held for 150 s, reach 13.97, 20.19 and 24.74 m/s against the 14.29, 20.20 and 24.74 m/s of Equation (2.3), so the second 5 % increment buys 4.55 m/s against 6.22 m/s for the first. The first plateau is 0.31 m/s short of its resting value because the approach from rest carries the longest time constant of the three, 33.3 s against 23.6 and 19.2 s. The same three plateaus show the time constant moving, because each approach is quicker than the one before it although the pedal steps that produced them are the same size. Figure 2.2 shows the approach from rest at 15 % of pedal. The vehicle reaches 24.64 m/s 119 s after the pedal is applied.

The two disturbances that a step test cannot distinguish are separated by holding the pedal still. With the pedal set before Run to 15 %, the whole percent nearest the 15.31 % of Equation (2.3), the vehicle starts and rests at 24.7436 m/s, a 5 m/s headwind takes it to 19.7436 m/s, a 5 m/s tailwind takes it to 29.7436 m/s, and removing the wind returns it to 24.7436 m/s. The unity disturbance gain of Equation (2.6) is reproduced to four decimal places in both directions. Adding 400 kg of cargo to the same standing experiment leaves the resting speed at 24.7436 m/s, because the mass appears nowhere in (2.3).

Table 2.4 Scenarios and Completion Requirements

	Scenario	Kind	Required to complete
1	Reference operating point	Experiment	In manual at the steady cruise of 25 m/s with the pedal at 15.31 %, run for 15 s and confirm that nothing moves.
2	Open-loop actuator step	Experiment	From the same cruise in manual, the pedal steps from 15.31 to 35.31 % at 5 s. Run through the step to 35 s, by which time the speed is 36.56 m/s and heading for 37.96 m/s.
3	Closed-loop setpoint step	Experiment	In automatic at 25 m/s, the setpoint steps to 29.5 m/s at 5 s. Run until the speed has stayed within $\pm 1\,\mathrm{m/s}$ of the new setpoint for 3 s.
4	Disturbance rejection	Experiment	In automatic at 25 m/s, the road grade steps from 0 to 5 % at 5 s. Run through the step to 35 s.
5	Recover by hand	Exploration	In manual at 25 m/s with the same grade step at 5 s, set the pedal by hand after the step to hold the speed, once settled, within half of its largest error since the step, or within $\pm 1\,\mathrm{m/s}$, for 3 s in manual.
6	Conservative tuning	Experiment	In automatic at 25 m/s with $K_c = 2.4\,\%\,\mathrm{s/m}$ and $\tau_I = 30\,\mathrm{s}$, the setpoint steps to 28.6 m/s at 5 s. Run through the step to 35 s.
7	Aggressive tuning	Experiment	As case 6 with $K_c = 12\,\%\,\mathrm{s/m}$ and $\tau_I = 6\,\mathrm{s}$.
8	Actuator limit	Demonstration	In automatic at 25 m/s with the pedal thrust lowered to 10 N/% (pedal 45.94 %), the setpoint steps to 40 m/s at 5 s. Run until the pedal reaches its limit and for at least 20 s.
9	MPC with the hill on the map	Experiment	In automatic at 25 m/s under the predictive controller with the map on, a 5 % grade begins at 20 s. Run 60 s from $t = 0$ under the MPC, then reset and run the same 60 s under the PID.
10	Return to target	Challenge	Starting from $K_c = 1.2\,\%\,\mathrm{s/m}$ with the road grade stepped from 0 to 5 % at 5 s, retune so that the measured speed stays within $\pm 2\,\mathrm{m/s}$ of 25 m/s over the last third of a 30 s run (19.8 to 30 s), with the model parameters and disturbances as given.†

†With the shipped K_c the vehicle is still at 21.77 m/s when judging starts and the worst measured error is 3.52 m/s (3.26 m/s in the true speed), so the case fails. The suggested $K_c = 6\,\%\,\mathrm{s/m}$ and $\tau_I = 12\,\mathrm{s}$ pass with a worst measured error of 0.918 m/s (0.806 m/s true), the same whether revealed before Run or carried through Reset. Raising K_c alone also passes, from 2.76 % s/m with the application's own noise sequence. Model runs with four other noise seeds give 3.35 to 3.43 m/s shipped and 0.84 to 0.93 m/s suggested.

Table 2.5 Experiment Sequence and the Mechanism Each One Exposes

Experiment	Setup	What it exposes
Pedal ladder	manual 5, 10, 15 % from rest	equal pedal steps buy less speed each time
Step test	2 % step from 2, 15 and 30 %	$\hat{K}_u$ and $\hat{\tau}$ both scale as $1/\bar{v}$
Cargo at fixed pedal	m_L 200 to 600 kg in manual	no steady-state effect at all
Wind at fixed pedal	w stepped to $\pm 5\,\mathrm{m/s}$	$K_w = -1$ exactly, at any speed
Hill, PI	s 0 to 4 %, $\tau_I = 12\,\mathrm{s}$	invariant $\int e\,dt$ and moving damping
Hill, P only	same upset, τ_I at its 120 s top (integral off)	offset of Equation (2.11)
Pedal on the stop	v_{SP} 25 to 40 m/s, 8 % grade, 15 m/s wind, 600 kg	windup bounded at the stop
Setpoint change on a hill	v_{SP} 25 to 25.5 m/s	integral history survives the change
Setpoint step on the flat	v_{SP} 25 to 28 m/s, τ_D 0 and 0.8 s	no derivative kick
Noise and derivative	τ_D 0 and 0.2 s at 25 m/s	pedal activity against speed variation

2.6.2 Grade Rejection at Three Cruise Speeds and Two Masses

Table 2.6 reports the identical load disturbance, a road grade opened from 0 to 4 %, applied to the same controller at three cruise speeds and at two masses. Each run chooses automatic, the setpoint at the cruise speed, the cargo and the default tuning before Run, so it starts at the steady state of that setpoint, and the grade is applied in one step 10 s into the run. The speedometer noise is turned off for these runs, so the offsets quoted are the deterministic ones.

Table 2.6 Response to a Road Grade Step From 0 to 4 % With $K_c = 6\,\%\,\mathrm{s/m}$, $\tau_D = 0$ and $\sigma_n = 0$, Measured Over the 300 s Following the Upset. Deviations and Times Are Read From the True Speed, IAE From the Measured Speed.

	$m_L = 200\,\mathrm{kg}$			$m_L = 600\,\mathrm{kg}$
$\bar{v}$ (m/s)	10	25	35	25
disturbance force d (N)	274.4	274.4	274.4	431.1
ω_n (rad/s)	0.1464	0.1464	0.1464	0.1168
ζ from Eq. (2.9)	0.950	1.058	1.129	0.844
Integral action, $\tau_I = 12\,\mathrm{s}$ ($K_I = 0.5\,\%/\mathrm{m}$)				
peak deviation (m/s)	1.029	0.957	0.914	1.389
predicted	1.019	0.949	0.907	1.378
time of peak (s)	6.9	6.7	6.5	9.0
predicted	6.9	6.7	6.6	9.0
IAE (m)	18.29	18.29	18.29	29.19
from Eq. (2.10)	18.29	18.29	18.29	28.74
pedal increment (%)	9.146	9.146	9.146	14.372
return within 0.05 m/s (s)	36.6	41.3	44.2	41.5
Proportional only, integral switched off at the upset				
final offset (m/s)	1.417	1.271	1.190	—
from Eq. (2.11)	1.409	1.266	1.185	—
pedal travel (%)	8.5	7.6	7.1	—

The linearized description of Section 2.4 accounts for every entry in Table 2.6 to within 1 %. The peak deviation under integral action falls from 1.029 m/s at 10 m/s to 0.914 m/s at 35 m/s, an 11 % improvement produced entirely by the rise in damping ratio from 0.950 to 1.129, and the predicted peaks of 1.019 and 0.907 m/s sit 0.8 to 1.0 % below the measurements. That remainder is the 0.1 s sample period, because a model run of the same loop executed every 0.01 s comes within 0.3 % of the predictions. The time of the peak barely moves, from 6.9 to 6.5 s, which is the invariant natural frequency of (2.9) showing itself directly. The integrated error is 18.29 m in all three of the 200 kg runs, matching Equation (2.10) to the digits reported, and the steady pedal increment is 9.146 % in all three, matching d/F_p. Figure 2.3 draws the measured rows of Table 2.6 as bars.

The proportional-only rows run the other way from the intuition that a falling process gain gives a worse loop. In these runs τ_I is set to the 120 s top of its range at the moment of the upset, which switches integral action off and freezes the integral at its settled value as the bias of a proportional law. The offset shrinks as the vehicle goes faster, from 1.417 m/s at 10 m/s to 1.190 m/s at 35 m/s, because the drag term $2k\bar{v}_r$ in the denominator of Equation (2.11) is itself a proportional feedback supplied by the vehicle itself and grows with speed. At 10 m/s that term contributes 14.7 N s/m against the 180 N s/m of the controller, and at 35 m/s it contributes 51.45 N s/m. The controller is doing 92 % of the work in the first case and 78 % in the second.

Cargo is the disturbance that changes the loop rather than the setpoint. The same grade step applied to the same controller with 600 kg aboard produces a peak deviation of 1.389 m/s against 0.957 m/s with 200 kg, because the added mass drops ω_n from 0.1464 to 0.1168/s and ζ from 1.058 to 0.844 while also raising the disturbance force from 274.4 to 431.1 N. The measured IAE of 29.19 m exceeds the 28.74 m of Equation (2.10) by 1.6 % for a reason worth stating, because $\zeta < 1$ in this run alone and the speed therefore overshoots the setpoint on the way back, so the absolute error integrated by the metric exceeds the signed error fixed by (2.10). Integrating the absolute value of the linearized response at $\zeta = 0.844$ gives 29.16 m against the signed 28.74 m, which accounts for the difference to within 0.1 %.

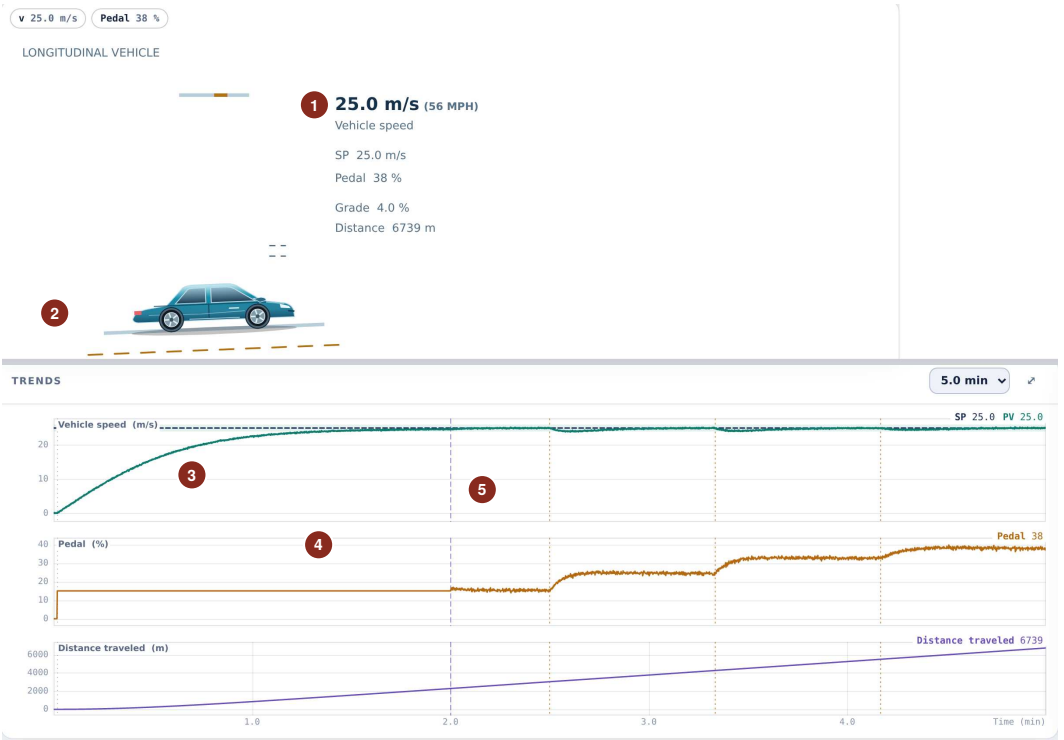


Figure 2.2 Start-up, transfer to automatic and three load disturbances in the first case with the default speedometer noise of 0.08 m/s, captured at $t = 300$ s and cropped to the vehicle view (top) and the trends (bottom). The pedal is set to zero before Run, so the case starts at rest, and to 15 % at $t = 1$ s. The vehicle reaches 24.64 m/s by $t = 120$ s, where the loop is switched to automatic with $v_{SP} = 25\,\mathrm{m/s}$, $K_c = 6\,\%\,\mathrm{s/m}$ and $\tau_I = 12$ s, and the pedal continues from the 15 % the driver held. A 4 % grade at $t = 150$ s, a 6 m/s headwind at $t = 200$ s and 400 kg of extra cargo at $t = 250$ s pull the true speed down to 24.03, 24.14 and 24.50 m/s, and after each the speed returns to setpoint while the pedal settles about the 24.46, 32.69 and 37.92 % the force balance requires. At $t = 300$ s the measured speed is 24.98 m/s and the pedal 37.87 %. (1) The speed readout with the setpoint, the pedal, the grade and the distance. (2) The animated vehicle on the grade, driven by the true speed. (3) The speed trend with the setpoint dashed, where the speedometer noise remains visible throughout. (4) The pedal trend. (5) The event markers, the switch to automatic dashed and the three upsets dotted.

> ✦ **DID YOU KNOW? Faster Car, Smaller Offset**
>
> From the simulator: with integral action switched off, the same 4 % grade leaves an offset of 1.417 m/s at 10 m/s but only 1.190 m/s at 35 m/s, although the process gain is lower at the higher speed. The drag itself supplies part of the restoring action (Table 2.6).

2.6.3 Actuator Saturation

A cruise setpoint of 40 m/s with 600 kg of cargo, an 8 % grade and a 15 m/s headwind demands more force than the powertrain can supply. The run starts in automatic at 25 m/s under that load, with the pedal at the 67.875 % that holds it, and the setpoint is raised to 40 m/s 10 s into the run. The pedal reaches its 100 % stop at that execution and stays there for the 290 s before the wind is removed at $t = 300$ s, and the vehicle settles at 38.956 m/s, which is what solving $F_p \cdot 100 = k(v+w)^2 + Mg\sin\theta$

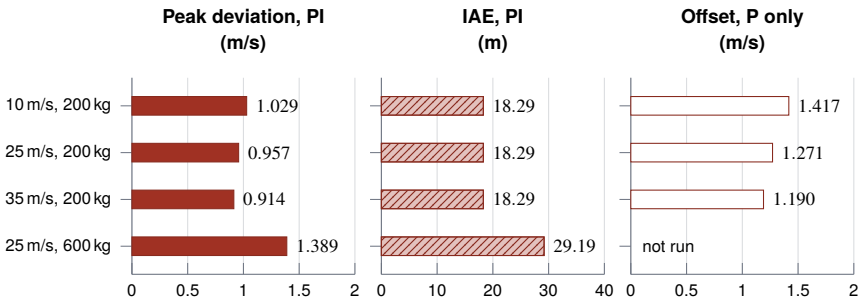


Figure 2.3 The grade step of Table 2.6 at three cruise speeds and two loads under one set of gains: the peak deviation (solid bars) and the integrated absolute error (hatched bars) under integral action, and the final offset with integral action switched off (open bars). Notice that the integrated error does not move with the cruise speed, that the peak and the offset both shrink as the car goes faster, and that cargo, not speed, is what makes the loop worse.

for the road speed returns. The controller is not at fault and the offset of 1.044 m/s is not a tuning failure, because the setpoint lies outside the set of speeds this road load allows. Meanwhile $u_b + I$ relaxes through Equation (2.18) from the starting 67.875 % to the stop itself, within 0.001 % of it 124 s after the setpoint change and at 100.000 % when the wind is removed.

Removing the headwind at $t = 300$ s does not release the pedal at once. It leaves the stop at the scan 1.2 s later, the first at which the measured speed exceeds 40 m/s, and is at 91.4 % at the scan 3 s after the removal and at 66.58 % at the scan 28 s after it. The speed overshoots to 42.896 m/s, which is 7.2 % above the setpoint, 9.9 s after the removal, comes back within 0.05 m/s of the setpoint after 54.1 s, and rests at 40.000 m/s with the pedal back at 67.875 %, because the air speed is 40 m/s in both steady states. The overshoot is the vehicle's own inertia carrying it through the 1024 N of excess thrust that existed at the instant the wind stopped, extended by the 1.2 s on the stop and by the 32 % of pedal the integral has to shed once the speed has crossed the setpoint. A model run of the same controller with the back-calculation removed, which the application does not offer, winds $u_b + I$ up to 323.2 % during the 290 s on the stop, holds the pedal down for 36.0 s after the wind is removed and carries the vehicle to 52.69 m/s. The difference between the two behaviors is the content of the last line of Equation (2.16).

2.6.4 Setpoint Changes

Raising the setpoint from 25.0 to 25.5 m/s at $t = 300$ s, while the loop is holding 25 m/s against a 4 % grade that arrived 10 s into the run, moves the pedal from 24.458 to 27.458 % on the next control execution, a step of 3.000 % which is the proportional gain acting on the 0.5 m/s change in error. The speed never falls below the old setpoint, peaks at 25.521 m/s, comes within 0.05 m/s of the new setpoint after 7.2 s, and rests at the 25.077 % of pedal that the force balance on the grade requires at the new speed. The 9.146 % of pedal accumulated against the hill survives the setpoint change, because the velocity form applies the new setpoint to the next increment and leaves the integral where it was.

On the flat, a setpoint step from 25 to 28 m/s 10 s into the run with the default tuning moves the pedal by $K_c \Delta e = 18$ % at the next execution, from 15.31 to 33.31 %, which is also the highest the pedal goes. The speed overshoots to 28.107 m/s and stays within 0.05 m/s of the new setpoint from 30.3 s after the change. Adding derivative action with $\tau_D = 0.8$ s ($K_D = K_c \tau_D = 4.8\,\%\,\mathrm{s}^2/\mathrm{m}$) leaves that first increment unchanged, because the derivative acts on the measurement and the setpoint never enters it, so the application has no derivative kick to show. The derivative instead opposes the acceleration the proportional step started, the speed overshoots further to 28.170 m/s, the approach takes 37.1 s, and the pedal travel over the 200 s following the change rises from 32.2 to 33.5 %.

2.6.5 Measurement Noise and Derivative Action

With the default 0.08 m/s of speedometer noise and the default tuning, a settled loop at 25 m/s holds the pedal to a standard deviation of 0.482 % against the $K_c\sigma_n = 0.480\,\%$ that the proportional term alone contributes, and the true speed varies by only 0.010 m/s because a 19 s time constant is an effective low-pass filter on the pedal activity. A derivative time of 0.2 s ($K_D = 1.2\,\%\,\mathrm{s}^2/\mathrm{m}$) nearly quadruples the pedal standard deviation to 1.793 % and raises the pedal travel from 5.5 to 24.7 % per second while the true speed variation rises to 0.011 m/s rather than falling. Each figure is taken over 600 s of a settled run with the application's own noise sequence. Model runs with four other noise seeds, which the application does not expose, move the pedal statistics by up to 3 % and the true speed variation by up to 13 %, and the rise in true speed variation with derivative action appears in every one of them. The short sample period is what makes derivative action expensive on this loop, because $K_D/\Delta t_c = 12\,\%\,\mathrm{s/m}$ is already twice the default proportional gain. Derivative action has nothing to recover on a first-order plant with no dead time, which is why τ_D defaults to zero.

2.6.6 The Shipped Cases Under the PID

The experiments above are built on case 1, because a settled loop is what the analysis of Section 2.4 describes. The other cases of Table 2.4 add results on the vehicle as shipped. Every case starts at steady operation, cruising at 25 m/s, so each scripted event at 5 s lands on a settled loop. In case 4 the 5 % grade pulls the true speed down by 1.21 m/s, to its lowest at 12.1 s, and the loop has it back within $\pm 1\,\mathrm{m/s}$ by 16.7 s.

> **TRY IT IN THE APP** **4 · Disturbance Rejection**
>
> Load the case and press Run. Watch the speed trend when the road turns uphill at 5 s. The true speed falls by 1.21 m/s, reaches its lowest at 12.1 s, and the loop has it back within 1 m/s of the setpoint by 16.7 s, while the pedal trend climbs to carry the hill.

In case 5 the same grade with the pedal untouched slows the vehicle to 14.70 m/s by 60 s on its way to 12.59 m/s, while a pedal of 27 % holds 25.2 m/s. The completion rule of case 5 credits only a pedal that holds the speed in the band once settled. A pedal of 26 or 27 % set at 5.1 s is credited at 8.1 s. A pedal of 20 %, which would let the speed fall toward 18.7 m/s, is never credited.

Cases 6 and 7 scale both tuning terms. The conservative case rises from 10 to 90 % of its step in 40.15 s with no overshoot, and the aggressive one in 2.9 s with 8.9 % overshoot of the true speed. The default tuning on the same step gives 7.0 s and 3.5 %. The scorecard counts 88 and 455 % of pedal travel over the 35 s run, against 224 % for the default tuning. Case 8 holds the pedal on its stop from the execution at 5 s, and the speed climbs toward the 36.89 m/s that full pedal holds on the weaker engine, 3.11 m/s short of the setpoint. The challenge is judged on the case as given, so a changed model parameter or disturbance voids the attempt.

> **TRY IT IN THE APP** **6 · Conservative Tuning**
>
> Run the case and time the climb to the new setpoint: 10 to 90 % of the step takes 40.15 s, with no overshoot. Then load 7 · Aggressive tuning and run it. The same step takes 2.9 s with 8.9 % overshoot of the true speed, and the scorecard counts 455 % of pedal travel against 88 %.

2.7 The Predictive Controller on the Shipped Cases

Unlike the experiments of Section 2.6, which start from case 1, Table 2.7 runs the shipped cases 3, 4, 8 and 9 over 60 s and the challenge over 30 s, with every value read on the 0.1 s log as the scorecard reads it. Each controller is selected before the run starts, so every run of a case meets the same events and noise. The PID keeps the shipped $K_c = 6\,\%\,\mathrm{s/m}$, $\tau_I = 12\,\mathrm{s}$ and $\tau_D = 0$, except that the challenge ships $K_c = 1.2\,\%\,\mathrm{s/m}$ and suggests $K_c = 6$. The MPC keeps the settings of Section 2.5.4, with the

Table 2.7 Five Cases Under the Shipped PID and Under the Predictive Controller, With the Same Speedometer Noise in Every Run of a Case

Case	Controller	IAE (m)	IAE_v	e_{max} (m/s)	OS	$t_{0.3}$ (s)	Travel (%)	Limit (%)
3, setpoint	PID	18.76	17.23	—	0.145	8.3	363.3	0
	MPC	7.10	4.48	—	0.172	1.5	286.6	0.8
4, grade	PID	23.61	22.67	1.205	—	26.1	334.6	0
	MPC	4.82	2.22	0.389	—	2.0	138.5	0
8, limit	PID	335.66	335.33	—	—	—	84.0	91.7
	MPC	335.85	335.54	—	—	—	62.2	91.7
9, mapped hill	PID	23.59	22.28	1.192	—	25.7	335.3	0
	MPC, no map	4.87	2.45	0.423	—	2.3	139.3	0
	MPC	4.25	1.66	0.111	—	0	131.4	0
10, challenge	PID, $K_c = 1.2$	64.69	64.42	3.256	—	—	35.8	0
	PID, $K_c = 6^a$	19.94	19.69	1.205	—	—	176.3	0
	MPC	2.76	1.40	0.389	—	2.0	83.0	0

IAE uses the speedometer reading, as the scorecard does, and IAE_v the true speed. e_{max} is the largest true deviation after the grade begins, OS the largest true overshoot of a new setpoint, $t_{0.3}$ the time from the first event until the true speed stays within 0.3 m/s of setpoint (a dash if it never does within the run), Travel the summed absolute pedal change and Limit the time on a pedal stop as a share of the run. [a]Suggested, with $\tau_I = 12$ s.

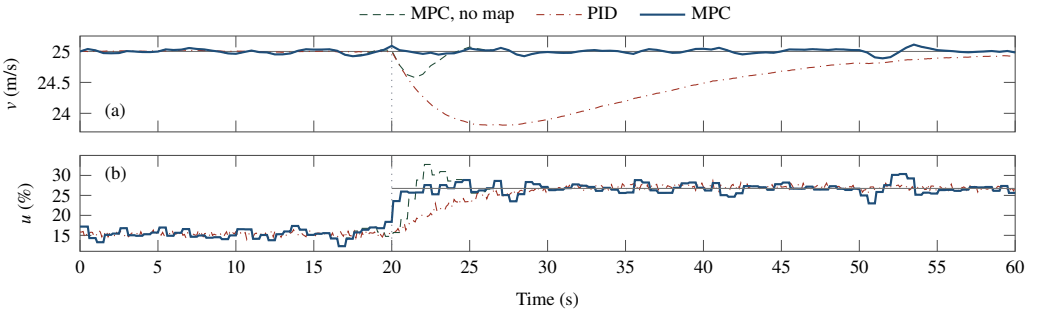


Figure 2.4 Case 9 under the predictive controller with the map (thick solid curve) and without it (dashed curve) and under the shipped PID (dash-dotted curve), with identical speedometer noise, from the 0.1 s log. A 5 % grade begins at 20 s (dotted vertical line). (a) True speed, with the setpoint as a thin gray line. (b) Pedal position, with the 26.74 % that holds the setpoint on the grade as a thin gray line.

map on only in case 9, which is also run without it. The PID has no feedforward, so in case 9 the MPC without the map is the like-for-like comparison.

In case 3, a setpoint step taken where the model was identified, the first plan puts the pedal on its 100 % stop for one decision period, where the proportional kick of the PID reaches 42.6 %. The true speed then covers 10 to 90 % of the step in 1.2 s against 7.0 s and overshoots by 0.172 against 0.145 m/s, and the pedal travels 21 % less.

Case 4 gives the MPC no map, so the grade reaches it only as lost speed. The estimate $\hat{\delta}$ books half of the 11.43 % pedal deficit within 1.2 s and 90 % within 2.2 s, and the true speed bottoms at 24.61 m/s 1.5 s after the grade, where the PID reaches 23.80 m/s after 7.1 s. The true-speed IAE of the PID, 22.67 m, is near the 22.85 m that Equation (2.10) fixes for this 342.8 N load whatever K_c is, while the estimate, integral action routed through the model, escapes that relation with 2.22 m and 41 % of the travel.

Case 8 is a tie. With the thrust at 10 N/% the setpoint lies above the 36.89 m/s that full pedal can hold, so both controllers keep the pedal on its stop from 5 s and reach 36.686 m/s at 60 s, the MPC with a scorecard IAE 0.06 % higher.

Case 9 cruises on the flat until a 5 % grade begins at 20 s, as shown in Figure 2.4. With the map, the pedal stays more than 0.5 % above its quiet level from 17.6 s and reaches 18.37 % at 19.9 s, so the

car meets the grade 0.09 m/s above its setpoint, and one second later it is at 25.95 % of the 26.74 % that the grade needs. The true speed then stays between 24.89 and 25.11 m/s. Without the map the MPC bottoms at 24.58 m/s and the PID at 23.81 m/s. Relying on its estimate alone, the MPC thus cuts the true-speed IAE from 22.28 to 2.45 m, and the map removes another 0.79 m, after which the true error averages 0.027 and 0.028 m/s before and after the grade. That remainder is the filter's response to the speedometer, each reading error of σ_n shifting $\hat{\delta}$ by $\ell_\delta \sigma_n = 0.28\,\%$, while the PID, whose white pedal noise the lag smooths, averages 0.0067 m/s before the grade. Of the scorecard IAE of 4.25 m, 3.92 m is the speedometer noise itself.

A static feedforward of $-p$ on the shipped PID, run outside the application, which offers none, cancels the grade exactly on the default vehicle, because pedal and grade enter Equation (2.1) alike, and reaches a true-speed IAE of 0.37 m. The MPC spreads the same pedal around the grade because κ prices each move. With 500 kg of cargo, when the map supplies only 70 % of the pedal the grade needs, the order reverses, 9.79 m for the feedforward PID against 1.83 m for the MPC with the map, whose estimate makes up the rest. The map then misses 30 % of the pedal, yet the lowest speed under the MPC with the map falls only from 24.89 to 24.84 m/s.

In the challenge the MPC reproduces its first 30 s of case 4 exactly, because it does not read K_c, and its largest measured error in the judged final third is 0.303 m/s, against 3.52 m/s for the shipped K_c, which fails, and 0.918 m/s for the suggested tuning. The MPC computed 660 plans and rejected none. A bound was active only in the first plan after the step of case 3 and in 110 plans of case 8, so cases 4, 9 and 10 were won by prediction and estimation alone. The mean solve time, a wall-clock value from one desktop computer, was 0.08 to 0.31 ms under Node.js and 311 and 325 µs in Chrome over two runs of case 9, never above 0.065 % of one core in real time. The MPC wins cases 3, 4, 9 and 10 and ties case 8.

2.8 Estimator Choice and Relay Tuning

The runs of this section last 180 s, the challenge 30 s, with every setting chosen before the run and the shipped noise sequence, so their integrals are longer than those of Table 2.7. With no bound active, Equation (2.23) is the least-squares problem that the filter solves recursively, and the two estimates coincide. The largest gap between them over 1799 executions of case 9 is $4.5 \times 10^{-10} \sigma_n$ in $\hat{\xi}$ and 7.7×10^{-10} % in $\hat{\delta}$. That residue comes from the Riccati iteration that designed the filter, which stops at a relative change of 10^{-11}. Iterated to its fixed point, the covariance brings the difference down to $1.4 \times 10^{-14} \sigma_n$. Each of the ten cases, run under the MPC once with each estimator, gives pedal records that agree within 2.0×10^{-9} %, and no window was rejected. The bound never comes close, since the lowest speed reached in any case is 12.68 m/s, in the manual case 5.

Judged against the true speed, which only the measuring script reads, both estimators place the speed of case 9 with a root-mean-square error of 0.030 m/s and a largest error of 0.114 m/s, inside the 0.08 m/s standard deviation of the speedometer itself. An estimate took a median of 248 µs and a 99th percentile of 501 µs under Node.js over 16,490 estimates. In Chrome, on the frame loop of the application during case 9, the estimates took 0.60 and the plans 0.09 percent of one processor core in real time and 6.02 and 0.47 percent at ten times real time, the larger of two runs taken while other work held the load average of the same computer at about two. The second figure is within a factor of two of the tenth of a core allowed for estimating and planning together.

The relay was pressed 6 s into case 3, run with its setpoint step removed so that the vehicle holds 25 m/s, and swung the pedal between 7.9 and 22.9 %. It converged after three cycles and 6.15 s on $a = 0.369\,\mathrm{m/s}$, $D = 0.46$, $P_u = 2.89\,\mathrm{s}$ and $K_{cu} = 33.9\,\%\,\mathrm{s/m}$, which Equation (2.24) turns into $K_c = 10.6\,\%\,\mathrm{s/m}$ and $\tau_I = 6.36\,\mathrm{s}$ against the shipped 6 and 12 s. Ten further noise seeds all converged, with K_{cu} from 34.1 to 60.5 and P_u from 1.93 to 3.29 s. None of these is the ultimate point of the vehicle. Equation (2.19) sampled with a zero-order hold at Δt_c reaches -180° only at the Nyquist frequency, where its ultimate gain is $(1+\phi)/\gamma = 469\,\%\,\mathrm{s/m}$ at a period of 0.2 s, because a first-order lag with no dead time carries no phase crossover of its own. The relay instead cycles where its

Table 2.8 Shipped and Relay-Tuned PI Against the Predictive Controller on Cases 3 and 4 Over 180 s, With the Same Speedometer Noise in Every Run

Case	Controller	IAE	IAE_v (m)	IAE_v^0	$e_{\max}$ (m/s)	Travel (%)
3, setpoint	PI, shipped	26.48	18.36	17.4	—	1010
	PI, relay	21.32	12.69	11.3	—	1799
	MPC	15.24	7.53	—	—	511
4, grade	PI, shipped	31.35	23.62	22.9	1.205	981
	PI, relay	17.22	8.51	6.86	0.708	1744
	MPC	12.96	5.17	—	0.389	364

Shipped PI $K_c = 6\,\%\,\text{s/m}$, $\tau_I = 12\,\text{s}$, relay PI $K_c = 10.6\,\%\,\text{s/m}$, $\tau_I = 6.36\,\text{s}$. IAE is the scorecard's on the speedometer reading, IAE_v the true-speed IAE on the 0.05 s step, IAE_v^0 the same with $\sigma_n = 0$, and $e_{\max}$ the largest true deviation after the grade. The MPC rows are identical with either estimator.

band puts it, as it does on the blending tank of Chapter 3. On the noise-free model it settles at $K_{cu} = 107\,\%\,\text{s/m}$ and $P_u = 3.2\,\text{s}$, and the application with the noise set to zero returns 103 and 3.20 s. The default swing is only 1.54 times the band, so $\psi = 40.6°$, and over the ten seeds ψ runs from 40.8 to 56.8°, every run past the flag of Section 2.5.6. The noise then lowers K_{cu} by a further factor of three.

The measured constants still make the faster loop on this vehicle, as listed in Table 2.8. They raise K_I from 0.5 to 1.67 %/m and ω_n from 0.146 to 0.267 rad/s, and lower ζ of Equation (2.9) from 1.058 to 0.948. For the 342.8 N grade of case 4, Equation (2.10) then predicts 6.86 m in place of 22.85 m, and the noise-free runs give 6.86 and 22.9 m. The pedal travels 1.78 times as far on both cases, close to the ratio of the proportional gains, 1.77, because the speedometer noise, amplified by the proportional gain, carries 95 to 99 % of the travel in these runs, the noise-free runs traveling 48.1 and 90.5 % on case 3 and 12.4 and 13.7 % on case 4. The MPC with either estimator remains ahead on both cases, with 5.17 m of true-speed IAE on case 4 at a fifth of the travel of the relay-tuned loop.

The relay result also moves with the state from which it is started. Started at 60 s of case 3, once the loop has settled at its new setpoint, the relay returns K_{cu} of 46.0 with the declared noise and 101 % s/m with the noise set to zero, beside the 469 % s/m of the sampled model. The second value calls for a K_c of 31.7, which the top of its range holds at 30. The first gives $K_c = 14.4\,\%\,\text{s/m}$ and $\tau_I = 6.46\,\text{s}$. Run on case 4 for 180 s, that pair lowers the true-speed IAE from 23.6 to 6.98 m at 2.4 times the pedal travel, against noise-free values of 22.85 and 5.13 m from Equation (2.10).

2.9 Technology in Practice

The loop of Section 2.3 reads a speedometer and moves a pedal. Figure 2.5 draws it with real devices in the place of the schematic's symbols.

What Measures It

In a cruise control system the measurement noise comes from the electronics that convert pulses measured on a rotating shaft into a velocity signal [6]. On a road vehicle the pulses commonly come from a variable-reluctance pickup facing a toothed wheel [20], described below. The white noise of 0.08 m/s in Equation (2.4) stands for that conversion, while the bias from tire wear and the quantization by the tooth count that Section 2.3.4 leaves out are what a real pickup adds.

> **TECHNOLOGY** **Sensor: The Wheel-Speed Pickup**
>
> A variable-reluctance pickup sits close to a toothed ferrous wheel. Each passing tooth changes the magnetic flux through its coil, so the output is an alternating voltage whose frequency and amplitude are both proportional to the speed of rotation. The pickup needs no external power and detects iron up to about 2.5 mm away. It cannot sense zero speed, and below about 5 km/h it is very susceptible to noise. Differential Hall-effect sensors resist interference better but need a minimum switching frequency [20].

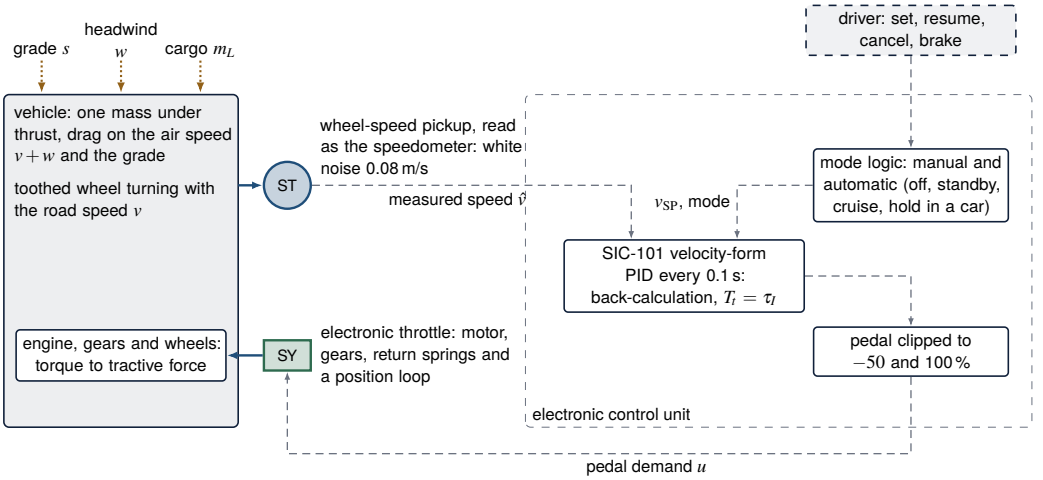


Figure 2.5 The loop of the chapter with real devices in place of the schematic's symbols. A wheel-speed pickup ST reads the road speed, the controller SIC-101 on the vehicle's electronic control unit computes a pedal demand under the driver's mode logic, and the electronic throttle SY turns it into engine torque. Notice that the three disturbances act on the vehicle and none of them is measured, so the controller learns of a hill, a wind or a load only from the speed it loses.

What Moves It

The final element of a cruise loop is the throttle, which sets the engine torque against disturbance forces from the slope of the road, the rolling resistance and the aerodynamic forces. A touch of the brake disconnects the cruise control altogether [6]. The signed pedal of Equation (2.1), clipped at -50 and 100 % with a force linear in its position, stands for throttle, engine and gears together, without their dead time or the throttle's own position loop.

> **TECHNOLOGY** **Final Element: The Electronic Throttle**
>
> The cruise controller sends its output to an actuator that sets the throttle, and the throttle sets the torque the engine delivers through the gears and wheels [6]. An electronic throttle is an electromechanical actuator that sets the angle of the throttle plate, and so the air flow into the engine, with a DC motor, reduction gears, return springs and a throttle position sensor. Its springs and friction are nonlinear and its parameters vary with operating conditions, so the plate angle is held by a position loop of its own. One laboratory throttle closed that loop every 0.01 s on a microcontroller [21].

Where the Controller Runs

Control in a car is executed by electronic control units, microprocessors with input and output interfaces, and a modern car has more than 100 of them. Computer control entered the car in the late 1970s to meet stringent emission requirements [6]. Loops are often nested, with the setpoint of one loop based on the variable of another, and they run with cycle times from milliseconds to minutes [22]. The cruise loop is such an outer loop over the throttle position, which one laboratory throttle closed every 0.01 s [21], and the PID of this chapter executes every 0.1 s. A digital PID reads the sensor, computes and sends the output, and only then updates its states, which keeps the delay short. It usually filters the derivative with a time constant of the derivative time divided by 5 to 20, and it resets its integrator at a tracking gain that is a multiple of the integral gain but not so large that noise resets it [6]. The velocity form of Section 2.5 acts on an unfiltered derivative and tracks with the integral time.

> **IN PRACTICE The Logic Around the PI Loop**
>
> A cruise controller is typically a PI loop inside mode logic. A finite state machine with four modes, off, standby, cruise and hold, answers the driver. Set takes the speed at the moment the button is released as the setpoint. Pressing the gas pedal speeds the car up, and it returns to the set speed when the pedal is released. A touch of the brake puts the loop in hold, which remembers the setpoint, and resume brings it back. Switching modes must not cause a transient, so in manual the controller's output tracks the manual input [6]. The application of this chapter keeps only the manual and automatic modes of that logic.

What Is Modernizing It

Sensing ahead is the first development. Adaptive cruise control, based on radar sensors, maintains a constant distance to the car in front [6], and cooperative adaptive cruise control adds wireless links over which vehicles share their speed, acceleration and position [23]. Preview is the second. An adaptive cruise controller built on model predictive control can read the road grade from a prior map of the road to predict the vehicle's motion and improve safety, comfort, tracking and energy use [24], as the predictive controller of Section 2.5.4 reads the hill of case 9. Software is the third. A partnership of vehicle makers, suppliers and software companies formed in 2003 has created standards for a common automotive software platform, so that functions become modular, transferable and reusable [6].

2.10 Limitations

The application is built for teaching rather than for vehicle design. A point mass with linear thrust cannot represent an engine map, a gearbox, a torque converter or a traction limit, so it must not be used to predict acceleration times, gradeability or fuel consumption. Rolling resistance is absent altogether, which makes every coast-down in the application longer than a real one and removes the resistance that dominates below about 10 m/s. Braking is a speed-independent negative thrust with no wheel dynamics, so nothing in the application can produce lock-up, anti-lock intervention or a stopping distance worth quoting, the subject of Volume 7, Chapter 4. The road has one grade and one wind at a time, both applied everywhere at once, so a real hill that arrives as a ramp is easier to reject than the step tested here. The speedometer carries white Gaussian noise and nothing else, so the tire-radius bias and encoder quantization that dominate a real wheel-speed measurement are not represented. Noise is drawn from a seeded generator that restarts at every case load and Reset, so every run of a case sees the same noise sequence and the spread of a noisy statistic across realizations cannot be seen from inside the application. There is no lead vehicle, no spacing loop and no lateral motion, so the adaptive cruise control and platooning questions that the literature of Section 2.2 is mostly about cannot be posed here at all. Four features were kept deliberately and are not candidates for removal, the quadratic drag law, the signed relative air speed, the single signed pedal with its two stops and the 0.1 s execution rate, because each of them produces something a tuning decision has to answer for. The operating region in which every remaining assumption breaks down is given in Section 2.3.4.

The model of the predictive controller, a first-order pair frozen at 25 m/s on the flat with 700 kg, omits the quadratic drag (its gain is 1.41 times the vehicle's at 35 m/s), the mass and the engine (three times too strong in case 8, which it only ties). Its map is exact, unlike any road database, and a feedforward of the same map does better on the default vehicle. In quiet running it lets the true speed wander four times as far as the PID, and every comparison here uses a single noise realization. The MHE bounds only the states of that same frozen model, and its one bound lies more than 12 m/s below any speed a case reaches, so on this vehicle it adds 0.60 % of a core in real time and nothing else. The relay measures the band and the noise more than the vehicle, because a lag with no dead time has no phase crossover for it to find.

OPEN CHALLENGE **Cruise Control in a String of Cars**

Cooperative adaptive cruise control lets vehicles share their speed, acceleration and position over wireless links, so that they can travel in platoons at shorter gaps. The links bring communication delay, a limited range, packet loss and sampling intervals, and real traffic keeps changing which vehicles talk to which. Many control methods have been proven in theory and in simulation, but few have been tested on real roads, and a system of several vehicles is hard to test over enough conditions and distance before it reaches the market [23]. Progress would be a controller whose spacing stays safe, with errors that do not grow down the string, when messages arrive late or not at all.

2.11 Discussion

The energy a vehicle spends against aerodynamic drag over a distance is the drag force times that distance, so it scales with the square of the speed held and is independent of how long the trip takes. The loop of this chapter therefore has an economic and environmental dimension that most regulatory loops do not. Steadier speed and lower speed both reduce fuel use, and the driving behavior changes grouped under eco-driving have been assessed as one of the cheapest available reductions in transport emissions [25]. A cruise controller that rejects grade and wind with a smaller pedal excursion is doing exactly the thing that assessment recommends, and the pedal travel row of Table 2.6 is the quantity that measures it. The trade is visible in the derivative results of Section 2.6, where raising τ_D from 0 to 0.2 s increases pedal travel from 5.5 to 24.7 % per second and makes the speed regulation slightly worse.

The safety dimension pulls in the same direction and is quantified separately. Systems that hold a vehicle at or below the posted limit, rather than at a driver-chosen setpoint, have been evaluated for their accident savings and their cost, and the estimated benefit is large enough that the intervention is worth mandating on cost grounds alone [26]. The mechanism the application makes visible is the one such a system has to live with, because the achievable set of speeds is bounded by the pedal limit and by the road load, and Section 2.6 shows a vehicle that cannot hold 40 m/s up an 8 % grade into a 15 m/s headwind whatever the controller demands. Adaptive cruise control resolves the conflict between a speed target and a spacing target by switching between two loops, and the speed loop it switches back to is the one of this chapter [8]. Whether the gains of that inner loop should be scheduled on the operating point is a question this application answers with a measurement rather than an assertion [3], and the answer for a plant whose gain and time constant fall together is that the schedule buys less than the gain variation alone suggests. The neutralizer of Chapter 6 is the opposite case. Its gain changes by four orders of magnitude across one valve stroke, and no single set of constants serves it.

OPEN CHALLENGE **Keeping Loops Well Tuned**

A large industrial plant may run hundreds of PID regulators [19]. More than 1730 tuning rules have been collected, yet industrial studies agree that a significant fraction of PID controllers are poorly tuned [6]. A loop tuned once drifts out of tune as its plant moves to another operating point, as the damping of this chapter's loop moves with speed. Progress would be tools that notice when a loop has drifted from its tuning and correct it from normal operating data, without a test that upsets production.

2.12 Conclusion

This chapter studied an automobile cruise control loop in which the load is quadratic in speed. The simulator integrates one state under a longitudinal force balance with thrust linear in pedal position, drag acting on the signed relative air speed, a gravity component set by the road grade, a pedal clipped at -50 and 100 %, and a speedometer that adds 0.08 m/s of Gaussian noise. The linearized description is first order with a gain and a time constant that both shrink in inverse proportion to the cruise speed while their ratio holds at the thrust-to-mass ratio of 0.0429 m/(s^2 %), and pedal steps run in the application follow that scaling to within the width of a finite step, with the identified gain down

by a factor of 3.26 and the identified time constant down by 3.09 between 9.04 and 34.99 m/s against the 3.87 the ratio of starting speeds predicts, the shortfall being the secant of a step that is 41 % of the starting speed at the slow end and 3 % of it at the fast end.

Adding 600 kg of cargo instead of none multiplies the identified time constant by 2.201 against a mass ratio of 2.200 and leaves the identified gain unchanged at 0.799 m/(s %). The three load disturbances separate cleanly, because cargo has no steady-state effect, a headwind removes its own magnitude from the road speed to four decimal places, and a 4 % grade costs a proportional loop, modeled with the integral frozen, 1.417 m/s at 10 m/s and 1.190 m/s at 35 m/s with identical gains. Under integral action the same grade is rejected with an integrated error of 18.29 m at every cruise speed, because that quantity is fixed by the disturbance force and the integral gain alone, while the peak deviation falls from 1.029 to 0.914 m/s as the damping ratio rises from 0.950 to 1.129 at a natural frequency that does not move from 0.146 rad/s.

The default controller is the velocity form of PID with the derivative on the measurement, bumpless transfer in both directions and back-calculation anti-windup with a tracking time equal to the integral time. That choice relaxes the integral to the stop during a saturation, so after 290 s on the stop the pedal stays there until the speed crosses the setpoint and the speed overshoots by 7.2 %, against 32 % when the correction is removed. A linear predictive controller keeps the true speed within 0.11 m/s of setpoint on the grade of case 9 with a map and 0.42 m/s without, against 1.19 m/s for the PID. Its moving horizon estimator reproduces that controller's runs to 2×10^{-9} % of pedal, and a relay test returns $K_{cu} = 33.9\,\%\,\mathrm{s/m}$ where the sampled vehicle has 469, yet its PI constants lower the true-speed IAE of the grade from 23.6 to 8.51 m at 1.8 times the pedal travel.

A fuller model would add rolling resistance, a gear-dependent tractive force, a lead vehicle for spacing control, and a scheduled gain so that the measured damping variation can be removed rather than merely observed. The result to carry into the chapters that follow is that the plant a step test finds depends on how fast the vehicle was going when the test was run.

KEY POINTS

- Equal steps of the pedal buy less and less speed as the car goes faster, because air drag grows with the square of the speed.
- The linearized gain and time constant both shrink in inverse proportion to the cruise speed, at the fixed ratio of thrust to mass.
- Under one PI tuning the natural frequency does not move with the cruise speed, and only the damping ratio does, from 0.950 to 1.129.
- Integral action rejects a 4 % grade with an integrated error of 18.29 m at every cruise speed.
- Back-calculation bounds windup: after 290 s on the stop the speed overshoots by 7.2 %, against 32 % without it.

Where else this applies. A gain and a time constant that move together with the operating point, at a fixed ratio, also shape the gravity-drained tank of Volume 1, Chapter 2.

References

[1] R. Rajamani, *Vehicle Dynamics and Control*, ser. Mechanical Engineering Series. Boston, MA: Springer US, 2012.

[2] W. Hucho and G. Sovran, "Aerodynamics of road vehicles," *Annual Review of Fluid Mechanics*, vol. 25, pp. 485–537, 1993.

[3] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.

[4] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle, III, *Process Dynamics and Control*, 4th ed. Hoboken, NJ: John Wiley & Sons, 2016.

[5] B. W. Bequette, *Process Control: Modeling, Design, and Simulation*. Upper Saddle River, NJ: Prentice Hall, 2003.

[6] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*, 2nd ed. Princeton, NJ: Princeton University Press, 2021.

[7] P. A. Ioannou and C. C. Chien, “Autonomous intelligent cruise control,” *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 657–672, 1993.

[8] L. Xiao and F. Gao, “A comprehensive review of the development of adaptive cruise control systems,” *Vehicle System Dynamics*, vol. 48, no. 10, pp. 1167–1192, 2010.

[9] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, “Automated vehicle control developments in the PATH program,” *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, 1991.

[10] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[11] J. C. Maxwell, “On governors,” *Proceedings of the Royal Society of London*, vol. 16, pp. 270–283, 1868.

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

[13] M. V. Kothare, P. J. Campo, M. Morari, and C. N. Nett, “A unified framework for the study of anti-windup designs,” *Automatica*, vol. 30, no. 12, pp. 1869–1883, 1994.

[14] 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.

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

[16] E. Hellström, M. Ivarsson, J. Åslund, and L. Nielsen, “Look-ahead control for heavy trucks to minimize trip time and fuel consumption,” *Control Engineering Practice*, vol. 17, no. 2, pp. 245–254, 2009.

[17] D. Goldfarb and A. Idnani, “A numerically stable dual method for solving strictly convex quadratic programs,” *Mathematical Programming*, vol. 27, no. 1, pp. 1–33, 1983.

[18] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, “Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations,” *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, 2003.

[19] K. J. Åström and T. Hägglund, “Automatic tuning of simple regulators with specifications on phase and amplitude margins,” *Automatica*, vol. 20, no. 5, pp. 645–651, 1984.

[20] W. Hernandez, “Improving the response of a wheel speed sensor by using a RLS lattice algorithm,” *Sensors*, vol. 6, no. 2, pp. 64–79, 2006.

[21] V. D. Phan, V. C. Le, V. N. Dinh, D. T. Duong, and S. P. Ho, “A novel adaptive terminal sliding mode tracking control of engine electronic throttle system using an enhanced observer under composite disturbances,” *Sensors*, vol. 26, no. 19, p. 6070, 2026.

[22] K. Stouffer, M. Pease, C. Tang, T. Zimmerman, V. Pillitteri, S. Lightman, A. Hahn, S. Saravia, A. Sherule, and M. Thompson, “Guide to operational technology (OT) security,” National Institute of Standards and Technology, Gaithersburg, MD, Special Publication 800-82r3, 2023.

[23] Z. Wang, G. Wu, and M. J. Barth, “A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2884–2891.

[24] R. Firoozi, S. Nazari, J. Guanetti, R. O’Gorman, and F. Borrelli, “Safe adaptive cruise control with road grade preview and V2V communication,” arXiv preprint arXiv:1810.09000, 2018.

[25] J. N. Barkenbus, “Eco-driving: An overlooked climate change initiative,” *Energy Policy*, vol. 38, no. 2, pp. 762–769, 2010.

[26] O. M. J. Carsten and F. N. Tate, “Intelligent speed adaptation: accident savings and cost-benefit analysis,” *Accident Analysis & Prevention*, vol. 37, no. 3, pp. 407–416, 2005.

2.13 Exercises

2.1 ○ Concept check · 2 min

Pedal step tests of the same small size are run on the default vehicle from a steady cruise at 10 m/s and at 35 m/s. How do the process gain K_u and the time constant τ fitted at the higher speed compare with those fitted at the lower speed?

(a) The gain is lower and the time constant is the same, because the mass has not changed
(b) The gain is the same and the time constant is lower, because drag acts on the dynamics alone
(c) Both are higher, because the slope of the drag curve grows with the cruise speed
(d) Both are lower by nearly the same factor, so their ratio stays at the thrust-to-mass ratio

2.2 Concept check · 1 min

The vehicle cruises on a flat road in Manual with the pedal fixed, and 400 kg of cargo is added. What does the model predict?

(a) The resting speed is unchanged, and a later pedal step takes longer to settle
(b) The resting speed falls, because a heavier vehicle needs more thrust to hold its speed
(c) The resting speed falls, and a later pedal step settles in the same time as before
(d) The resting speed is unchanged, and a later pedal step settles in the same time as before

2.3 Concept check · 2 min

A 2 % pedal step from a steady 9.04 m/s returns a fitted gain of 1.871 m/(s %). The linearization at the starting speed gives 2.259 m/(s %). Which explanation is right?

(a) The fit is biased low by the speedometer noise, which is largest at low speed
(b) The step raises the speed by 41 %, so the secant gain falls below the tangent gain
(c) The control period adds a dead time that the first-order fit reads as lost gain
(d) The Euler integration step loses a share of the response when the vehicle is slow

2.4 Theory and calculation · 5 min

The default vehicle has a total mass $M = 700\,\mathrm{kg}$, a thrust parameter $F_p = 30\,\mathrm{N/\%}$ and a drag constant $k = 0.735\,\mathrm{kg/m}$. It cruises at 30 m/s on a flat road with no wind.

(a) Find the pedal that holds this speed. *Checkpoint:* 22.05 %.

(b) Find the process gain and the time constant of the linearized model at this speed. *Checkpoint:* 0.680 m/(s %) and 15.87 s.

(c) Find the ratio of the gain to the time constant and compare it with the value at 25 m/s. *Checkpoint:* 0.0429 m/(s^2 %) at both speeds.

2.5 Theory and calculation · 8 min

The default vehicle ($M = 700\,\mathrm{kg}$, $F_p = 30\,\mathrm{N/\%}$, $k = 0.735\,\mathrm{kg/m}$) holds 25 m/s under the default PI tuning, $K_c = 6\,\%\,\mathrm{s/m}$ and $\tau_I = 12\,\mathrm{s}$. The road grade steps from 0 to 5 %.

(a) Find the load force along the road and the pedal increment that rejects it in steady state. *Checkpoint:* 342.8 N and 11.43 %.

(b) Find the integral of the error over the whole rejection, and say whether it changes at a cruise speed of 35 m/s. *Checkpoint:* 22.85 m at either speed.

(c) Integral action is switched off before the grade arrives. Estimate the steady offset. *Checkpoint:* 1.58 m/s.

2.6 Theory and calculation · 5 min

A setpoint is worth tuning for only if the pedal can hold it. Use the steady force balance with $k = 0.735\,\mathrm{kg/m}$ and the pedal at its 100 % stop.

(a) Scenario 8 lowers the thrust parameter to 10 N/% on a flat road with no wind. Find the highest speed the vehicle can hold, and say whether a setpoint of 40 m/s is reachable. *Checkpoint:* 36.89 m/s, so it is not.

(b) With the default 30 N/%, the vehicle carries 600 kg of cargo ($M = 1100$ kg) up an 8 % grade into a 15 m/s headwind. Find the highest road speed it can hold. *Checkpoint:* 38.96 m/s.

2.7 App challenge · 9 min

Load scenario 2, Open-loop actuator step, and change no setting. The pedal steps from 15.31 to 35.31 % at 5 s. Run to 35 s with the default seed and export the CSV file.

(a) Read the true speed, column `true_pv`, at 35 s. *Checkpoint:* 36.56 m/s, within 0.05 m/s.

(b) Compute the speed at which this pedal settles and the gain of the whole step, the settled speed change divided by the pedal change. *Checkpoint:* 37.96 m/s and 0.648 m/(s %).

(c) Find in the file the time after the step at which the true speed has covered 63.2 % of the settled change, and compare it with the time constant of the linearized model at 25 m/s. *Checkpoint:* 14.1 s, within 0.3 s, against 19.05 s.

2.8 App challenge · 8 min

Load scenario 4, Disturbance rejection. The road grade steps from 0 to 5 % at 5 s. Use the default seed and run each case for 60 s.

(a) Run the scenario as shipped. Read the IAE tile of the scorecard, and find in the exported file the lowest true speed and its time. *Checkpoint:* IAE 23.6 m, within 0.2 m. Lowest speed 23.80 m/s at 12.1 s.

(b) Press the reload button, set the Integral time to its top, 120 s, which switches integral action off, and run again. Read the true speed at 60 s. *Checkpoint:* 23.41 m/s, within 0.03 m/s.

Compare both results with the values that Equations (2.10) and (2.11) predict for this grade.

2.9 App challenge · 8 min

Load scenario 9, MPC with the hill on the map. A 5 % grade begins at 20 s. Use the default seed and run each case for 60 s. For each run read the IAE tile of the scorecard and find in the exported file the lowest true speed after 20 s.

(a) Run the scenario as shipped, with the Controller at MPC and Road ahead on. *Checkpoint:* 4.25 m and 24.89 m/s.

(b) Set Road ahead off, press Reset and run again. *Checkpoint:* 4.87 m and 24.58 m/s.

(c) Set the Controller to PID, press Reset and run again. *Checkpoint:* 23.59 m and 23.81 m/s.

Accept an IAE within 0.1 m and a speed within 0.03 m/s. State what the disturbance estimate and the map each contribute.

2.10 Application area · 3 min

The model omits rolling resistance. Take a rolling resistance coefficient of 0.01 on the default vehicle of 700 kg, with $k = 0.735\,\mathrm{kg/m}$ and $F_p = 30\,\mathrm{N/\%}$.

(a) Find the rolling resistance force and the pedal it would add at any speed. *Checkpoint:* 68.6 N and 2.3 %.

(b) Find the speed below which it exceeds the aerodynamic drag, and name one prediction for which the app must not be used as a result. *Checkpoint:* 9.7 m/s.

2.11 Application area · 4 min

A fleet operator compares cruise setpoints of 25 and 30 m/s for a 100 km run on a flat road with no wind, with the default vehicle ($k = 0.735\,\mathrm{kg/m}$).

(a) Find the drag force and the propulsive power at each speed. *Checkpoint:* 459.4 N and 11.5 kW, 661.5 N and 19.8 kW.

(b) Find the energy spent against drag over the run at each speed and the time saved at the higher speed. *Checkpoint:* 45.9 and 66.2 MJ, and 11.1 min.

2.12 Review the agent · 5 min

An AI assistant is asked to improve the cruise controller and replies as follows. “The process gain is $K_u = F_p/(2k\bar{v})$, so it is 2.5 times higher at 10 m/s than at 25 m/s and 1.4 times lower at 35 m/s. A fixed K_c lets the loop gain change by a factor of 3.5 over the speed range. Schedule the gain as $K_c = 6\,\bar{v}/25$ with $\tau_I = 12$ s, which gives 2.4 at 10 m/s and 8.4 at 35 m/s, and the loop will respond in the same way at every speed.” Find the flaw in the proposal and say how it could be checked.