Open Loop Simulation of a Thrust Vector Controlled Rocket (Kalman-TVC)

Implementation of a Gyroscope Model and Kalman Filtering
Course Project Part 3
AESE-404
Joshua Kraus (jjk226@lehigh.edu)

1. Vehicle and Purpose

The "Kalman" thrust vector control (TVC) rocket is a finless solid propellant rocket running on an AeroTech I49N-P motor. The vehicle is approximately 1.44 m long and has a liftoff mass of approximately 1.73 kg. It is steered only by gimballing (tilting) the motor about a pivot that sits $l = 0.46$ m behind the center of gravity (CG). Figure 1 shows the vehicle in CAD, with the aft end, where the motor is gimballed, enlarged.



Figure 1. The Kalman TVC rocket in CAD. The aft end, where the motor is gimballed, is enlarged. The parts are numbered and named under the vehicle.

The purpose of the flight is to demonstrate a fully autonomous controlled ascent using pure thrust vector control and internal reaction systems. The demonstrator vehicle gives us a comprehensive testbed for developing code that can be used to further the research and development of the Lehigh University rocketry team's liquid bipropellant rocket line.

The purpose of this report is to cover the addition of a gyroscope sensor model, sensor processing, and Kalman filter state estimation to the open loop simulation of the Kalman TVC rocket. A controlled ascent needs to know how the rocket is pointed and how fast it is turning, and the flight computer only has what its sensors report, which carries a bias and noise. The thrust curve and the gimbal angles are still chosen in advance and no estimate is used by any input, so the simulation is still open loop. Feedback control is not within the scope of this report.

2. Implemented Equations and Results

2.1 Gyroscope Sensor Model

The sensor chosen is the rate gyroscope of the ICM-42688-P inertial measurement unit (TDK InvenSense, 2021). It is one of the three IMUs in the triple redundant stack of the CUAV Pixhawk V6X flight computer planned for this vehicle (PX4, n.d.). A review of the datasheets of the three IMUs on board, the

BMI088 (Bosch Sensortec, 2018), the ICM-42688-P, and the ICM-20649 (InvenSense, 2016), showed that the ICM-42688-P has the lowest gyroscope noise, with a noise density of 0.0028 deg/s per $\sqrt{\text{Hz}}$ against 0.014 and 0.0175 deg/s per $\sqrt{\text{Hz}}$ for the other two, and it was chosen for that reason. A gyroscope measures the angular rates p, q, and r about the body axes. As these are states that the simulation already calculates, the true signal needs no change to the equations of motion. Bias and noise as sources of error in an inertial sensor must be modeled. The model returns the true rate plus a constant bias plus Gaussian white noise, with a new independent noise draw for every axis at every sample,

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b} + \mathbf{n} \tag{1}$$

where $\boldsymbol{\omega}_m$ is what the gyroscope reports, $\boldsymbol{\omega}$ is the true rate, $\mathbf{b}$ is the bias, and $\mathbf{n}$ is the noise. Each noise sample is drawn from a zero mean normal distribution with standard deviation σ. The sample time is $T_s = 0.01\,\text{s}$, which is 100 Hz and equal to the simulation step. The datasheet gives a rate noise density $\rho_g = 0.0028\,\text{deg/s}$ per $\sqrt{\text{Hz}}$ (TDK InvenSense, 2021). A sensor sampled at 100 Hz can only represent frequencies below half the sample rate, because at least two samples per cycle are needed to see a wave (MathWorks, n.d.). This limit, $1/(2T_s) = 50\,\text{Hz}$, is the Nyquist frequency. Higher frequencies fold down into lower ones, so for example a 70 Hz vibration sampled at 100 Hz would appear as a 30 Hz signal. Thus, the noise standard deviation per sample is

$$\sigma = \rho_g \sqrt{\frac{1}{2\,T_s}} = 0.0028\sqrt{50} = 0.0198\,\text{deg/s} = 3.46 \times 10^{-4}\,\text{rad/s} \tag{2}$$

The bias is $b = 0.5\,\text{deg/s}$ on each axis. This is the datasheet bound on the initial zero rate output and it is used here as the worst case (TDK InvenSense, 2021). Random numbers are generated by MATLAB with a fixed seed such that each run gives the same noise. The sensor is a MATLAB Function block that runs once every T_s. Its key lines are

```
noise = gyro_noise_sigma * randn(3, 1);
omega_measured = omega_true + gyro_bias + noise;
```

For testing purposes, the run starts 2 s before ignition, which is a divergence from the previous iteration of the model. The model has no ground, so for the first 2 s a support force equal to the weight is put through the thrust table and the rocket sits at “rest” on the pad. The support ends 0.014 s after ignition, when the thrust first exceeds the weight, and after that the rates and angles are identical to a run that starts at ignition. The pad period lets the sensor measure its own bias while the rocket is still. This was implemented such that the bias indicated by the datasheet may be measured and corrected for, in anticipation of a bias correction on the pad for implementation on the vehicle. Times in this report are measured from ignition, so the pad period is T−2 to T0 s.

The sensor model was run on the demo case of Section 4.2, a small gimbal sine in pitch and yaw. Figure 2 shows the true and measured pitch rate (a) and the measurement error, the measured value minus the true value (b). Both are the raw sensor with nothing done to the measurement. The roll and yaw axes behave the same way.

- The black line is the true rate and the orange crosses are the measured rate. The crosses sit above the line by the 0.5 deg/s bias, with a small scatter. The roll rate stays at zero, so every cross on that axis is sensor error.
- The mean error on the roll, pitch, and yaw axes is 0.4995, 0.5005, and 0.5001 deg/s against the 0.5 deg/s that was set. The standard deviation is 0.0201, 0.0199, and 0.0197 deg/s against $\sigma = 0.0198$

deg/s, and 67.6, 68.1, and 68.0 percent of the points fall within one σ of the bias, against 68 percent for normal noise.

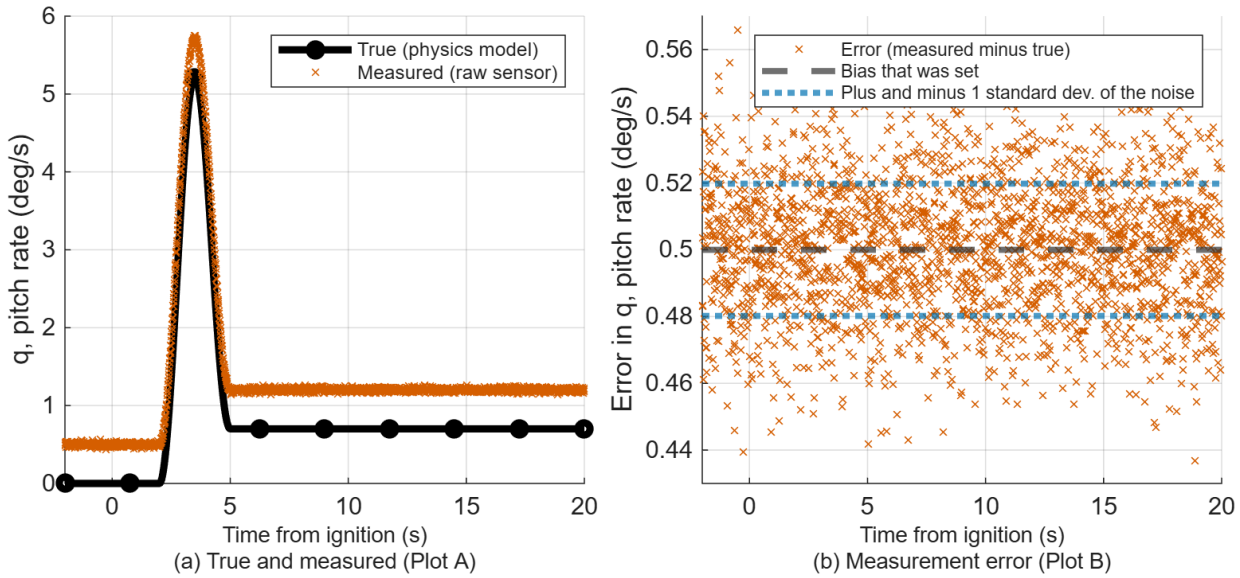


Figure 2. True and measured pitch rate (a) and measurement error (b) for the demo case (Plots A and B). The line is the true rate from the physics model and the crosses are the raw sensor output. The dashed line is the bias that was set and the dotted lines are one standard deviation of the noise above and below it.

2.2 Sensor Processing

Sensor processing has two parts, a bias calibration and a filter. A first order low-pass filter was chosen, with the cutoff frequency as its design parameter. A moving average filter was also built, with the window size as its parameter, so that the two techniques may be compared on the same measurement (Fetzer, 2026b). While the rocket sits on the pad its true rate is zero, so the average of the first N_c readings is the bias plus a little noise, and that average is subtracted from every reading.

$$\hat{b} = \frac{y_1 + y_2 + \cdots + y_{N_c}}{N_c}, \qquad y_k^{\text{cal}} = y_k - \hat{b} \tag{3}$$

The calibration window is 1.5 s at a time step of 0.01 s, giving $N_c = 150$ samples from T−2.0 to T−0.5 s. The pad averages for the roll, pitch, and yaw axes are 0.498, 0.501, and 0.502 deg/s against the 0.500 deg/s that was set. Pad calibration is enabled for all runs in this report.

The low-pass filter is the discrete first order filter below, where z_k is the input, y_k is the output, f_c is the cutoff frequency, and a is the filter gain, a number between 0 and 1 that is worked out from the cutoff and the sample time. A lower cutoff gives a smaller a, so each new reading moves the output less.

$$y_k = y_{k-1} + a\,(z_k - y_{k-1}), \qquad a = \frac{2\pi\, T_s f_c}{2\pi\, T_s f_c + 1} \tag{4}$$

The moving average is a separate technique. It is the mean of the latest w readings, where w is the window size.

$$y_k = \frac{z_k + z_{k-1} + \cdots + z_{k-w+1}}{w} \tag{5}$$

Both filters smooth the newest reading with older data, so the output lags a changing signal. Each setting is measured three ways. The delay is a time, the noise left is the standard deviation of the noise after filtering, which is the raw sensor noise times a noise gain, and the lag error is the largest difference between the filter run on the noise free true signal and the true signal, so it shows what the delay costs on this maneuver. For the two filters the delay τ and the noise gain g are

$$\tau_{LP} = \frac{1}{2\pi f_c}, g_{LP} = \sqrt{\frac{a}{2-a}}$$
$$\tau_{MA} = \frac{(w-1)\,T_s}{2}, g_{MA} = \frac{1}{\sqrt{w}} \quad (6)$$

The low-pass cutoffs are 1, 5, and 20 Hz, which give a = 0.059, 0.239, and 0.557, where a is the filter gain of Equation (4). The moving average windows were chosen so that the two techniques leave the same noise. Setting the two noise gains of Equation (6) equal gives $w = (2-a)/a$, which rounds to windows of 33, 7, and 3 samples for the three cutoffs. The two techniques can then be compared at equal noise.

Figure 3 shows the true, measured, and low-pass filtered pitch rate for three cutoff frequencies, and Figure 4 shows the same with the true signal subtracted. Each row is a separate cutoff applied to the same bias calibrated measurement. The left column is the maneuver, where the true rate is changing. The right column is the pad and the straight flight before the maneuver, where the true rate is zero and the noise can be seen. The gray band is the calibration window. Table 1 lists the delay, the noise left, and the lag error of each setting.

Table 1. Delay, noise left, and lag error of each setting for the pitch rate.

Setting	Delay (ms)	Noise left (deg/s)	Lag error (deg/s)
Raw sensor, no filter	0	0.0196	0
Low-pass, f_c = 1 Hz (a = 0.059)	159.2	0.0034	0.823
Low-pass, f_c = 5 Hz (a = 0.239)	31.8	0.0072	0.173
Low-pass, f_c = 20 Hz (a = 0.557)	8.0	0.0122	0.043
Moving average, w = 33 samples	160.0	0.0034	0.863
Moving average, w = 7 samples	30.0	0.0074	0.164
Moving average, w = 3 samples	10.0	0.0113	0.055

- A lower cutoff leaves less noise and adds more delay. From 5 Hz down to 1 Hz the noise left falls from 0.0072 to 0.0034 deg/s, about half, while the lag error grows from 0.173 to 0.823 deg/s, nearly five times. The 1 Hz line trails the true rate by 159 ms in Figure 3. The 20 Hz line follows the true rate closely, with a lag error of 0.043 deg/s, but it leaves 0.0122 of the 0.0196 deg/s of raw noise.
- The moving average windows of 33, 7, and 3 samples leave about the same noise as the cutoffs of 1, 5, and 20 Hz, and have about the same delay and lag error. Neither technique does better than the other at equal noise.
- The sensor is already quite quiet. The raw noise is 0.0196 deg/s on a maneuver that peaks at 5.2 deg/s, so a filter mostly adds delay here. A filter earns its place when the noise is larger. The calibration removes the bias, so in the right column the processed lines sit on the true line and the raw crosses sit on the pad average of 0.501 deg/s.

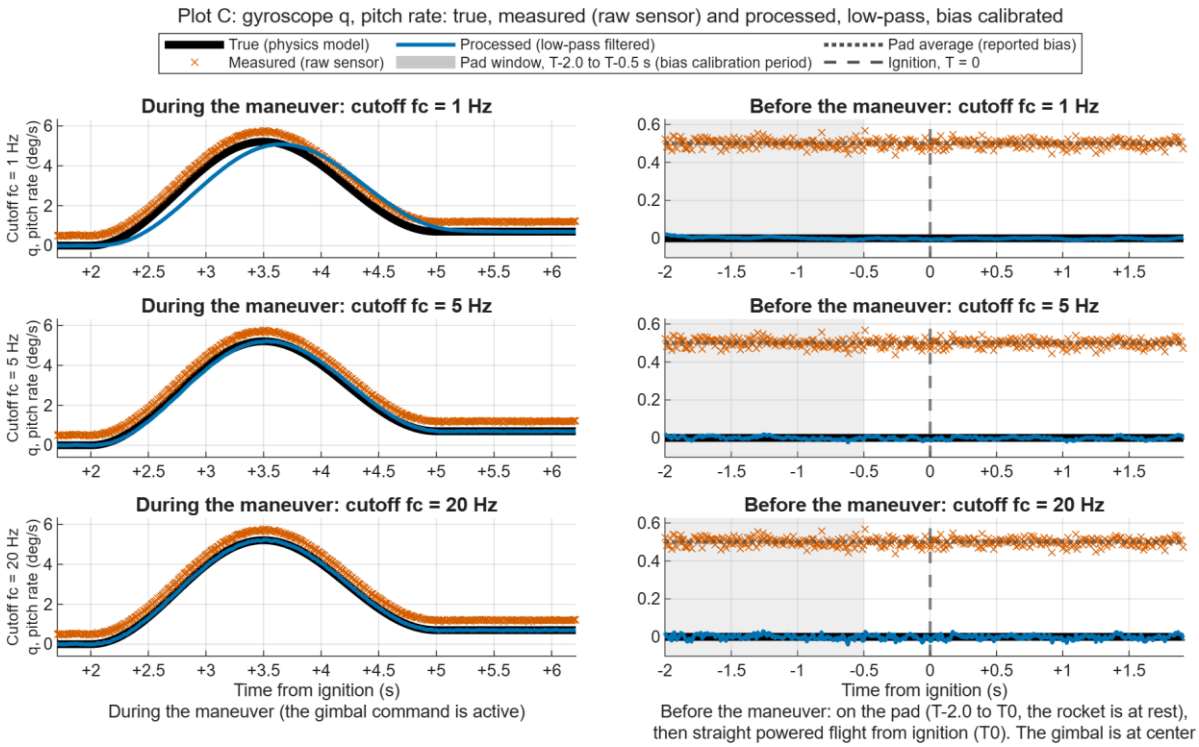


Figure 3. True, measured, and low-pass filtered pitch rate for three cutoff frequencies (Plot C). Each row is a separate cutoff on the same measurement. The left column is the maneuver and the right column is the pad and straight flight before it.

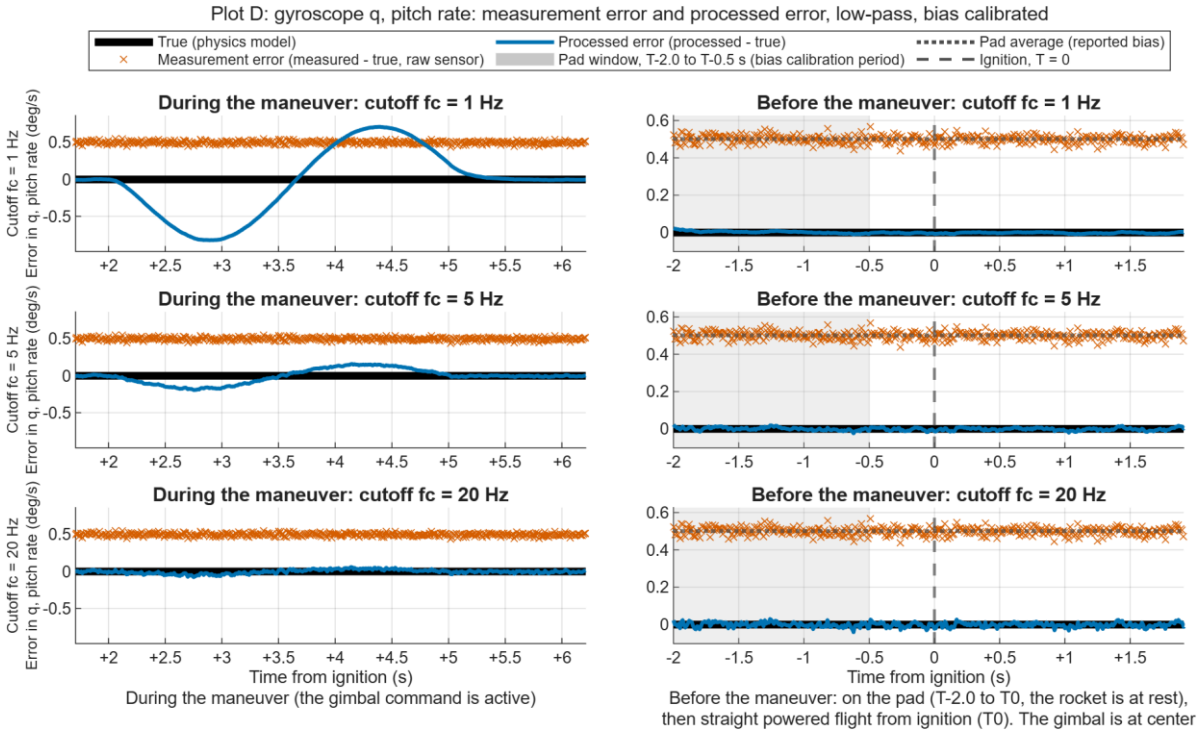


Figure 4. Measurement error and processed error, each minus the true signal, for the same three cutoffs (Plot D). The true signal is the zero line.

2.3 Kalman Filter Overview

A Kalman filter combines a model of the vehicle with the sensor readings (Schultz, 2004). It keeps an estimate $\hat{x}$ of the state, the parameters that describe the motion of the vehicle, and a table P that dictates

how uncertain the estimate is. Each sample the predict step uses the model to move the estimate forward by one sample, and P grows as the physics model is not perfect. The update step compares the new reading with the reading the prediction expected, and moves the estimate toward the reading by a fraction of the difference. This fraction is captured by the gain W, which is large when the filter is unsure of its prediction or favors the sensor, and small when the filter is sure of its prediction or the sensor is particularly noisy. After the update P shrinks. The equations used are those of Lecture 03d (Fetzer, 2026c, slide 15), which the lecture takes from Bar-Shalom et al. (2001, as cited in Fetzer, 2026c). They assume a linear model and noise that is zero mean, white, and independent. The derivation of the Kalman filter is attached in Appendix B and Section 2.6. The filter is given the raw reading with the pad bias removed and not the low-pass output, because its noise model describes the white noise of the sensor and a low-pass filter would correlate the noise and add a delay that the model does not know about.

2.4 Pitch Plane Model

The gyroscope measures only the angular rates, while the full vehicle parameters include angles, velocities, and positions. The filter therefore uses the subset that the sensor can see. For simplicity, this is modeled solely as the pitch plane. Pitch and yaw are assumed to be the same, per the assumption of a symmetric vehicle, so the yaw plane can be derived the same way and would require a copy of the current filter. The state is the pitch angle θ and the pitch rate q, the input is the pitch torque M_y that the gimbal makes, and the measurement is the gyro pitch rate (see Figure 5). The pitch angle is the Euler angle of the nose from straight up (Weisstein, n.d.), measured from the launch reference (Stevens et al., 2016).

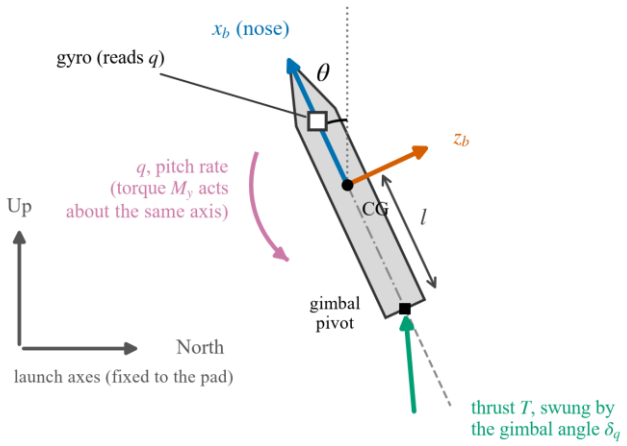


Figure 5. Pitch plane model. The pitch angle θ is measured from the vertical. The pitch rate q and the torque M_y act about the body y axis, the thrust is swung away from the nose axis by the gimbal angle δ_q, and the gyro reads q. This figure was generated with Claude AI and placed in Microsoft Word.

The rotational equation for pitch and the Euler angle kinematics from the equations of motion (Stevens et al., 2016) are, with the torque of the gimbal,

$$\dot{q} = \frac{M_y - (I_x - I_z)\,p\,r}{I_y}, \dot{\theta} = q\cos\varphi - r\sin\varphi$$
$$M_y = l\,T\sin\delta_q \tag{7}$$

Here I_x, I_y, and I_z are the moments of inertia about the body axes, T is the thrust, l is the lever arm, and δ_q is the pitch gimbal angle. These equations are nonlinear. Three assumptions about the roll motion linearize them for the pitch plane. The pitch angle itself does not need to be small, since it does not appear on the right side of these equations, and it reaches 18.8 degrees in the demo case. The roll angle φ stays below 2 degrees in this flight (1.6 degrees at most), so $\cos\varphi = 1$. The yaw rate r is at most 2.6 deg/s, so $r \sin\varphi$ is at most 0.07 deg/s and is dropped. There is no roll torque, so $p = 0$ and the $p\,r$ term is zero. The aerodynamic moment is switched off pending scope expansion using ANSYS CFD. The pitch plane model is then $\dot{\theta} = q$ and $\dot{q} = M_y/I_y$, which in matrix form is

$$\dot{x} = A\,x + B\,u, \quad x = \begin{bmatrix}\theta \\ q\end{bmatrix}, \quad A = \begin{bmatrix}0 & 1 \\ 0 & 0\end{bmatrix}, \quad B = \begin{bmatrix}0 \\ 1/I_y\end{bmatrix}, \quad u = M_y \tag{8}$$

2.5 Discretization and Noise

The filter runs once every $T_s = 0.01$ s, with the torque assumed constant over each step, which holds for a torque that changes little over such a short time. The state transition matrix F is the matrix exponential of $A\,T_s$, which stops after two terms as $A\,A = 0$. The input matrix G is the integral of $e^{A\sigma}$ over one step, times B. For this A the integral is $I\,T_s + A\,T_s^2/2$. The hand derivation of these matrices is in Appendix B.

$$F = e^{AT_s} = I + A\,T_s = \begin{bmatrix}1 & T_s \\ 0 & 1\end{bmatrix}, \qquad G = \left(I\,T_s + A\,\frac{T_s^2}{2}\right)B = \begin{bmatrix}T_s^2/(2I_y) \\ T_s/I_y\end{bmatrix} \tag{9}$$

With $T_s = 0.01$ s and $I_y = 0.487\ \text{kg}\cdot\text{m}^2$ this gives $G = [1.027 \times 10^{-4},\ 2.054 \times 10^{-2}]^T$, in rad per N·m and rad/s per N·m. The discrete model that the filter uses also has the process noise v and the measurement noise w.

$$x(k+1) = F\,x(k) + G\,u(k) + \Gamma\,v(k), \qquad y(k) = H\,x(k) + w(k) \tag{10}$$

The process noise v is an unknown angular acceleration held over one step. It stands for everything the model does not describe. Over one step it changes the angle by $T_s^2/2$ times its size and the rate by T_s times its size, so $\Gamma = [T_s^2/2,\ T_s]^T = [5 \times 10^{-5},\ 0.01]^T$. The gyro reads the rate and not the angle, so $H = [0\ \ 1]$. Written out, $y = H\,x = [0\ \ 1]\,[\theta\ \ q]^T = q$, so the 0 removes the angle from the reading and the 1 keeps the rate. The torque does not appear in the reading, so $J = 0$. The noises v and w have variances Q and R. The measurement noise variance is the sensor variance, $R = \sigma^2 = 1.19 \times 10^{-7}\ (\text{rad/s})^2$.

The process noise Q is built from the inherent slop of the system. The model assumes that the gimbal angle is exactly the angle that was commanded. The real linkage has measurable play. The play was measured as approximately 0.25 mm of total rattle at an effective radius of 51.7 mm, which is 0.277 degrees of gimbal angle (Kraus, 2026c). The play comes mainly from the ball end joints of the push arms that steer the motor (see Figure 6), due to the manufacture of the joint, in which the ball end rattles inside at both ends of the push arm. Assuming the real angle can be anywhere inside the play with equal chance, the gimbal angle error has a standard deviation of the play divided by $\sqrt{12}$, which is 0.080 degrees. At the peak thrust, one degree of gimbal angle error becomes 59.9 deg/s² of angular acceleration error through the lever arm l and the pitch inertia. So $\sigma_\alpha = 4.79\ \text{deg/s}^2 = 0.0837\ \text{rad/s}^2$ and $Q = \sigma_\alpha^2 = 7.00 \times 10^{-3}\ (\text{rad/s}^2)^2$. The hand calculation is in Section 3.2.



(a) (b)

Figure 6. Where the play of the control system is. (a) The vehicle built, with its two push arms on the motor case. (b) A ball end joint of a push arm at the thrust vector ring, one of the joints where the play is.

The simulated gimbal moves exactly as commanded, so this play is not in the simulation and the allowance in Q is not used up in the runs in this report. It has been demonstrated that this makes the filter more cautious, so the estimate follows the sensor closely. However, it is kept as it best describes the measured hardware, and it makes the error bars of the filter describe the real vehicle and not the ideal simulation. Section 4.4 varies Q as a key parameter. To check observability, a test of whether the readings can reveal every state was performed. The rows of H and $H\,F$ are the same, so the observability matrix has a rank of 1, thus not full rank, meaning that the rate gyro cannot correct the pitch angle.

$$\begin{bmatrix} H \\ HF \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \quad \Rightarrow \quad \text{rank} = 1 < 2 \tag{11}$$

The rate converges within a sample or two and the angle keeps whatever error it started with. A second sensor that reads the angle directly would make the angle observable. The Pixhawk V6X has a magnetometer (PX4, n.d.), which reads the direction of the Earth's magnetic field in the body axes, and that direction changes with the angle of the vehicle. Adding it is a possible extension and is outside the scope of this report.

2.6 Equations Implemented

The predict and update steps of Lecture 03d (Fetzer, 2026c, slide 15) for this model are

$$\begin{aligned} \hat{x}^{-} &= F\,\hat{x} + G\,u \\ P^{-} &= F\,P\,F^{T} + \Gamma\,Q\,\Gamma^{T} \end{aligned} \tag{12}$$

$$\begin{aligned} \tilde{y} = y - H\,\hat{x}^{-}, \qquad S = H\,P^{-}H^{T} + R, \qquad W = P^{-}H^{T}S^{-1} \\ \hat{x}^{+} = \hat{x}^{-} + W\,\tilde{y} \\ P^{+} = (I - WH)\,P^{-}(I - WH)^{T} + W\,R\,W^{T} \end{aligned} \tag{13}$$

Conventional notation of a minus sign marks the prediction before the reading is used, and a plus sign marks the result after the innovation is applied. The innovation $\tilde{y}$ is the reading minus the reading the

prediction expected, S is how big that difference is expected to be, and again, W is the gain. Because $H = [0\ 1]$, $S = P_{22}^{-} + R$ and the gain on the rate is $W_2 = P_{22}^{-}/(P_{22}^{-} + R)$. When the prediction is much more uncertain than the sensor the gain is near 1 and the filter believes the reading. A larger Q makes P_{22}^{-} larger, so the filter leans on the sensor.

The filter starts from $\hat{x}(0|0) = [3^{\circ},\ 5^{\circ}/\mathrm{s}]^{T}$, while the true state on the pad is zero, so the start-up of the filter shows in the plots. The starting uncertainty is $P(0|0) = \mathrm{diag}\left((5^{\circ})^{2},\ (10^{\circ}/\mathrm{s})^{2}\right)$. These are a choice, made so that the starting errors sit inside one standard deviation of the starting guess, as Lecture 03d recommends for a guess that is not well known (Fetzer, 2026c, slide 16).

2.7 Simulink Implementation

The sensor and the Kalman filter are blocks in the same Simulink model as the vehicle dynamics (see Figure 7). The Kalman Filter subsystem reads the measured angular velocity and the torque from the Force and Torque Generator, which is the known input u. Inside it (see Figure 8), two gain blocks, the triangles, pick the pitch rate and the pitch torque out of their three element signals, the Pad Bias Calibration block removes the bias, and the Kalman filter block holds the equations. A MATLAB Function block has no memory of its own. Thus, three Unit Delay blocks, the boxes marked 1/z, provide the estimate, the uncertainty, and the torque of the previous sample. The key lines of the Kalman filter block are

```
x_pred = kf_F*x_prev + kf_G*u_prev;
P_pred = kf_F*P_prev*kf_F' + kf_Gamma*kf_Q*kf_Gamma';
innovation = y - kf_H*x_pred;
S = kf_H*P_pred*kf_H' + kf_R;
W = P_pred*kf_H'/S;
x_hat = x_pred + W*innovation;
I_WH = eye(2) - W*kf_H;
P = I_WH*P_pred*I_WH' + W*kf_R*W';
```

The low-pass filter and the moving average run in a script on the saved measurement after the simulation, as nothing in the open loop model requires their output.

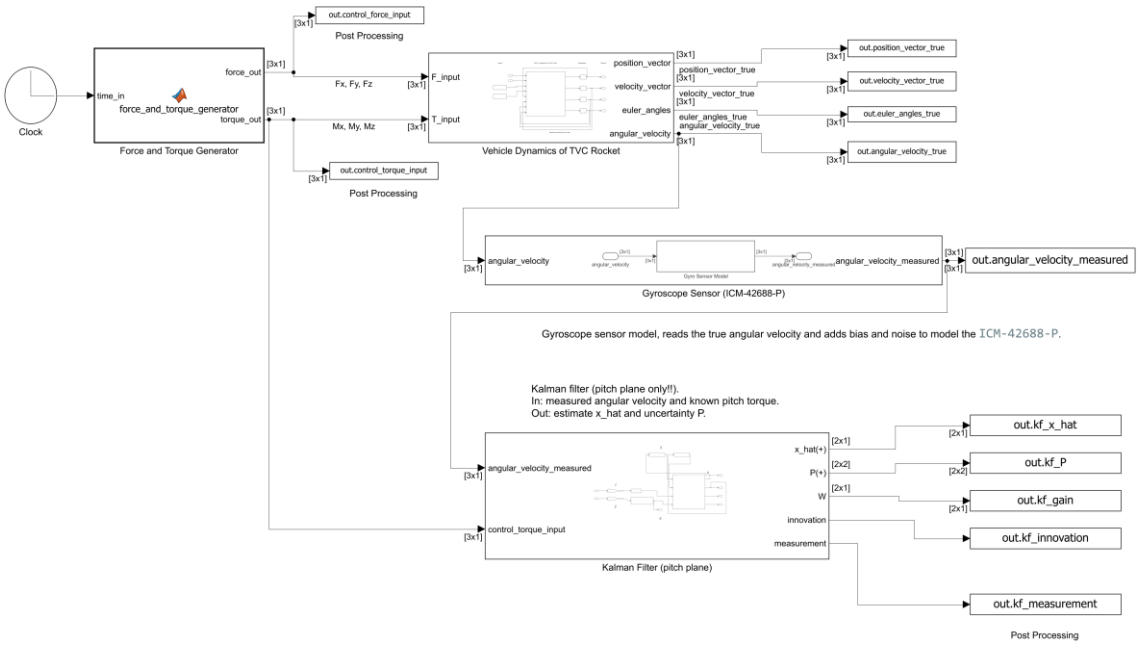
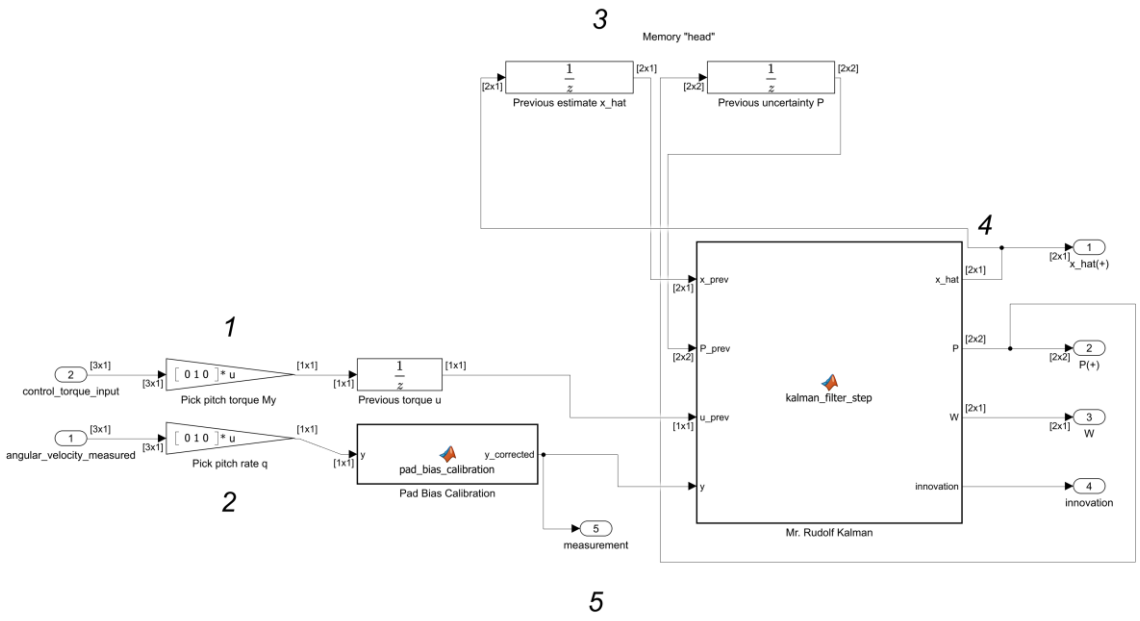


Figure 7. Top level of the Simulink model. The Gyroscope Sensor subsystem reads the true angular velocity. The Kalman Filter subsystem reads the measured angular velocity and the torque from the Force and Torque Generator. Their outputs are saved for plotting.



1) down-select the pitch rate q out of [p q r] vector, then remove the bias found on the pad

2) down-select the known pitch torque My out of [Mx My Mz] and hold it one sample, the torque that acted during the step just finished.

3) memory stores the estimate x_hat and the uncertainty table P of the previous step. starts at kf_x0 and kf_P0 (per init file, section 3j).

4) Kalman_Filter_Step predicts and updates, per the equations of Dr Fetzer Lecture 03d

5) measurement is the reading the filter was given (eg the pitch rate with the bias removed), saved for the plots.

Figure 8. Inside the Kalman Filter subsystem. The three Unit Delay blocks, the boxes marked 1/z, carry the estimate, the uncertainty, and the torque from one sample to the next.

2.8 Model Parameters

The parameters in Table 2 are defined at the top of the initialization file. The rows shaded gray are the vehicle parameters, which are unchanged. The white rows are new.

Table 2. Model parameters.

Quantity	Value	Source
Liftoff mass, m	1.7293 kg	Measured (weighed at about 1.7 kg) and OpenRocket
CG to gimbal pivot, l	0.46 m	Measured at pad weight
Roll inertia, I_x	0.002 kg·m²	OpenRocket (Kraus, 2026a)
Pitch and yaw inertia, $I_y = I_z$	0.487 kg·m²	OpenRocket (Kraus, 2026a)
Gimbal limit	±5°	Measured (4.97°) (Kraus, 2026b)
Thrust curve	7.94 s burn, 384 N·s, 63.4 N peak	NAR Standards and Testing (2009), ThrustCurve.org (n.d.)
Drag coefficient, C_d	0.35 to 0.30	OpenRocket (Kraus, 2026a)
Reference area, S, and diameter, D	70.88 cm², 9.5 cm	OpenRocket (Kraus, 2026a)
Gravity, g, and air density, ρ	9.80665 m/s², 1.225 kg/m³	Standard values
Sensor sample time, T_s	0.01 s (100 Hz)	Equal to the simulation step
Gyro noise density, ρ_g, and standard deviation, σ	0.0028 deg/s per $\sqrt{\text{Hz}}$, 0.0198 deg/s	Datasheet (TDK InvenSense, 2021), Equation (2)
Gyro bias, b	0.5 deg/s on each axis	Datasheet bound (TDK InvenSense, 2021)
Random number seed, run length	1, T−2 s to T+20 s (22 s)	Chosen
Pad time before ignition	2 s (liftoff 0.014 s after ignition)	Chosen, liftoff from the thrust curve
Bias calibration window, N_c	1.5 s (150 samples), enabled	Chosen
Low-pass cutoff frequencies, f_c	1, 5, 20 Hz	Chosen
Moving average windows, w	33, 7, 3 samples	Chosen to match the noise of the cutoffs
Gimbal play (total rattle at radius)	0.25 mm at 51.7 mm, 0.277°	Measured (Kraus, 2026c)
Gimbal angle error, σ_δ, and acceleration error, σ_α	0.080°, 4.79 deg/s² (0.0837 rad/s²)	Derived, Section 3.2
Process noise, Q	7.0×10⁻³ (rad/s²)²	Derived, Section 3.2
Measurement noise, R	1.19×10⁻⁷ (rad/s)²	Derived, σ^2
Matrices F, G, Γ, H, J	$F = [[1, 0.01], [0, 1]], G = [1.027 \times 10^{-4}, 2.054 \times 10^{-2}]^T$, $\Gamma = [5 \times 10^{-5}, 0.01]^T, H = [0, 1], J = 0$	Derived, Section 2.5
Starting estimate, $\hat{x}(0\|0)$	$[3^\circ, 5^\circ/\text{s}]^T$ (the true state is zero)	Chosen, wrong on purpose
Starting uncertainty, $P(0\|0)$	diag(25 deg², 100 (deg/s)²)	Chosen

The thrust curve in Table 2 comes from a motor test. The National Association of Rocketry certified the AeroTech I49N after three static firings, in which the motor is held on a test stand while its thrust is recorded against time, and published the averaged results as a total impulse of 383 N·s, a burn time of 7.68 s, a peak thrust of 63.7 N, and an average thrust of 49.4 N (NAR Standards and Testing, 2009). The 28 point file used here was made from that certification data and shared on ThrustCurve.org (Koelsch, 2009). Its total impulse of 384.1 N·s is 0.3 percent above the certified value, well inside the 1.8 percent spread between the three firings.

The shape of the curve comes from the motor itself. The thrust of a rocket is the mass flow out of the nozzle times the exit speed plus a pressure term, and the mass flow is set by the throat area and by the pressure and temperature in the chamber (NASA Glenn Research Center, n.d.). In a solid motor that pressure follows how fast the propellant burns, which depends on the pressure (NASA, 1972, p. 10), and how much propellant surface is burning, which is set by the shape of the propellant grain. The grain is shaped to deliver the thrust over time that the motor is designed for (NASA, 1972, p. 5). For the I49N the thrust rises to about 49 N in the first 0.03 s, peaks at 63.4 N near 1.3 s, falls slowly to about 42 N by 6 s, and drops to zero at 7.94 s.

Initial conditions of the vehicle remain unchanged. The rocket starts on the pad at position zero, at rest, pointing straight up with the Euler angles at zero and the angular rates at zero. However, a 2 second pad hold, with a support force acting as the ground reaction, was added to best model this system.

2.9 Modeling Assumptions

Table 3. Modeling assumptions and their justifications.

Assumption	Justification
The sensor error is a constant bias plus white Gaussian noise	The handout asks for noise characteristics. Scale factor error, cross axis sensitivity, quantization, temperature drift, and vibration are also error sources (Fetzer, 2026a) and are left out of this scope (Section 5).
The sensor samples at 100 Hz, equal to the simulation step, and the noise bandwidth is the Nyquist frequency	This keeps the true and measured signals on the same time points. The real chip runs from 12.5 Hz to 32 kHz (TDK InvenSense, 2021).
The bias is constant and is measured on the pad with the rocket still	A rocket at rest has a true rate of zero, so the pad average is the bias plus a little noise. A real vehicle would calibrate while armed and still, until the first movement.
The pad is a support force equal to the weight for 2 s	The model has no ground. The support lets the calibration happen before the motor lights and ends when the thrust exceeds the weight, so the flight after ignition does not change.
Small roll angle and no roll rate ($\cos\varphi = 1$, $r\sin\varphi = 0, p = 0$)	The roll angle stays below 2 degrees and the yaw rate below 2.6 deg/s. The pitch angle itself can be large. The dropped terms add up to 0.09 degrees of pitch angle over the run.
The torque given to the filter is the model torque, known exactly	In the simulation the torque is a signal of the model. A real vehicle would compute it from the commanded gimbal angle, the servo output, and the thrust, and Q covers the difference.
The process noise comes from the gimbal play only, uniform inside the play and independent from sample to sample	The play is the one measured hardware effect that the model leaves out. The peak thrust is used, so the allowance is generous late in the burn.
The process and measurement noises are zero mean, white, and independent. The filter is given the raw reading	These are the assumptions of the Kalman filter equations (Fetzer, 2026c, slide 15). The low-pass output would add a delay that the model does not know.
Only the pitch plane is estimated, and nothing uses the estimates	The yaw plane is the same problem with its own torque and rate. The simulation stays open loop.
The vehicle model is unchanged	Rigid body that is symmetric about the roll axis ($I_y = I_z$), flat non rotating Earth, constant gravity, mass, CG, and inertia, thrust from the published curve (Bigwood et al., 2021, p. 154), and no aerodynamic moment, which is still the largest assumption. A simple moment arm is accurate for the small gimbal angles used (Stepp, 2024).

3. How the Signals Flow

3.1 Process Flow from Truth to Estimate

Figure 9 shows the whole chain, and the steps below follow it from left to right.

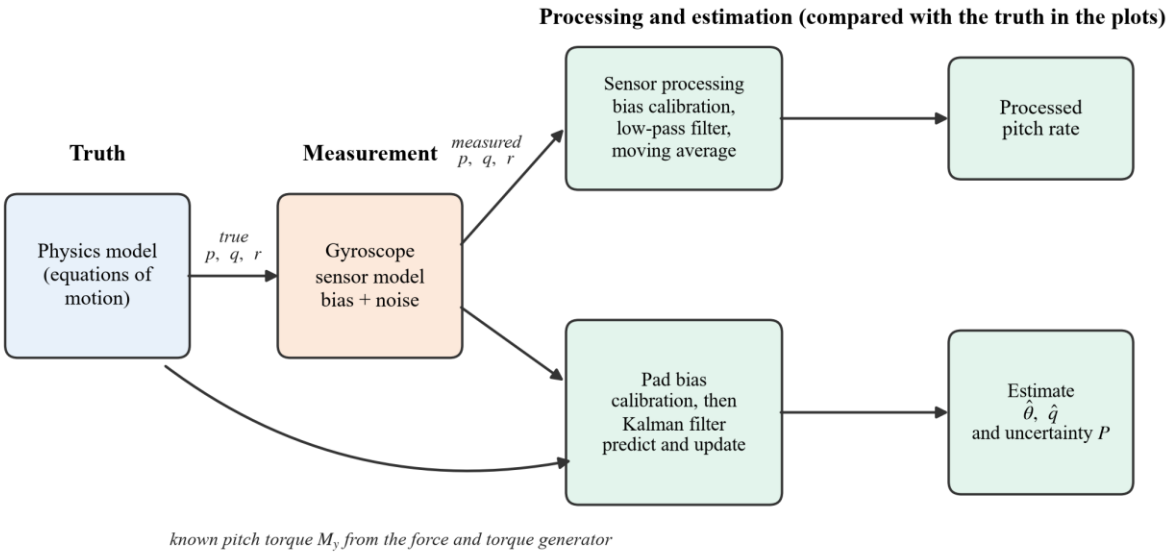


Figure 9. How the true rate becomes a measurement, a processed signal, and an estimate. Blue is the physics model, orange is the sensor, and green is the processing and estimation. This figure was generated with Claude AI and placed in Microsoft Word.

- Step 1. The vehicle model calculates the true rates p, q, r and the torque every 0.01 s (7). The gyroscope block adds the bias and a new noise sample to each rate (1).
- Step 2. On the pad, the average of the first 1.5 s of readings is taken as the bias and subtracted from every reading (3). The low-pass filter and the moving average then smooth the reading in a script, Equations (4) and (5), which are not implemented for the Kalman filter.
- Step 3. In the Kalman filter, the predict step moves the estimate one sample using the known torque (12), and the update step corrects it with the reading (13). The torque is the one that acted over the sample that just finished, so it is held back by one sample with a Unit Delay.
- Step 4. The estimate, its uncertainty, the gain, and the reading are saved, and the plots compare them with the true signal.

3.2 Hand Calc of the Process Noise

The 0.25 mm of play is the total rattle, with the arm pushed all the way to one side and then all the way to the other, at an effective radius of 51.7 mm. The play as an angle, and the standard deviation of an error that is uniform inside it, are

$$\theta_{play} = \frac{0.25}{51.7} = 4.84 \times 10^{-3}\ \text{rad} = 0.277^{\circ}, \qquad \sigma_{\delta} = \frac{\theta_{play}}{\sqrt{12}} = \frac{0.277^{\circ}}{3.464} = 0.080^{\circ}$$

The peak thrust is $T = 63.4$ N (ThrustCurve.org, n.d.). The angular acceleration error per unit of gimbal angle error, the acceleration error, and the process noise are

$$\frac{l\,T}{I_y} = \frac{(0.46)(63.4)}{0.487} = 59.9\ \text{deg/s}^2\ \text{per degree}, \qquad \sigma_{\alpha} = (59.9)(0.080) = 4.79\ \text{deg/s}^2 = 0.0837\ \text{rad/s}^2$$

$$Q = \sigma_{\alpha}^2 = (0.0837)^2 = 7.0 \times 10^{-3}\ (\text{rad/s}^2)^2$$

3.3 Hand Calc of One Filter Step

Take the starting estimate $\hat{x}(0|0) = [0.05236,\ 0.08727]^T$ (3° and 5°/s in radians) and $P(0|0) = \mathrm{diag}(7.615 \times 10^{-3},\ 3.046 \times 10^{-2})$. On the pad the torque is zero, but to show every term at work take a torque $u = 0.03\ \mathrm{N \cdot m}$ and a reading $y = 0.0005\ \mathrm{rad/s}$. The predict step gives

$$\hat{x}^- = \begin{bmatrix} 0.05236 + 0.01\,(0.08727) + (1.027 \times 10^{-4})(0.03) \\ 0.08727 + (0.02054)(0.03) \end{bmatrix} = \begin{bmatrix} 0.05324 \\ 0.08788 \end{bmatrix}$$

$$P^- = F\,P\,F^T + \Gamma\,Q\,\Gamma^T = \begin{bmatrix} 7.618 \times 10^{-3} & 3.046 \times 10^{-4} \\ 3.046 \times 10^{-4} & 3.046 \times 10^{-2} \end{bmatrix}$$

The update step uses $H = [0\ \ 1]$, so the innovation, S, and the gain are

$$\tilde{y} = 0.0005 - 0.08788 = -0.08738\ \mathrm{rad/s}, \qquad S = 3.0462 \times 10^{-2} + 1.19 \times 10^{-7} = 3.0463 \times 10^{-2}$$

$$W = \frac{1}{S}\begin{bmatrix} P_{12}^- \\ P_{22}^- \end{bmatrix} = \begin{bmatrix} 0.0099998 \\ 0.9999961 \end{bmatrix}$$

$$\hat{x}^+ = \begin{bmatrix} 0.05324 + (0.0099998)(-0.08738) \\ 0.08788 + (0.9999961)(-0.08738) \end{bmatrix} = \begin{bmatrix} 0.05236 \\ 0.00050 \end{bmatrix}$$

$$P^+ = (I - WH)\,P^-(I - WH)^T + W\,R\,W^T = \begin{bmatrix} 7.615 \times 10^{-3} & 1.19 \times 10^{-9} \\ 1.19 \times 10^{-9} & 1.194 \times 10^{-7} \end{bmatrix}$$

The rate estimate moved from 5 deg/s to the reading, and its standard deviation fell from 10 deg/s to 0.0198 deg/s, which is the sensor noise. The angle estimate moved from 3.050° to 3.000°. That is only the 0.05° that the wrong rate guess had added during the step, so the 3° error of the angle stays, as the observability result says.

4. Simulation Results

4.1 Test Matrix and What Was Entered

Every result in this report comes from the same model. The test box and the sensor and filter settings at the top of the initialization file were changed between runs, and Table 4 describes each case. All runs start 2 s before ignition and stop at T+20 s. As an example, these are the lines entered for the demo case in Section 4.2.

```
test_case = 2;   pitch_amp_deg = 0.1;   yaw_amp_deg = 0.05;   gimbal_start_s = 2;
gimbal_end_s = 5;
gimbal_period_s = 3;   sensor_Ts = 0.01;   gyro_noise_density_dps = 0.0028;   gyro_bias_dps
= 0.5;
pad_time_s = 2;   bias_cal_enable = 1;   lp_cutoffs_hz = [1 5 20];   ma_windows = [33 7 3];
kf_x0 = [deg2rad(3); deg2rad(5)];   kf_P0 = diag([deg2rad(5)^2, deg2rad(10)^2]);
```

Table 4. Lines entered in the initialization file for each case.

Case	Line changed from the demo case	Shown in
Demo case	none (the lines above)	Sections 2.1, 2.2, 4.2, and 4.3
Low-pass cutoffs and moving average windows	`lp_cutoffs_hz` and `ma_windows` (all three values in the demo run)	Section 2.2
Smaller process noise	`kf_Q = 1e-5;`	Section 4.4
10x the sensor noise	`gyro_noise_density_dps = 0.028;`	Section 4.4
Larger starting error	`kf_x0 = [deg2rad(20); deg2rad(-15)];`	Section 4.4
Slower start	`kf_Q = 1e-5; kf_x0 = [deg2rad(3); deg2rad(0.1)]; kf_P0 = diag([deg2rad(5)^2, deg2rad(0.01)^2]);`	Section 4.4

4.2 Demo Case

The demo case applies a sine of 0.1° in pitch and 0.05° in yaw from 2 to 5 s after ignition. Figure 10 has four plots. The first three are inputs to the dynamics block and only the fourth is a result.

- The gimbal angles (top left) are the command and the only thing the rocket is told to do. The thrust force (top right) follows the thrust curve and falls to zero at burnout, and the sideways pieces F_y and F_z sit on top of zero at this scale. The torque about the CG (bottom left) has the same shape as the gimbal angles.
- The angular velocity (bottom right) is the response. The pitch rate peaks at 5.21 deg/s at T+3.5 s and settles at 0.70 deg/s. The yaw rate peaks at −2.61 deg/s and settles at −0.35 deg/s, and the roll rate stays at zero. The pitch angle is 18.8 degrees at T+20 s, and the rocket peaks at 978 m at T+16.4 s. The pitch rate q is the true signal that the gyroscope reads in the rest of this report.

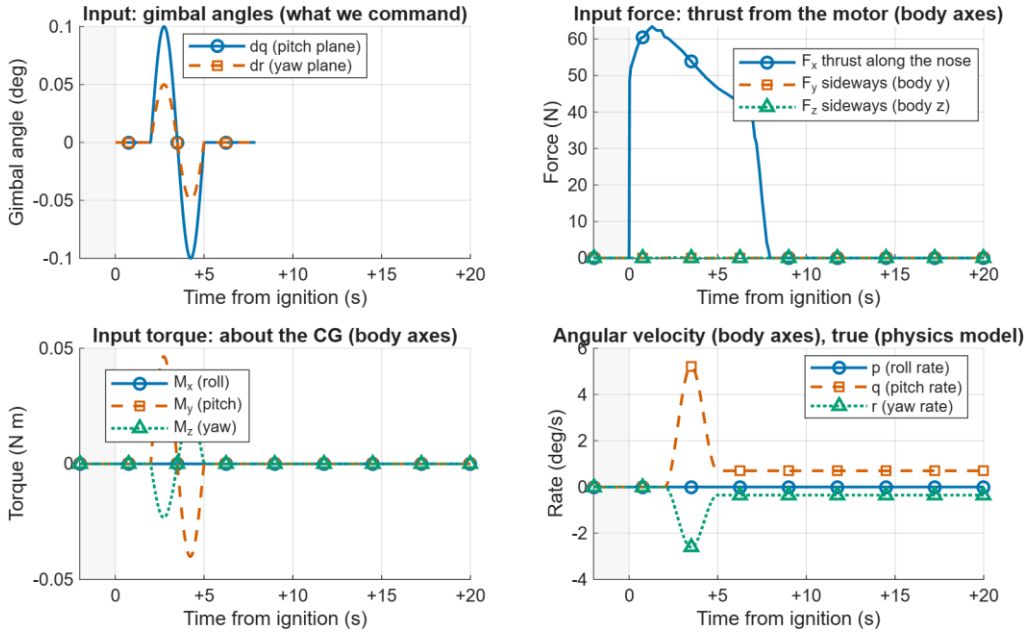


Figure 10. Inputs and angular velocity for the demo case. The first three plots are inputs and the fourth is the response of the rocket.

The commanded angles are small on purpose, because the vehicle is very sensitive to the gimbal angle, as Section 6 explains. The results of the sensor model and of the sensor processing are shown in Sections 2.1 and 2.2, next to their equations. The results of the Kalman filter follow in Sections 4.3 and 4.4.

4.3 Kalman Filter

Figures 11 to 13 demonstrate the difference between the sensor measurement and the Kalman filter state estimate on the demo case. Figure 11 compares the measured and estimated values, Figure 12 compares the measurement error with the estimation error, and Figure 13 shows how the covariance matrix varies over time. The filter starts on the pad at T−2 s from the wrong guess of 3° and 5 deg/s, drawn as the purple square one sample before the first reading. The true state is zero. The starting error of the rate is removed at the first reading (right columns of Figures 11 and 12) and the covariance falls at the same step (Figure 13).

The gyroscope measures the pitch rate, which is the rate of change of the pitch angle. Measuring a rate shows how the angle changes but not where it started, so the rate is observed and the angle is not. Convergence therefore applies to the rate, while the angle keeps its starting error and its error bar does not shrink. A sensor that reads the angle directly, such as the magnetometer of the Pixhawk V6X (PX4, n.d.), would let the angle converge as well.

The Kalman filter differs from the low-pass filter of Section 2.2 in what it knows. A low-pass filter uses only past readings of one signal, so it lags a changing rate, by 31.8 ms at a cutoff of 5 Hz and 8.0 ms at 20 Hz (Table 1). The Kalman filter also has a model of the vehicle and the known torque, so it follows the maneuver without lag, and it can combine measurements from more than one source (Fetzer, 2026c, slide 2), where a low-pass filter smooths one signal at a time. Over the run the rms error of the pitch rate is 0.0174 deg/s for the Kalman filter with the Q from the measured play, against 0.0164 deg/s for the 20 Hz low-pass filter and 0.0431 deg/s for the 5 Hz one, and 0.0045 deg/s for the Kalman filter with the smaller Q of Section 4.4. With the Q from the measured play the Kalman filter is therefore about as accurate as the 20 Hz low-pass filter over the whole run, and during the maneuver alone it is more accurate, 0.0174 deg/s against 0.0298 deg/s.

- Figure 11 shows the measured and estimated values. The estimate of the pitch rate follows the true rate through the maneuver with no delay, because the model of the filter already knows the torque. At the start the rate estimate moves from the 5 deg/s guess to the reading in one sample, since the gain on the first sample is 0.999996. The gain is that large because the starting uncertainty of the rate, a bar of 10 deg/s, is much larger than the sensor noise of 0.02 deg/s, so the filter has no reason to hold on to its guess. The first corrected reading is exactly zero, because the pad average that is subtracted includes that reading, so the first update is cleaner than the later ones. A slower start, in which the convergence takes several samples, is shown in Section 4.4. The pitch angle has no measurement. Its estimate runs parallel to the true pitch angle, 3.0 degrees above it for the whole run.
- Figure 12 shows the errors. The rate estimation error has an rms of 0.0174 deg/s against 0.0199 deg/s for the measurement the filter was given, which is 12 percent less noise. 99.95 percent of the rate errors fall inside three error bars of the filter and 70.7 percent inside one, against 68 percent for normal noise. The angle error is 3.00 degrees at the start and 3.08 degrees at the end, well inside its bars of plus and minus 15 degrees. The extra 0.08 degrees comes from the terms dropped in the linearization of Section 2.4.
- Figure 13 shows the covariance matrix. The rate variance P_{22} falls from 100 deg²/s² to 3.92×10^{-4} deg²/s² at the first update, a standard deviation of 0.0198 deg/s, the sensor noise. It settles at 3.41×10^{-4} deg²/s² (0.0185 deg/s) after three samples, and the gain on the rate settles at 0.871. The angle variance P_{11} stays at 25 deg², a bar of 5 degrees, because the readings carry no information about the angle.

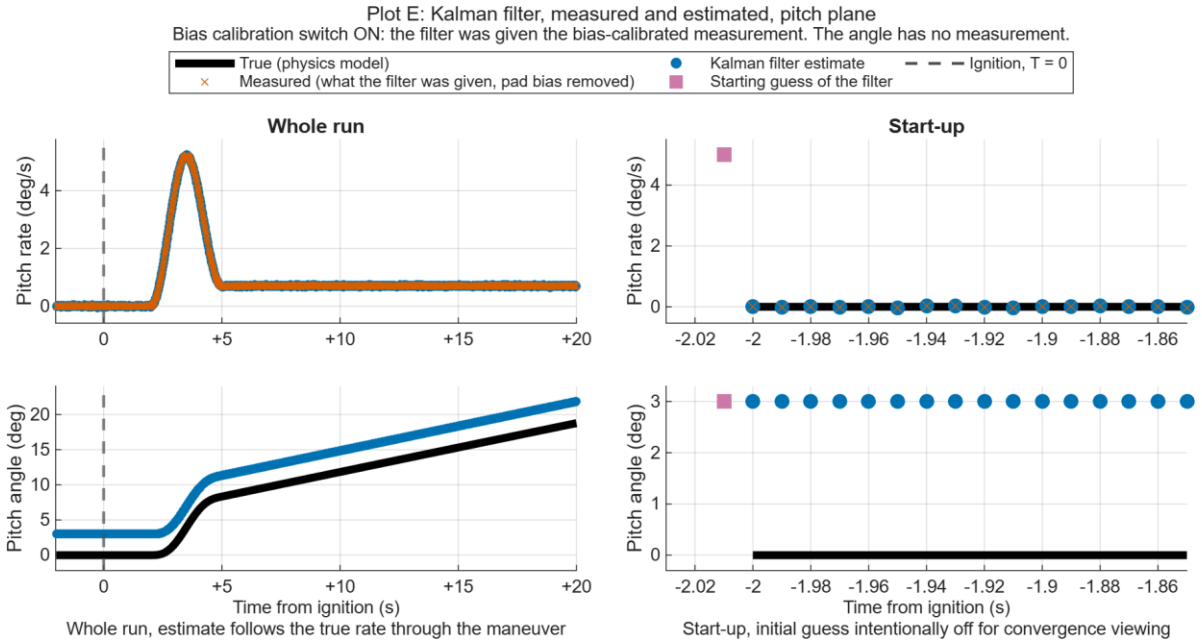


Figure 11. Measured and estimated pitch rate and pitch angle for the demo case (Plot E). The purple square is the starting guess of the filter, drawn one sample before the first reading. The angle has no measurement.

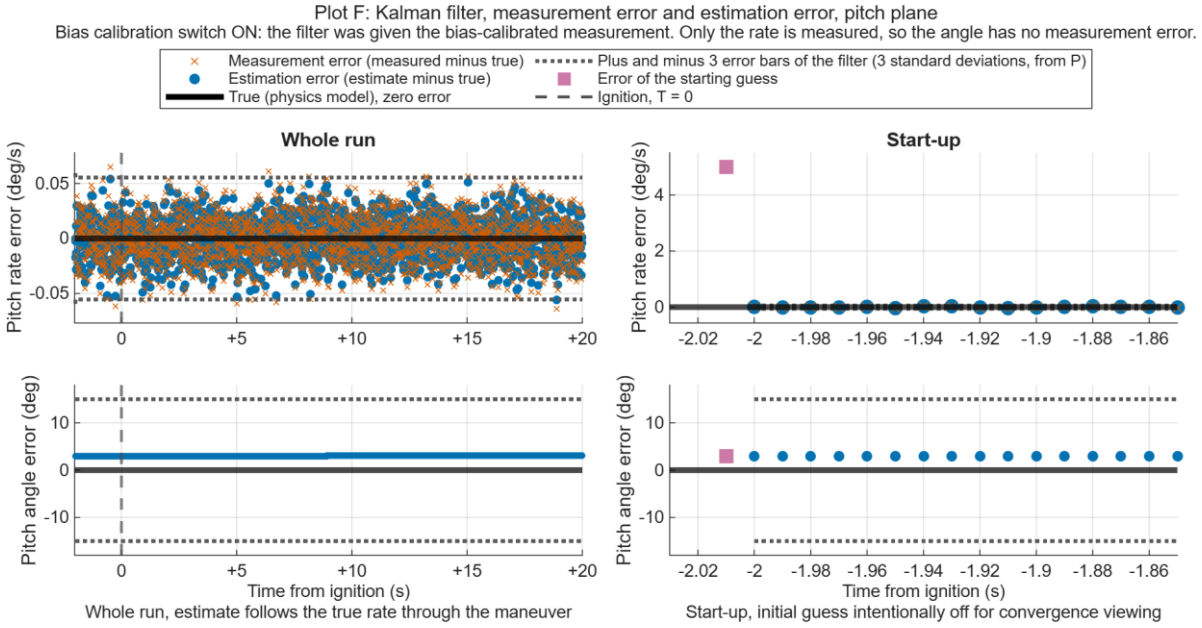


Figure 12. Measurement error and estimation error, each minus the true signal, for the demo case (Plot F). The dotted lines are plus and minus three error bars of the filter.

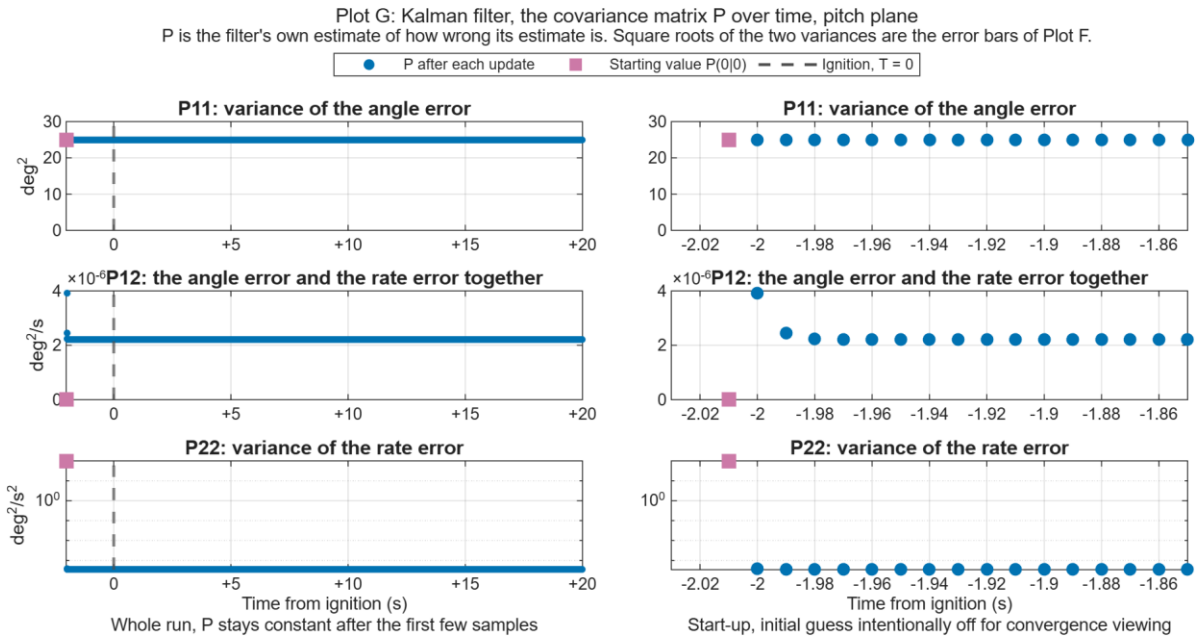


Figure 13. The covariance matrix of the filter over time for the demo case (Plot G). The matrix is symmetric, so only P_{11}, P_{12}, and P_{22} are shown.

The filter removes only 12 percent of the noise because of Q. The Q from the measured play says that the model may be wrong by 4.8 deg/s² on every sample, so the filter leans on the sensor, with a gain of 0.87. The simulated gimbal has no play, so the model is better than Q admits. The average of the innovation squared divided by its expected size S is 1 for a filter whose noise numbers match the simulation, and here it is 0.23. The error bar of 0.0185 deg/s is a little above the actual error of 0.0174 deg/s. This is the cost of an allowance for hardware that the simulation does not have, and it is what makes the filter match the real vehicle.

4.4 Varying Key Parameters

The filter design values were varied in Section 2.2. For the Kalman filter, four runs were made, and Table 5 gives the results. Each curve in Figures 14 and 15 is a separate run.

Table 5. Kalman filter results for the demo case and the four varied runs. The innovation measure is the average of the innovation squared divided by its expected size, which is 1 for a matched filter.

Case	Measurement error rms (deg/s)	Estimation error rms (deg/s)	Error bar (deg/s)	Inside 3 bars (%)	Innovation measure	Angle error at end (°)
Demo case, $Q = 7.0 \times 10^{-3}$	0.0199	0.0174	0.0185	99.95	0.23	3.08
Smaller process noise, $Q = 1 \times 10^{-5}$	0.0199	0.0045	0.0059	100.00	0.96	3.08
10x the sensor noise	0.1987	0.0677	0.0917	100.00	0.88	2.98
Larger starting error, 20° and −15°/s	0.0199	0.0174	0.0185	99.95	0.23	20.08
Slower start, $Q = 1 \times 10^{-5}$, rate guess off by 0.1 deg/s with a bar of 0.01 deg/s	0.0199	0.0048	0.0059	99.50	0.96	3.08

- Smaller Q (Figure 14a, green squares). A Q of 1×10^{-5} is about 700 times smaller than the Q from the measured play. The estimation error falls to 0.0045 deg/s rms, the error bar to 0.0059 deg/s, and the innovation measure becomes 0.96, so the filter matches this simulation. Nothing that was

measured supports that value and the simulation does not contain the gimbal play, so the measured value was kept in the demo case and in the runs that vary something else.

- 10x the sensor noise (Figure 14b). The measurement error rises to 0.199 deg/s rms and the estimation error to 0.068 deg/s, which is 66 percent less. The gain on the rate settles at 0.21 instead of 0.87. The filter helps most where the sensor is the weak link.
- Larger starting error. With the starting uncertainty of the demo case the rate is fixed in one sample whatever the guess, with an error of 0.00006 deg/s after the first update. The angle error stays at 20.0 degrees and ends at 20.1 degrees, which is four of its 5 degree error bars, so the bars are not honest for this guess. A starting uncertainty should be at least as large as the starting error if the error bars are to be honest.
- Slower start (Figure 15b). In the demo case the rate converges at the first reading because the filter is unsure of its starting guess, with a bar of 10 deg/s, and the sensor is quiet, so the gain is near 1 (Figure 15a). The convergence takes several samples when the filter is more sure of its starting guess than of the sensor, and trusts its model. In this run the starting rate error is 0.10 deg/s with a bar of 0.01 deg/s, and Q is 1×10^{-5}. The demo case keeps the Q from the measured play, and this run changes it only to show how the convergence depends on the starting values. With the Q from the measured play the gain on the rate cannot fall below 0.87, so the convergence would still be over within a sample or two. The rms rate error is 0.057 deg/s over the first five updates, 0.028 deg/s over the next five, 0.014 deg/s over the next ten, and 0.007 deg/s over the next twenty, and the error bar falls from 0.0090 to 0.0059 deg/s. A larger starting uncertainty gives faster convergence, and a smaller one makes the filter more stubborn.

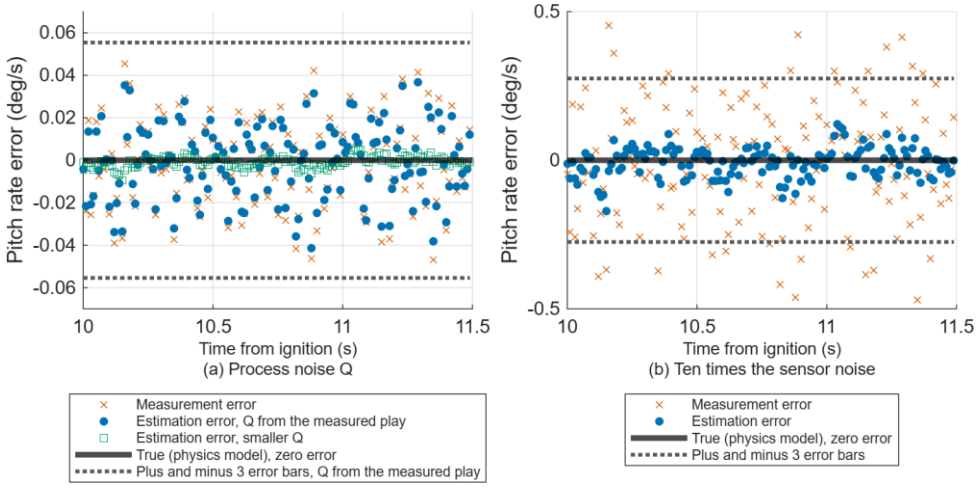


Figure 14. Pitch rate error from 10 to 11.5 s after ignition for the varied runs. (a) The Q from the measured play and a smaller Q, with the three error bars of the first. (b) 10x the sensor noise. Each curve is a separate run.

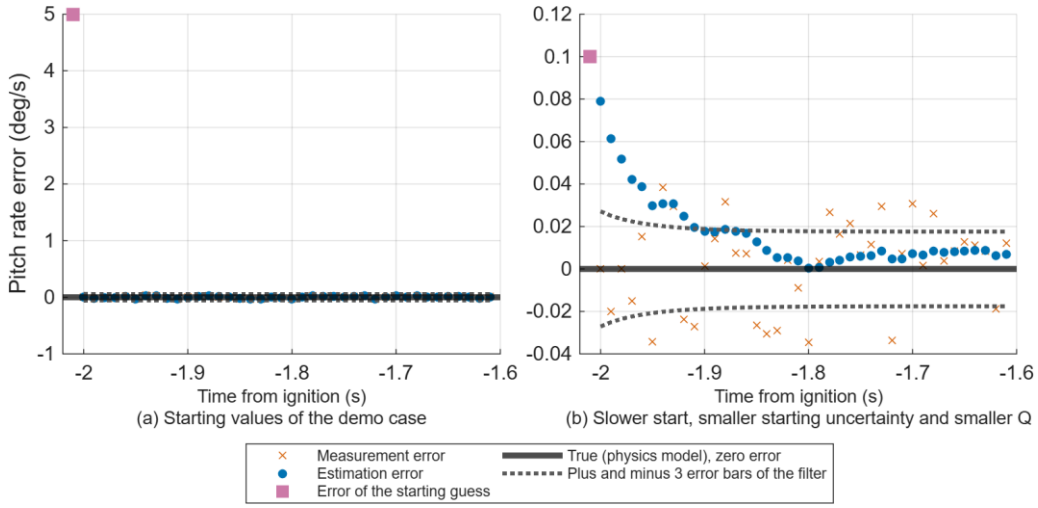


Figure 15. Pitch rate error of the first 40 updates, on the pad. (a) The starting values of the demo case, where the error of the starting guess is removed at the first reading. (b) A slower start, with a smaller starting uncertainty and a smaller Q, where the error decays over several samples. The purple square is the error of the starting guess, drawn one sample before the first reading.

In the demo case the rate estimate forgets its starting guess in one sample, but the angle estimate never does. Correcting the angle would need a second sensor whose reading depends on the angle, as noted in Section 2.5.

5. Limitations and Uncertainties

The sensor model has a constant bias and white noise only. A real gyroscope also has scale factor error, cross axis sensitivity, quantization, and a bias that drifts with temperature, at 0.005 deg/s per degree Celsius on the datasheet (TDK InvenSense, 2021). It should be noted that the Pixhawk V6X has onboard heating resistors that hold its IMUs at a controlled temperature (PX4, n.d.), which reduces the temperature drift. Motor vibration is not simulated, and at a 100 Hz sample rate vibration at hundreds of hertz would fold into the 0 to 50 Hz band unless the sensor filters it first. The 10x noise run in Section 4.4 was an attempt to model this excess motor vibration noise. The Kalman filter is given the torque of the model, which is exact. On the real vehicle the torque would be computed from the commanded gimbal angle, the servo output, and a thrust estimate, with the servo delay and the play of the linkage, and Q stands in for those differences. The play is measured but is not in the simulated gimbal, so the results show the cost of the allowance, though not its benefit.

It should be noted to the reader that the aerodynamic moment is switched off and is still the largest assumption of the vehicle model. The filter estimates the pitch plane only, and a real vehicle would start with a tighter starting uncertainty, as the pad attitude is known to a degree or two. The larger starting values are used here to make the starting error visible in the plots.

6. Observations and Next Steps

The results show that the vehicle has a very oversensitive CG in its current configuration. The pivot is 0.46 m behind the CG and the pitch inertia is only 0.487 kg·m², so a gimbal command of 0.1 degrees already gives a peak pitch rate of 5.2 deg/s and a pitch angle of 18.8 degrees at the end of the run. The measured play of 0.277 degrees is larger than the commands of the demo case, which means that the linkage can move more than the controller asks. A separate study with the same model moved the CG toward the pivot by adding ballast at the tail. About 1 kg of ballast made a 0.30 degree command give a peak pitch rate of 4.8 deg/s, similar to the demo case, but it lowered the thrust to weight ratio, peak thrust over weight, from 3.7 to 2.4 and cut the apogee from 978 m to 451 m.

These observations come from the physics model, and they say that the vehicle needs mechanical updates for stability, either more weight, a CG closer to the pivot, or a combination of both. The plan is to vary the mass and the CG location separately from this report, in the same model, to find where a healthy regime lies. The success criterion for the flight is stable vertical flight, meaning that the vehicle is pitched in a stable way within the range of motion that the gimbal allows, 5 degrees each way. Apogee is not a requirement, so trading apogee for stability, for example by adding weight to increase the inertia of the rocket, is acceptable.

Works Cited

Bigwood, J., Connor, R., Gilbert, C., Kuhnle, C., Mitkov, A., Rutkowski, N., Schrader, C. M., & Ternent, J. (2021). *Design, analysis, and test of a high-powered model rocket-2* (Major Qualifying Project No. JB3-2102). Worcester Polytechnic Institute.

Bosch Sensortec. (2018). *BMI088 data sheet* (Document revision 1.3, BST-BMI088-DS001-13).

Fetzer, K. L. (2026a). *AESE 404 lecture 02b. Navigation, sensors* [Lecture slides]. Lehigh University.

Fetzer, K. L. (2026b). *AESE 404 lecture 03c. Navigation, sensor processing* [Lecture slides]. Lehigh University.

Fetzer, K. L. (2026c). *AESE 404 lecture 03d. Navigation, state estimation* [Lecture slides]. Lehigh University.

InvenSense. (2016). *ICM-20649 datasheet* (DS-000192, Revision 1.0).

Koelsch, M. (2009). *AeroTech I49N* [RockSim data file]. ThrustCurve.org. https://www.thrustcurve.org/simfiles/5f4294d20002e90000000662/

Kraus, J. (2026a). *OpenRocket model and simulation of the Kalman TVC rocket* [Unpublished simulation data].

Kraus, J. (2026b). *Measured parameters of the Kalman TVC rocket* [Unpublished raw data].

Kraus, J. (2026c). *Measured play of the gimbal linkage of the Kalman TVC rocket* [Unpublished raw data].

MathWorks. (n.d.). *What is the Nyquist theorem?* MATLAB & Simulink. https://www.mathworks.com/discovery/nyquist-theorem.html

NAR Standards and Testing. (2009). *AeroTech I49N certification data* [Motor certification sheet]. National Association of Rocketry. https://www.thrustcurve.org/motors/cert/62db4bb504f4d60004889087/I49N.pdf

NASA. (1972). *Solid propellant grain design and internal ballistics* (NASA SP-8076). https://ntrs.nasa.gov/citations/19730007077

NASA Glenn Research Center. (n.d.). *Thrust equations summary*. In *Beginners guide to aeronautics*. https://www1.grc.nasa.gov/beginners-guide-to-aeronautics/thrust-equations-summary/

PX4. (n.d.). *CUAV Pixhawk V6X*. In *PX4 guide (main)*. https://docs.px4.io/main/en/flight_controller/cuav_pixhawk_v6x

Schultz, D. W. (2004). *Application of the Kalman filter to rocket apogee detection* (NAR 63255).

Stepp, N. A. (2024). *MSFC-TVC TB-03, Derivation of thrust vector control (TVC) actuator-force / gimbal-torque transformation matrix* (NASA/TM-20240012525). NASA Marshall Space Flight Center. https://ntrs.nasa.gov/citations/20240012525

Stevens, B. L., Lewis, F. L., & Johnson, E. N. (2016). *Aircraft control and simulation. Dynamics, controls design, and autonomous systems* (3rd ed.). Wiley.

TDK InvenSense. (2021). *ICM-42688-P datasheet* (DS-000347, Revision 1.6).

ThrustCurve.org. (n.d.). *AeroTech I49N* [Motor data]. https://www.thrustcurve.org/motors/AeroTech/I49N

Weisstein, E. W. (n.d.). *Euler angles*. In *MathWorld, A Wolfram Resource*. https://mathworld.wolfram.com/EulerAngles.html

Appendix A. Figures and Tools

Figure 1 is a CAD picture of the vehicle and Figure 6 shows photographs of it, both made by the author. Figures 5 and 9 were drawn with Claude AI and placed in Microsoft Word. Figures 7 and 8 are screenshots of the Simulink model, Figures 3, 4, and 10 to 13 are the plot windows that the Simulink model and its plot file produce, and Figures 2, 14, and 15 are made from the same runs. The equations were typeset in Microsoft Word. Where Claude AI helped with the MATLAB files, it is noted in the code.

Appendix B. Kalman Filter Hand Derivations

The handwritten derivations of the Kalman filter matrices follow.