Linear Systems TTK4115 - Helicopter lab

Alex Danielsen-Haces – 764088 Sindre Hansen – 732719
Daniel Nakken – 740939

October 2016

NTNU – Trondheim
Norwegian University of
Science and Technology

Contents

1 Part 1 - Mathematical modeling

1.1 Problem 1

The helicopter model shown in fig. 1 and fig. 2 is used to derive the equations of motion.

Figure 1: Helicopter model from the assignment depicting forces and joint axes [2, p.12].

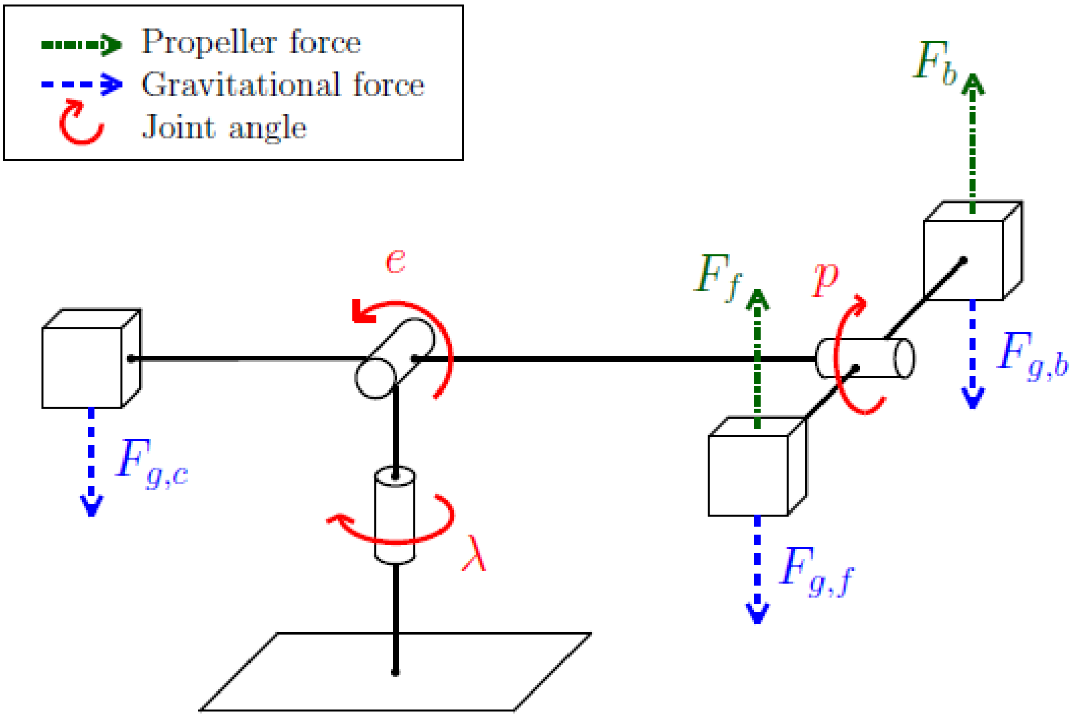
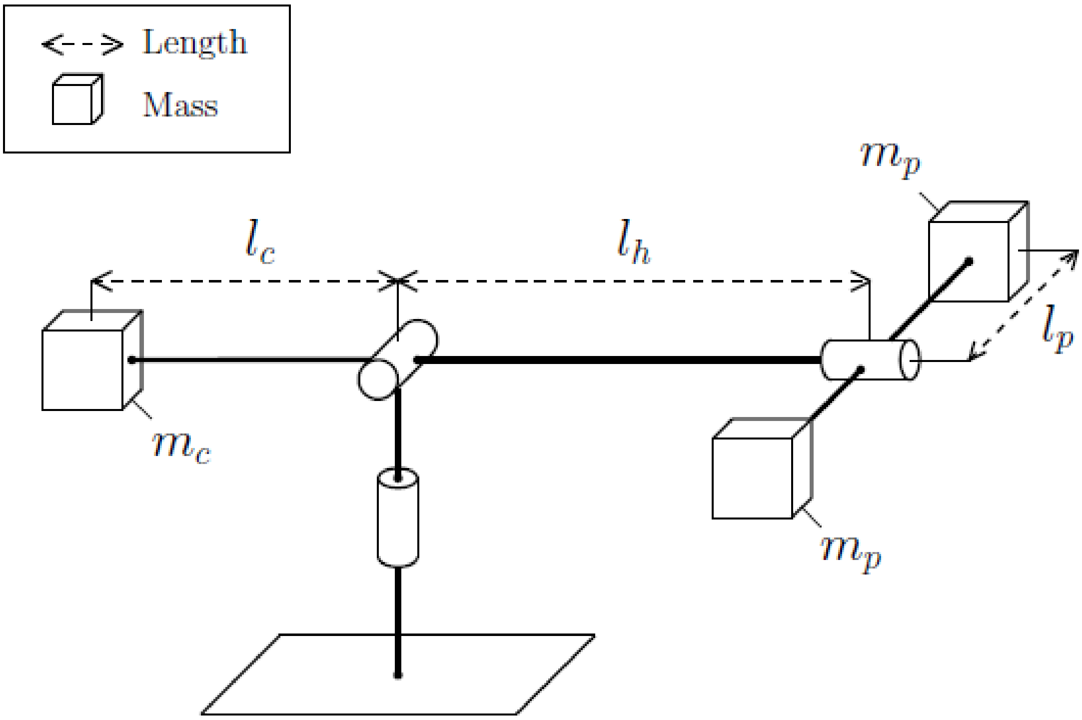


Figure 2: the helicopter model figure 8 from the assignment depicting masses and distances between the joint axes and the point masses [2, p.13].



Only forces perpendicular to a moment arm, with the moment arm again perpendicular to the axis of rotation in question produce a moment, or torque. The equation of motion for the pitch is found through the momentum around the pitch axis in the clockwise direction as shown in fig. 1. It becomes:

$$\begin{aligned} J_p\ddot{p} &= l_p(F_{g,b} - F_b - F_{g,f} + F_f) \\ &= l_p(m_pg - mp_g + K_fV_f - V_b) \\ &= l_pK_f(V_f - V_b) \end{aligned}$$

Where the lengths, including l_p, are shown in fig. 2. Since $V_d = V_f - V_b$, this can be written as:

$$J_p\ddot{p} = l_pK_fVd \tag{1}$$

Therefore $L_1 = l_pK_f$.

The equation of motion for the elevation angle is found similarly through the momentum in the counter-clockwise direction around the elevation axis, again as shown in fig. 1.

$$J_e\ddot{e} = arm_cF_{g,c} - arm_h(F_{g,f} + F_{g,b}) + l_h(F_{f,p} + F_{b,p})$$

where arm_c is the moment arm between the counterweight point mass and the elevation axis, and arm_h is the moment arm between any of the two motor point masses and the elevation axis. $F_{f,p} = F_{f,perpendicular}$ is the perpendicular component of F_f, and $F_{b,p} = F_{b,perpendicular}$ is the perpendicular component of F_b. As shown in fig. 3, $arm_c = l_c cos(e)$, and $arm_h = l_h cos(e)$, and the moment arm for the motor forces is indeed l_h.

Figure 3: moment arms around the elevation axis, with relevant forces. Also, the positive direction of momentum is displayed as counter-clockwise around the e-axis.

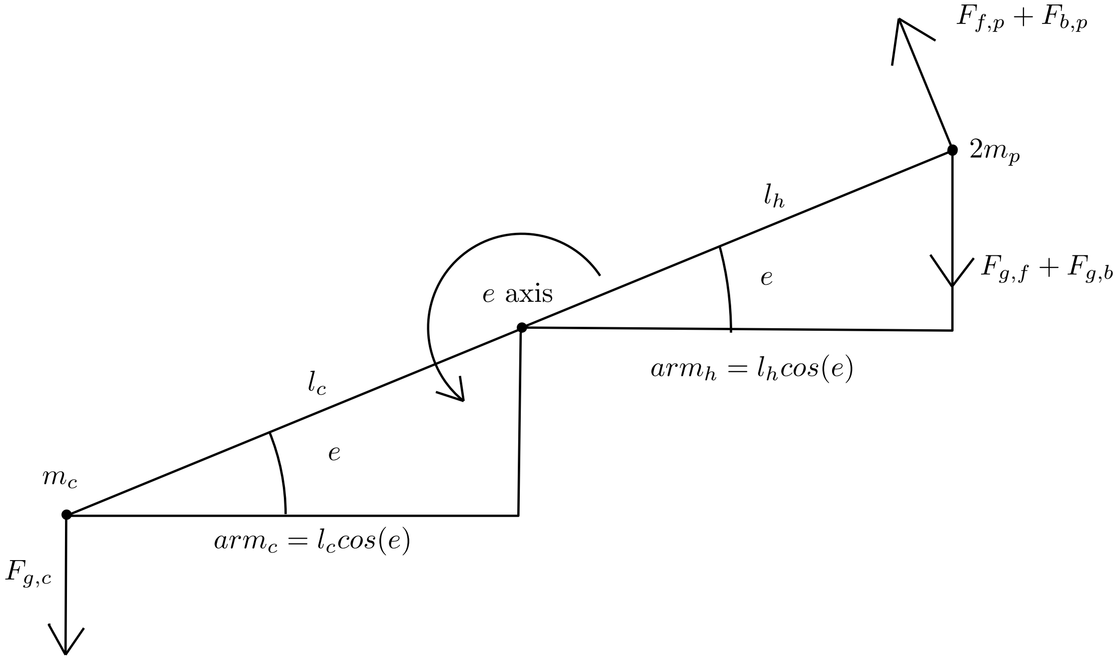
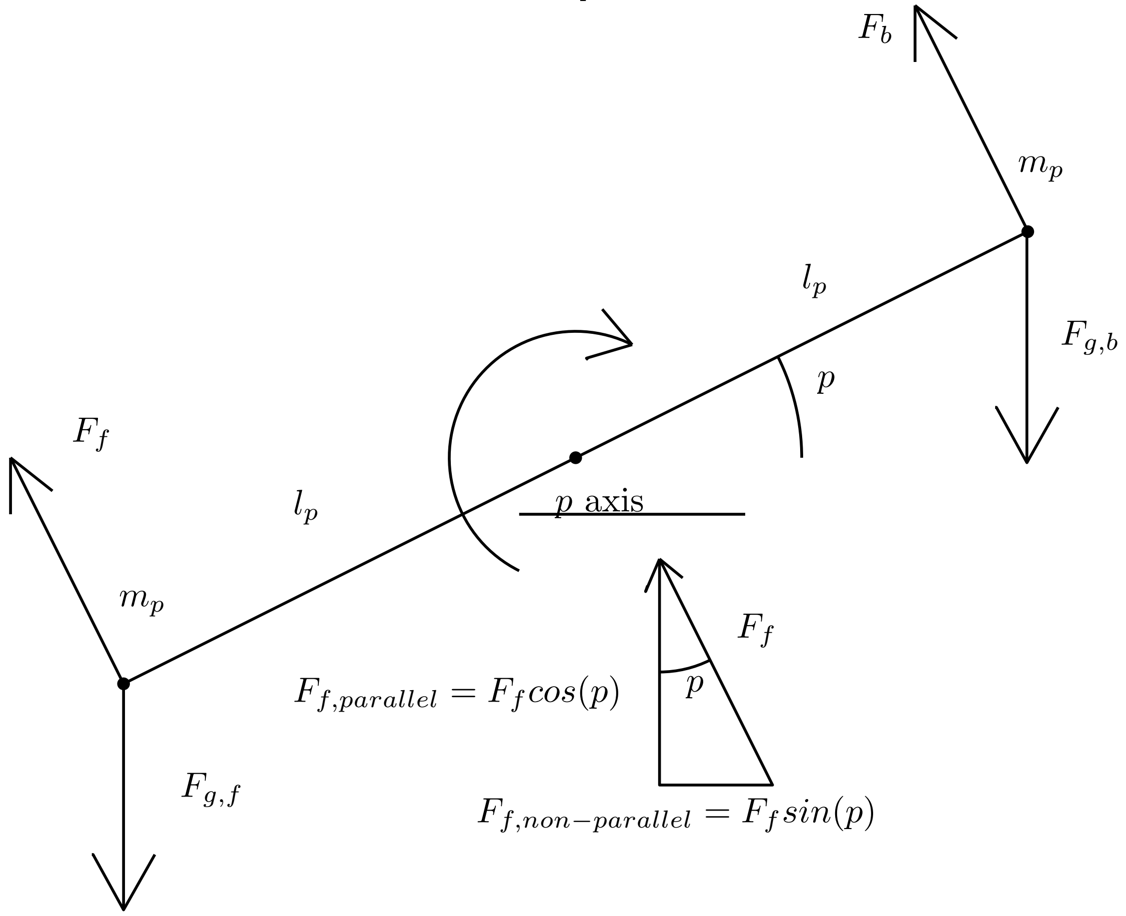


Figure 4: gravitational and motor forces on the helicopter head, as well as the decomposition of the motor forces into vertical and horizontal components.



As shown in fig. 4, the perpendicular components of the motor forces $F_{f,p}$ and $F_{b,p}$ are $F_f cos(p)$ and $F_b cos(p)$ respectively.

$$\begin{aligned} J_e\ddot{e} &= arm_c F_{g,c} - arm_h(F_{g,f} + F_{g,b}) + l_h cos(p)(F_f + F_b)) \\ &= arm_c m_c g - arm_h(m_p g + m_p g) + l_h K_f cos(p)(V_f + V_b)) \end{aligned}$$

Substitute in $V_s = V_f + V_b$ and the two moment arms and the equation becomes:

$$J_e\ddot{e} = l_c cos(e) m_c g - l_h cos(e) 2 m_p g + l_h K_f cos(p) V_s)$$

The equation of motion for the elevation angle then has the final form:

$$J_e\ddot{e} = g(l_c m_c - 2 l_h m_p) cos(e) + l_h K_f V_s cos(p) \tag{2}$$

Therefore $L_2 = g(l_c m_c - 2l_h m_p)$, and $L_3 = l_h K_f$.

Finally, the equation of motion for the travel angle is found through the momentum around the travel axis in the clockwise direction, as shown in fig. 1. As seen in fig. 4, the only forces with a moment arm perpendicular to the travel axis are the components of the motor forces in the horizontal direction, $F_{f,h} = F_f sin(p)$ and $F_{b,h} = F_b sin(p)$, has fig. 3 have the moment arm of length $arm_h = l_h cos(e)$:

$$\begin{aligned} J_\lambda \ddot{\lambda} &= arm_h(F_{f,h} + F_{b,h}) \\ &= l_h cos(e)(K_f sin(p)(V_f + V_b)) \end{aligned}$$

By substituting $V_s = V_f + V_b$, the final equation of motion for the travel angle becomes:

$$J_\lambda \ddot{\lambda} = l_h K_f V_s cos(e) sin(p) \tag{3}$$

Therefore $L_4 = l_h K_f$.

Equations (1) to (3) correspond respectively to equations (2a) to (2c) from the assignment [2, p.13].

1.2 Problem 2

To linearize the system about the point with all state variables equal to zero ($(p, e, \lambda)^T = (\dot{p}, \dot{e}, \dot{\lambda})^T$ = (0, 0, 0)) the inputs in the equations of motion (V_s^* and V_d^*) must be set to values that make this an equilibrium point.

At the linearization point, the equation of motion for pitch, eq. (1), reduces to $0 = L_1 * V_d^*$ therefore:

$$V_d^* = 0 \tag{4}$$

At the linearization point, the equation of motion for elevation, eq. (2), reduces to $0 = L_2 + L_3 * V_s^*$ therefore:

$$V_s^* = -L_2/L_3 \tag{5}$$

While the equation of motion for travel, eq. (3), at the linearization point simply reduces to $0 = 0$. The following transformation is performed to simplify the analysis [2, p.14]:

$$\begin{bmatrix} \tilde{p} \\ \tilde{e} \\ \tilde{\lambda} \end{bmatrix} = \begin{bmatrix} p \\ e \\ \lambda \end{bmatrix} - \begin{bmatrix} p^* \\ e^* \\ \lambda^* \end{bmatrix} \text{ and } \begin{bmatrix} \tilde{V}_s \\ \tilde{V}_d \end{bmatrix} = \begin{bmatrix} V_s \\ V_d \end{bmatrix} - \begin{bmatrix} V_s^* \\ V_d^* \end{bmatrix} \tag{6}$$

The equations of motion in the transformed system are therefore:

$$J_p \ddot{\tilde{p}} = L_1 \tilde{V}_d \tag{7a}$$
$$J_e \ddot{\tilde{e}} = L_2 cos(\tilde{e}) + L_3(\tilde{V}_s + L_2/L_3) cos(\tilde{p}) \tag{7b}$$
$$J_\lambda \ddot{\tilde{\lambda}} = L_4(\tilde{V}_s + L_2/L_3) cos(\tilde{e}) sin(\tilde{p}) \tag{7c}$$

By choosing the state to be x = ($\tilde{p}$, $\tilde{e}$, $\tilde{\lambda}$, $\dot{p}$, $\dot{e}$,$\dot{\lambda}$) the nonlinear state equations become:

$$\dot{x}_1 = x_4 \tag{8a}$$
$$\dot{x}_2 = x_5 \tag{8b}$$
$$\dot{x}_3 = x_6 \tag{8c}$$
$$\dot{x}_4 = (L_1/J_p) V_d \tag{8d}$$
$$\dot{x}_5 = (L_2/J_e) cos(x_2) + (L_3/J_e)(V_s + L_2/L_3) cos(x_1) \tag{8e}$$
$$\dot{x}_6 = (L_4/J_\lambda)(V_s + L_2/L_3) cos(x_2) sin(x_1) \tag{8f}$$

If the above system is expressed as $\dot{x} = h(x, u)$, where x is the state and u is the input, the system is linearized by finding the Jacobians of h with respect to the state and the input and then inserting the equilibrium values.

$$\frac{\partial h}{\partial x} = A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{L_4 L_2}{J_\lambda L_3} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial h}{\partial u} = B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & L_1/J_p \\ L_3/J_e & 0 \\ 0 & 0 \end{bmatrix} \tag{9}$$

where L_1, L_2, L_3 and L_4 were calculated in section 1.1 and J_p, J_e and J_λ were given in the assignment description [2, p.14]. The linearized equations of motion can therefore be written in the following form:

$$\ddot{\tilde{p}} = K_1 \tilde{V}_d \qquad K_1 = \frac{L_1}{J_p} \tag{10a}$$

$$\ddot{\tilde{e}} = K_2 \tilde{V}_s \qquad K_2 = \frac{L_3}{J_e} \tag{10b}$$

$$\ddot{\tilde{\lambda}} = K_3 \tilde{p} \qquad K_3 = \frac{L_4 L_2}{J_\lambda L_3} \tag{10c}$$

1.3 Problem 3

Check if we added gain to the output of joystick

The helicopter is difficult to control using only feed-forward. The physical behaviour of the helicopter differs from the (2a) - (2c) [2, p.13] model because it does not take into consideration drag, ground effects, etc.

In eq. (10) the model is linearized around a point with all angles equal to zero, where elevation is defined as zero when the elevation arm is parallel to the table, pitch is defined as zero when the pitch arm is parallel to the table and travel angle is defined as zero at the initial travel angle. Since the physical system is not linear, the linear assumption breaks down when the system is too far from this point.

1.4 Problem 4

Elevation is the only variable that needs to be changed, since travel and pitch are zeroed correctly upon initialization. To correct for the offset of the arm at startup, subtract 30 degrees from the elevation to get the correct point of zero elevation.

The value of V_s^* in order to stabilize the helicopter at the equilibrium point, was measured to be 7.5 V.

The motor force constant, K_f, which relates F_f and V_f is calculated to be $-(L_2/l_h) * V_s^*$, and has the value 0.1332.

2 Part 2 – Monovariable control

2.1 Problem 1

The following controller form is given in the assignment[2, p.15]:

$$\tilde{V_d} = K_{pp}(\tilde{p_c} - \tilde{p}) - K_{pd}\dot{\tilde{p}} \tag{11}$$

Substituting in the equation for pitch angle (eq. (1)) gives the following.

$$\ddot{\tilde{p}} = K_1 K_{pp}(\tilde{p_c} - \tilde{p}) - K_1 K_{pd}\dot{\tilde{p}} \tag{12}$$

To find the transfer function, $\frac{\tilde{p}(s)}{\tilde{p_c}(s)}$, the Laplace transform is taken.

$$\begin{gathered} \ddot{\tilde{p}} + K_1 K_{pd}\dot{\tilde{p}} + K_1 K_{pp}\tilde{p} = K_1 K_{pp}\tilde{p_c} \\ \mathcal{L} \rightarrow \\ s^2\tilde{p}(s) + sK_1 K_{pd}\tilde{p}(s) + K_1 K_{pp}\tilde{p}(s) = K_1 K_{pp}\tilde{p_c}(s) \end{gathered}$$

Which gives the transfer function

$$\frac{\tilde{p}(s)}{\tilde{p_c}(s)} = \frac{K_1 K_{pp}}{s^2 + K_1 K_{pd}s + K_1 K_{pp}} \tag{13}$$

The linearized pitch dynamics can be regarded as a second-order linear system, which means that by putting eq. (13) in the form shown in eq. (14), K_{pp} and K_{pd} can be determine from ω and ζ.

$$h(s) = \frac{\omega^2}{s^2 + 2\zeta\omega^2 s + \omega^2} \tag{14}$$

This gives the following relations:

$$\omega = \sqrt{K_1 K_{pp}} \tag{15}$$

$$2\zeta\omega^2 = K_1 K_{pd}$$

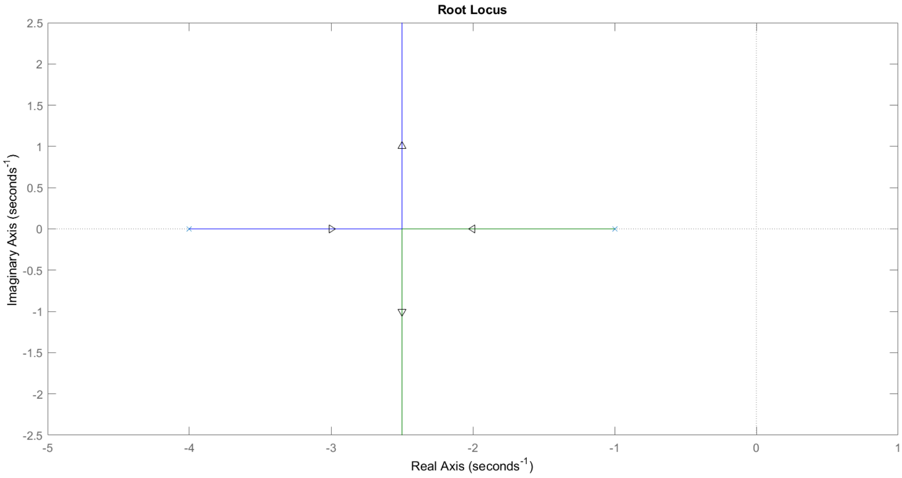
$$\zeta = \frac{K_1 K_{pd}}{2\omega^2} = \frac{K_{pd}}{2K_{pp}} \tag{16}$$

For a critically damped system $\zeta = 1$, which gives the following relationship

$$K_{pd} = 2K_{pp} \tag{17}$$

Beginning with a $K_{pp} = 3$, and then from the relation in eq. (17) a $K_{pd} = 6$, the response of the pitch angle to the input was slower than desired. Therefore, K_{pp} was increased to $K_{pp} = 12.5$ and K_{pd} was lowered to underdamp the system, until it was sufficiently responsive at $K_{pd} = 0.7K_{pp} = 8.75$. At these values the system responded faster with only minor oscillations. It was observed that larger values of K_{pp} gave rise to larger oscillations.

Figure 5: Change in pole position by increasing K_{pp} given a constant K_{pd}



At the critically damped point, the poles lie on the same point on the x-axis. When K_{pp} is decreased in relation to K_{pd} the system is over-damped the poles move away from each other along the x-axis. When K_{pp} is increased in relation to K_{pd}, the system is under damped and the poles move away from each other vertically from the critically damped point.

With the PD controller, it was significantly easier to control the helicopter than with just feed forward joystick control as no scaling of the joystick input was needed.

2.2 Problem 2

By plugging the P controller for travel

$$\tilde{p} = K_{rp}(\dot{\tilde{\lambda}}_c - \dot{\tilde{\lambda}}) \tag{18}$$

into the equation of motion for travel eq. (10c), the transfer function between $\dot{\tilde{\lambda}}$ and $\dot{\tilde{\lambda}}_c$ can be derived. By substituting the right side of the controller into the equation of motion for travel the equation becomes:

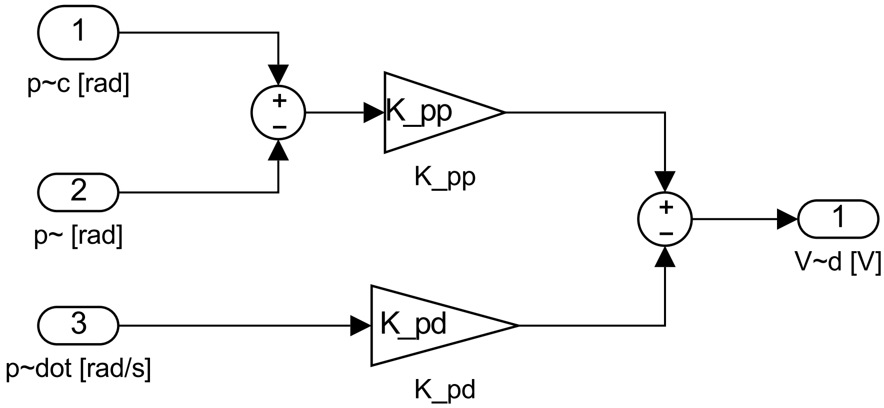
$$\ddot{\tilde{\lambda}} = K_3(K_{rp}(\dot{\tilde{\lambda}}_c - \dot{\tilde{\lambda}})) \tag{19}$$

After taking the laplace transform and rearranging terms, the transfer function is as follows:

$$\frac{\dot{\tilde{\lambda}}(s)}{\dot{\tilde{\lambda}}_c(s)} = \frac{K_3 K_{rp}}{s + K_3 K_{rp}} \tag{20}$$

This controller was quite quick and stable after correctly tuning K_{rp}. The value that was deemed best was K_{rp} = -40. Higher values, ie. more negative values, of K_{rp} led to large oscillations, while lower values, or less negative values, of K_{rp} led to a slower response to the joysticks input. It was necessary to add a gain to the x value of the joystick to limit the input range of the controller.

Figure 6: Simulink implementation of PD controller



3 Part 3 - Multivariable control

3.1 Problem 1

Given the state variables:

$$\boldsymbol{x} = \begin{bmatrix} \tilde{p} \\ \dot{\tilde{p}} \\ \dot{\tilde{e}} \end{bmatrix} \tag{21}$$

and eq. (10), the system matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are as follows:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & K_1 \\ K_2 & 0 \end{bmatrix} \tag{22}$$

3.2 Problem 2

The rank of the controllability matrix, $\boldsymbol{\mathcal{C}}$, determines whether the system is controllable.

$$\boldsymbol{\mathcal{C}} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A^2B} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & K_1 & 0 & 0 \\ 0 & K_1 & 0 & 0 & 0 & 0 \\ K_2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{23}$$

which has full rank: $rank(\boldsymbol{\mathcal{C}}) = 3$, and is thus controllable.

The system can be controlled by adding an LQR controller with reference feed-forward, $u = \boldsymbol{Pr} - \boldsymbol{Kx}$. K is derived from the Matlab lqr function, which requires the A and B matrices, in addition to weighting matrices Q and R. When finding appropriate Q and R matrices, Byrson's rule guided the first iteration. The rule states:

$$Q_{ii} = \frac{1}{\text{maximum acceptable value of } x_i^2} \tag{24}$$

$$R_{jj} = \frac{1}{\text{maximum acceptable value of } u_j^2} \tag{25}$$

where x_i represents the i^{th} state and u_j represents the j^{th} input. All other entries to Q and R are 0. This resulted in:

$$Q = \begin{bmatrix} 1/(\pi/8)^2 & 0 & 0 \\ 0 & 1/(\pi/2)^2 & 0 \\ 0 & 0 & 1/(\pi/8)^2 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{26}$$

These initial values represent the decision to have a small range of motion in pitch, a large max speed in pitch to better control the pitch and a relatively slow maximum speed in elevation. Furthermore, both inputs max value is 1.

After tuning, the following Q and R matrices seemed to perform with the quickest response without large overshoots:

$$Q = \begin{bmatrix} 60 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 100 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{27}$$

A higher value for $q_{1,1}$ yields a more oscillatory pitch behavior, while a lower value yields a slower response. As for $q_{2,2}$ a higher value slowed the pitch response. With $q_{3,3} > 100$ the system becomes difficult to control.

Explain why the system becomes difficult to control

$\boldsymbol{P}$ is defined such that as time goes to infinity, the states $\tilde{p}$ and $\dot{\tilde{e}}$ tend to their reference values $\tilde{p}_c$ and $\dot{\tilde{e}}_c$. This happens when $\dot{\boldsymbol{x}} = 0$, as the system reaches a stable equilibrium around the reference values:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{Ax} + \boldsymbol{Bu} \\ &= \boldsymbol{Ax} + \boldsymbol{B}(\boldsymbol{Pr} - \boldsymbol{Kx}) \\ &= (\boldsymbol{A} - \boldsymbol{BK})\boldsymbol{x} + \boldsymbol{BPr} = 0 \end{aligned}$$

When $\dot{\boldsymbol{x}} = 0$, $\boldsymbol{x}$ has reached its final value, so $\boldsymbol{x} = \boldsymbol{x}_{\boldsymbol{\infty}}$:

$$\begin{aligned} (\boldsymbol{BK} - \boldsymbol{A})\boldsymbol{x}_{\boldsymbol{\infty}} &= \boldsymbol{BPr} \\ \Leftrightarrow \boldsymbol{x}_{\boldsymbol{\infty}} &= (\boldsymbol{BK} - \boldsymbol{A})^{-1}\boldsymbol{BPr} \\ \Rightarrow \boldsymbol{y}_{\boldsymbol{\infty}} = \boldsymbol{Cx}_{\boldsymbol{\infty}} &= \boldsymbol{C}(\boldsymbol{BK} - \boldsymbol{A})^{-1}\boldsymbol{BPr} \end{aligned}$$

Therefore, the output $\boldsymbol{y}_{\boldsymbol{\infty}}$ is equal to our reference $\boldsymbol{r}$ when:

$$\boldsymbol{P} = [\boldsymbol{C}(\boldsymbol{BK} - \boldsymbol{A})^{-1}\boldsymbol{B}]^{-1} \tag{28}$$

This achieves the desired result, that $\boldsymbol{y}$ goes to $\boldsymbol{r}$ as time goes to infinity:

$$\lim_{t\to\infty} \boldsymbol{y}(t) = \boldsymbol{y}_{\boldsymbol{\infty}} = \begin{bmatrix} \tilde{p} \\ \dot{\tilde{e}} \end{bmatrix} = \begin{bmatrix} \tilde{p}_c \\ \dot{\tilde{e}}_c \end{bmatrix} = \boldsymbol{r},$$

3.3 Problem 3

With integral effect, the new state space matrices become:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & K_1 \\ K_2 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{29}$$

The matrices used to calculate the controllers gains, Q and R, also needed to be updated:

$$Q = \begin{bmatrix} 60 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 30 \end{bmatrix} \qquad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{30}$$

Without integral effect, the system could track pitch without much of a problem. But the elevation rate had a noticeable deviation in the ranges of elevation not close to our linearization point $\dot{e} = 0$.

With integral effect, this deviation in elevation rate is asymptotically removed. This means that with a zero Y output from our joystick, there is a corresponding near zero elevation rate, and the elevation angle also stays asymptotically constant for angles not close to its linearization point.

Below, fig. 7 and fig. 8 show the input and output of an automatic run of the LQR controller without integral effect, while fig. 9 and fig. 10 show the input and output of an automatic run of the LQR controller with integral effect. One noticable difference in the input is the need for a small burst in y at the beginning of the run to increase the helicopter to 0 degrees in elevation. This is because this controller does a better job regulating elevation rate, and with an input of 0, it would fly very close to the table at the start. Aside from this the 2 input routines are identical. The output is very similar in pitch and travel. However, in elevation, at the end of the routine without integral effect, the elevation angle returns to zero. While with integral effect, although there is a bounce back, the controller does not bring the helicopter's elevation angle back to zero, but instead brings the elevation rate to zero.

Figure 7: Input for no Integral Effect

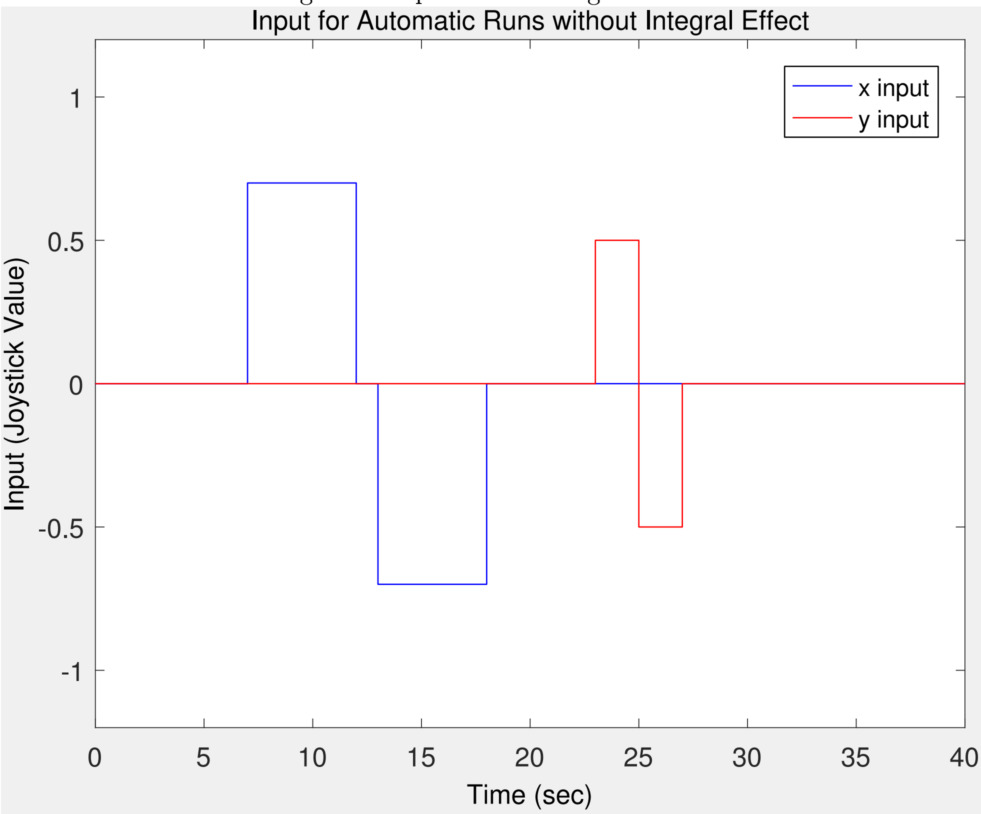


Figure 8: LQR Controller with no Integral Effect

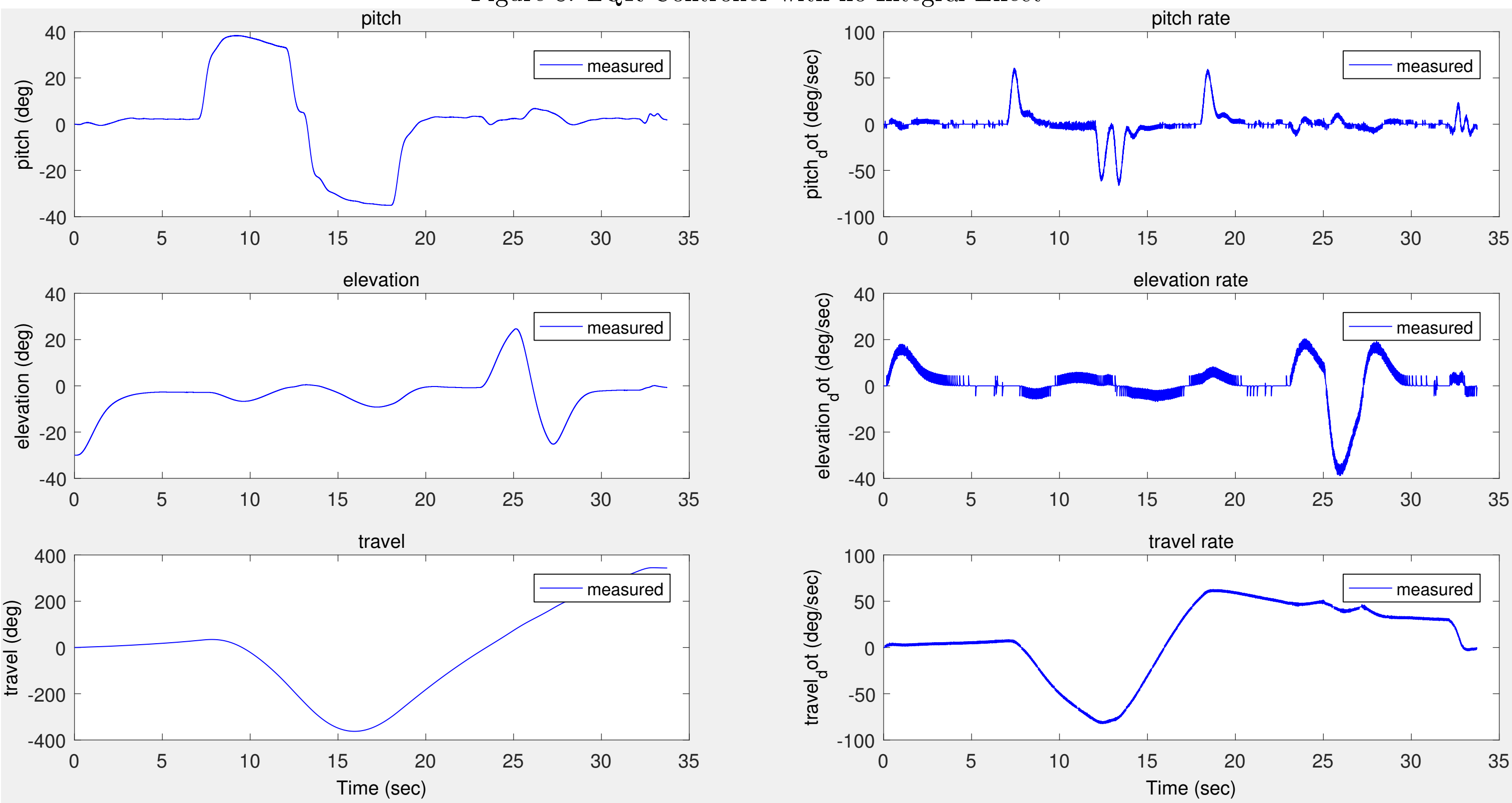


Figure 9: Input with Integral Effect

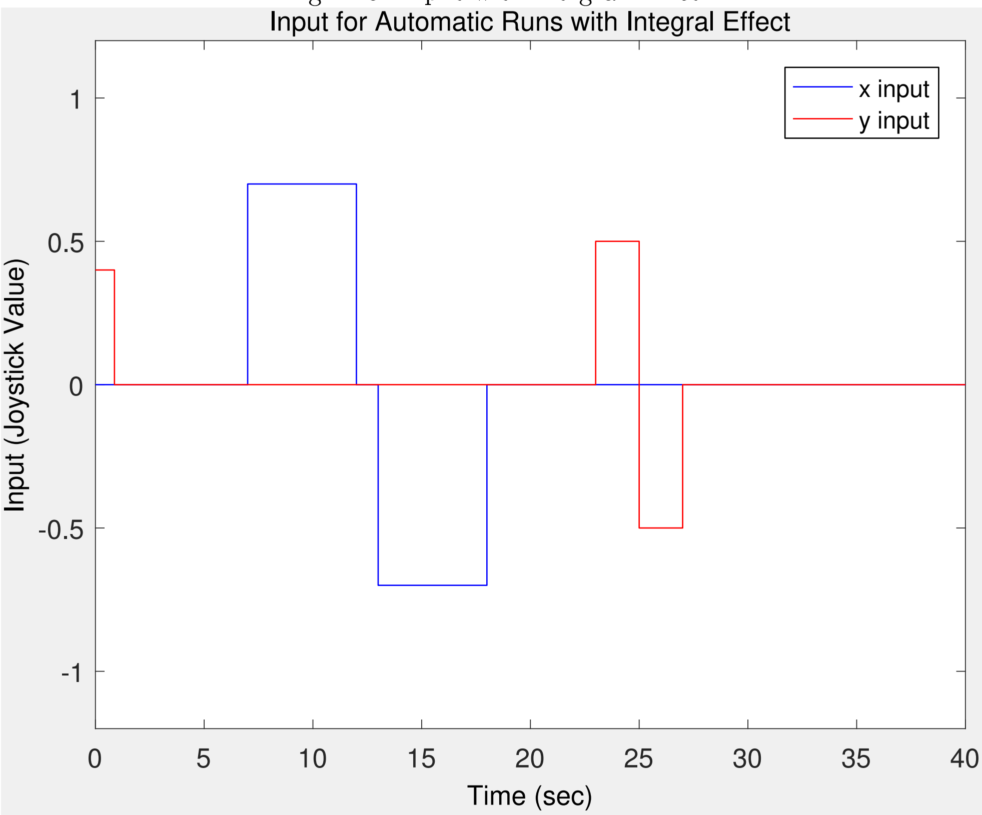
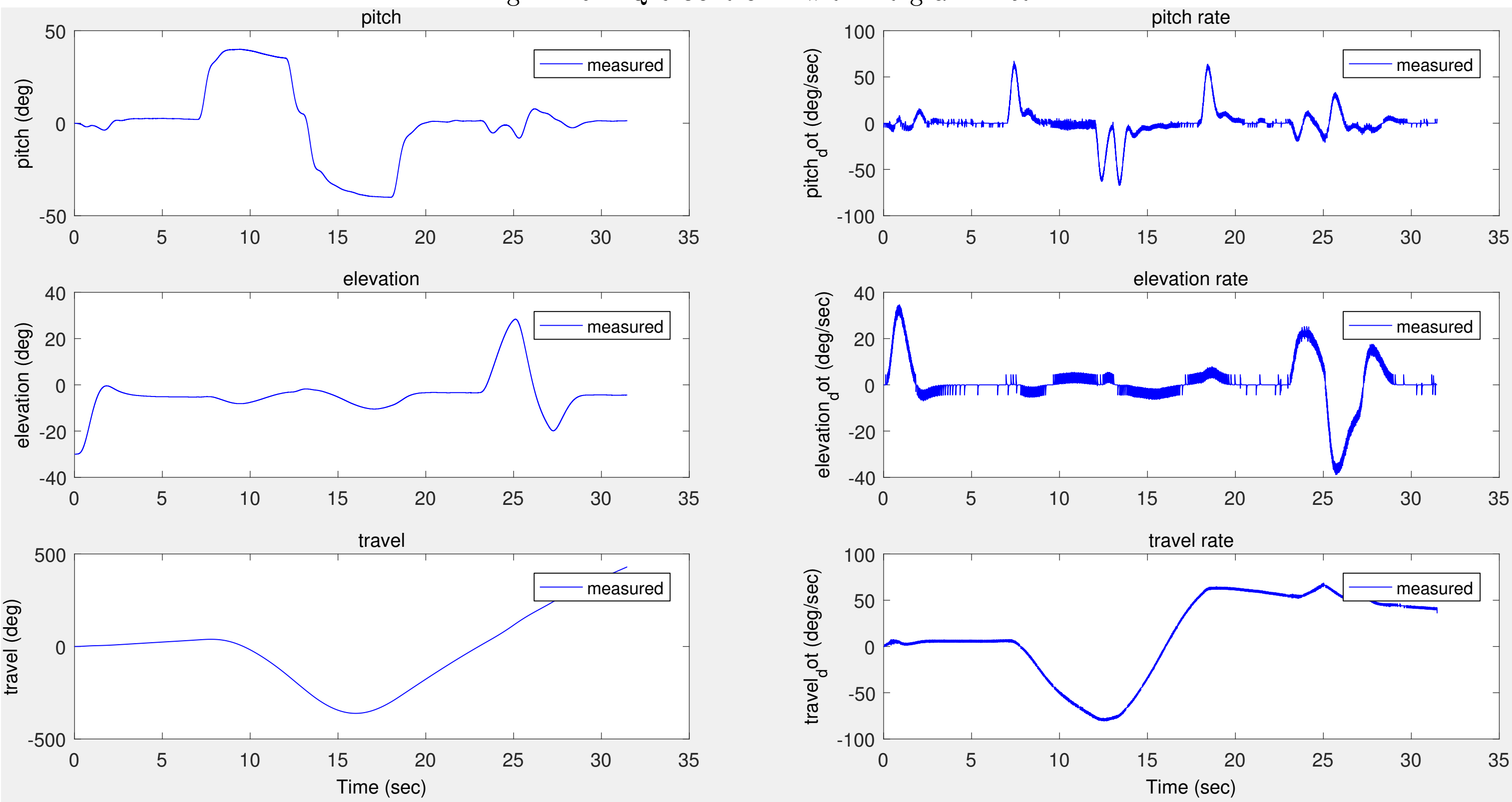


Figure 10: LQR Controller with Integral Effect



4 Part 4 – State estimation

This section consists of the development of an observer to estimate the nonmeasured angular velocities.

4.1 Problem 1

By describing the system in eq. (10) in the following state-space form

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x} \end{aligned} \tag{31}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are matrices. The state -, input - and output vector are given by

$$\boldsymbol{x} = \begin{bmatrix} \tilde{p} \\ \dot{\tilde{p}} \\ \tilde{e} \\ \dot{\tilde{e}} \\ \tilde{\lambda} \\ \dot{\tilde{\lambda}} \end{bmatrix}, \quad \boldsymbol{u} = \begin{bmatrix} \tilde{V_s} \\ \tilde{V_d} \end{bmatrix} \quad \text{andc} \quad \boldsymbol{y} = \begin{bmatrix} \tilde{p} \\ \tilde{e} \\ \tilde{\lambda} \end{bmatrix} \tag{32}$$

This gives the following $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ matrices

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ K_3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & K_1 \\ 0 & 0 \\ K_2 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{33}$$

Where K_1, K_2 and K_3 are given by eq. (10).

4.2 Problem 2

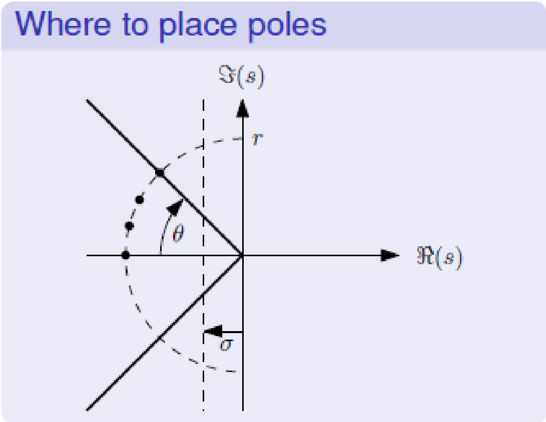
The observer matrix can be used. For a 6 state system, it is defined by:

$$\boldsymbol{\mathcal{O}} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \boldsymbol{C} * \boldsymbol{A}^2 \\ \boldsymbol{C} * \boldsymbol{A}^3 \\ \boldsymbol{C} * \boldsymbol{A}^4 \\ \boldsymbol{C} * \boldsymbol{A}^5 \end{bmatrix} \tag{34}$$

This can be calculated using MATLAB's $obsv(\boldsymbol{A}, \boldsymbol{C})$ function. The resulting 18x6 matrix has rank 6, thereby full rank. Since it has full rank, the system is fully observable.

The observer gain matrix $\boldsymbol{L}$ is to be set in such a way that the poles of the observer are faster than the system, in order to drive the error to zero.

Figure 11: illustrating how to place poles during state, or estimated state feedback, on a semi-circle with the same radius, within the region shown.



Finn navnene på theta og sigma i forhold til regtek

It is recommended that the poles of the observer be placed as in fig. 11. The real value of the observer poles must be larger than σ, which for the observer is the value of the largest real value of the controlled systems poles. This way, all of the linear observers poles are such that the observer is faster than the controlled system. θ is the largest angle of the observer poles. If this is too large, the system will be underdamped and cause overshoots in the estimate. However, if the radius is too large, undesired high frequency noise from the measurements becomes amplified to unwanted levels. Furthermore, the observer will become increasingly unstable the closer the poles are.

$\theta = 15$, and r as 50 times the maximum length of the controlled systems poles worked well for this system.

The observer itself has the state space formulation:

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{L}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}})$$

$$\begin{aligned}
\dot{\boldsymbol{e}} &= \dot{\tilde{\boldsymbol{x}}} - \dot{\hat{\boldsymbol{x}}} \\
&= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} - \boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{B}\boldsymbol{u} - \boldsymbol{L}(\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{x}}) \\
&= \boldsymbol{A}(\boldsymbol{x} - \hat{\boldsymbol{x}}) - \boldsymbol{L}(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{C}\hat{\boldsymbol{x}}) \\
&= \boldsymbol{A}\boldsymbol{e} - \boldsymbol{L}\boldsymbol{C}\boldsymbol{e}
\end{aligned}$$

Meaning the error has the state space formulation:

$$\dot{\boldsymbol{e}} = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\boldsymbol{e} \tag{35}$$

As previously stated, for this error to converge to zero the poles of this state space formulation should be placed such that they are faster than the poles of the system itself. The poles of this state space system can be placed arbitrarily because $\{\boldsymbol{A}, \boldsymbol{C}\}$ is observable:

$$det(\lambda\boldsymbol{I} - \boldsymbol{A} + \boldsymbol{L}\boldsymbol{C}) = 0$$

For this equation, λ are the values that solves the equation, and also the poles of the observer. By choosing values for λ, an $\boldsymbol{L}$ emerges in order to make the determinant equal to zero. The matlab function *place* places the poles as desired for us:

$\boldsymbol{L} = place(\boldsymbol{A}^T, \boldsymbol{C}^T, \boldsymbol{\lambda})^T$, where $\boldsymbol{\lambda}$ is the vector of the observer poles in this instance.

In fig. 12 the poles are palced on the real line at -20, -40, -60, -80, -100, and -120. This type of observer has minimal overshoot, as there are no complex parts, and the poles are spread far apart to avoid unstable behavior from the observer. Furthermore, the maximum radius is also rather large leading to a fast response. Placing the poles on the real line like this was slightly better than placing them evenly on a circle with radius $r = 60$, and an angle $\theta = 22.5$, because with this pole setup the observer was too oscillatory.

This r and theta are different than the one mentions previously (about 2 paragraphs back

Figure 12: LQR Controller Estimating p, e and λ

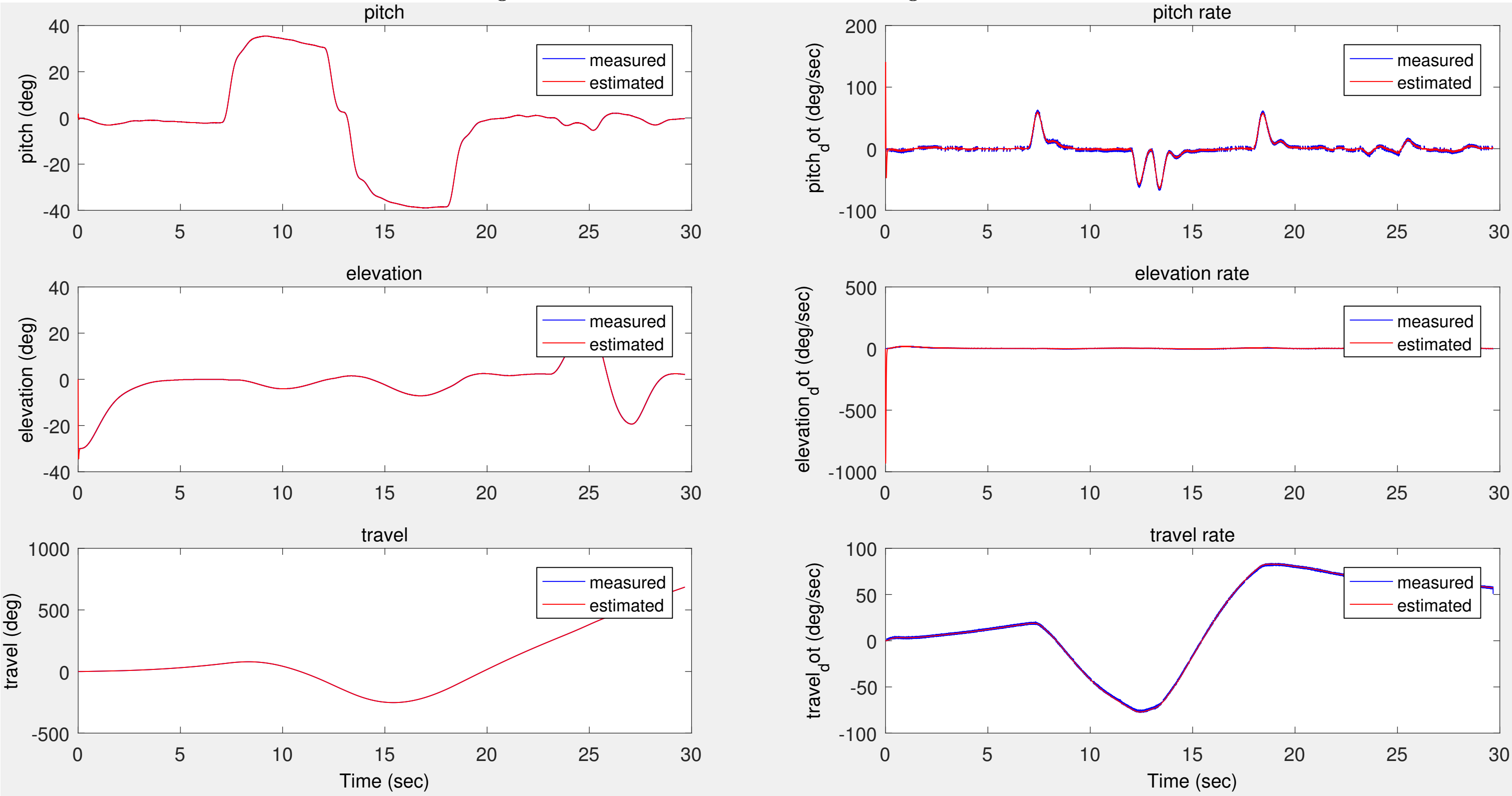


Figure 13: LQR Controller with Integral Effect Estimating p, e and λ

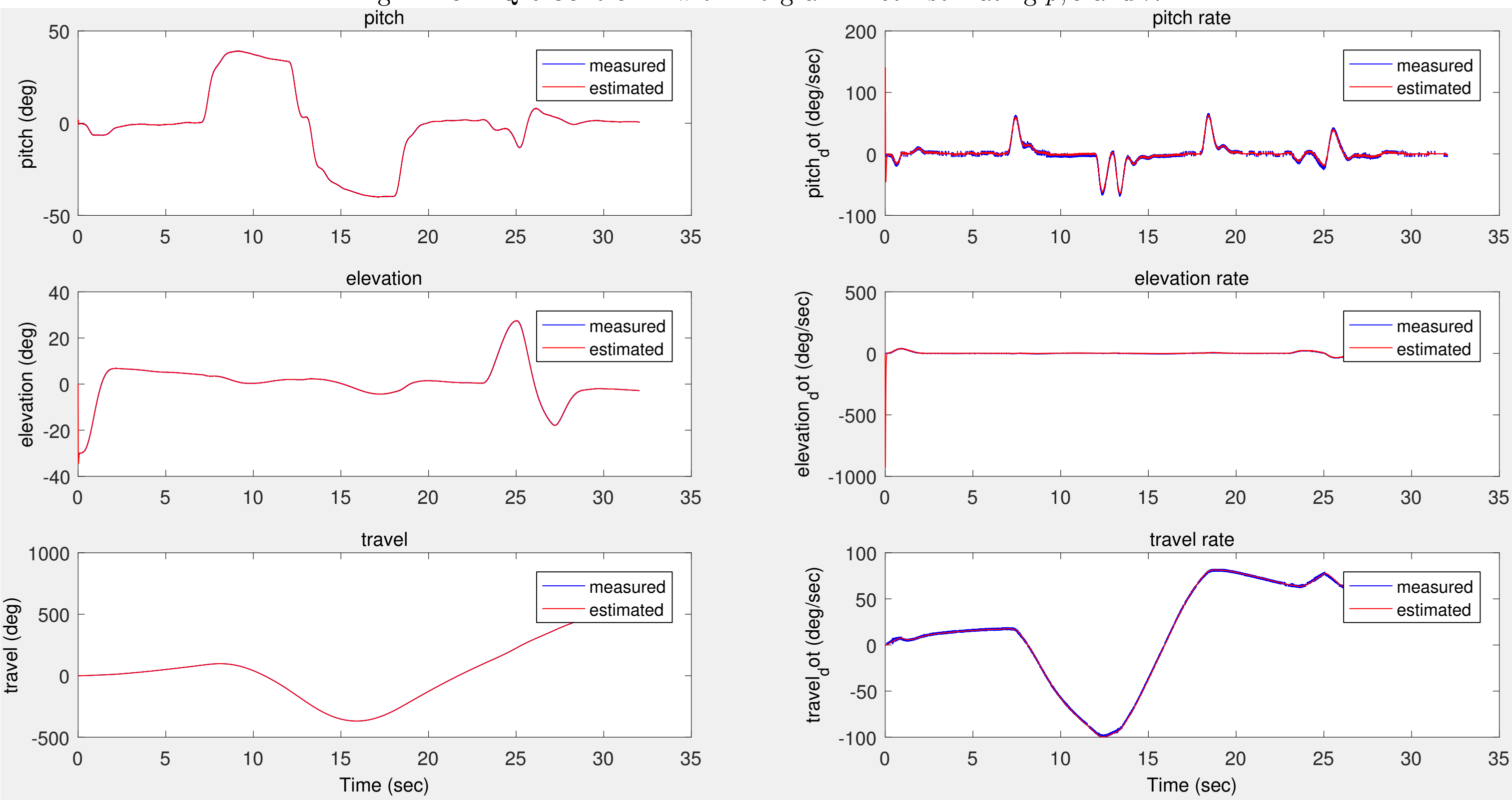


Figure 14: LQR Controller Estimating p, e and λ

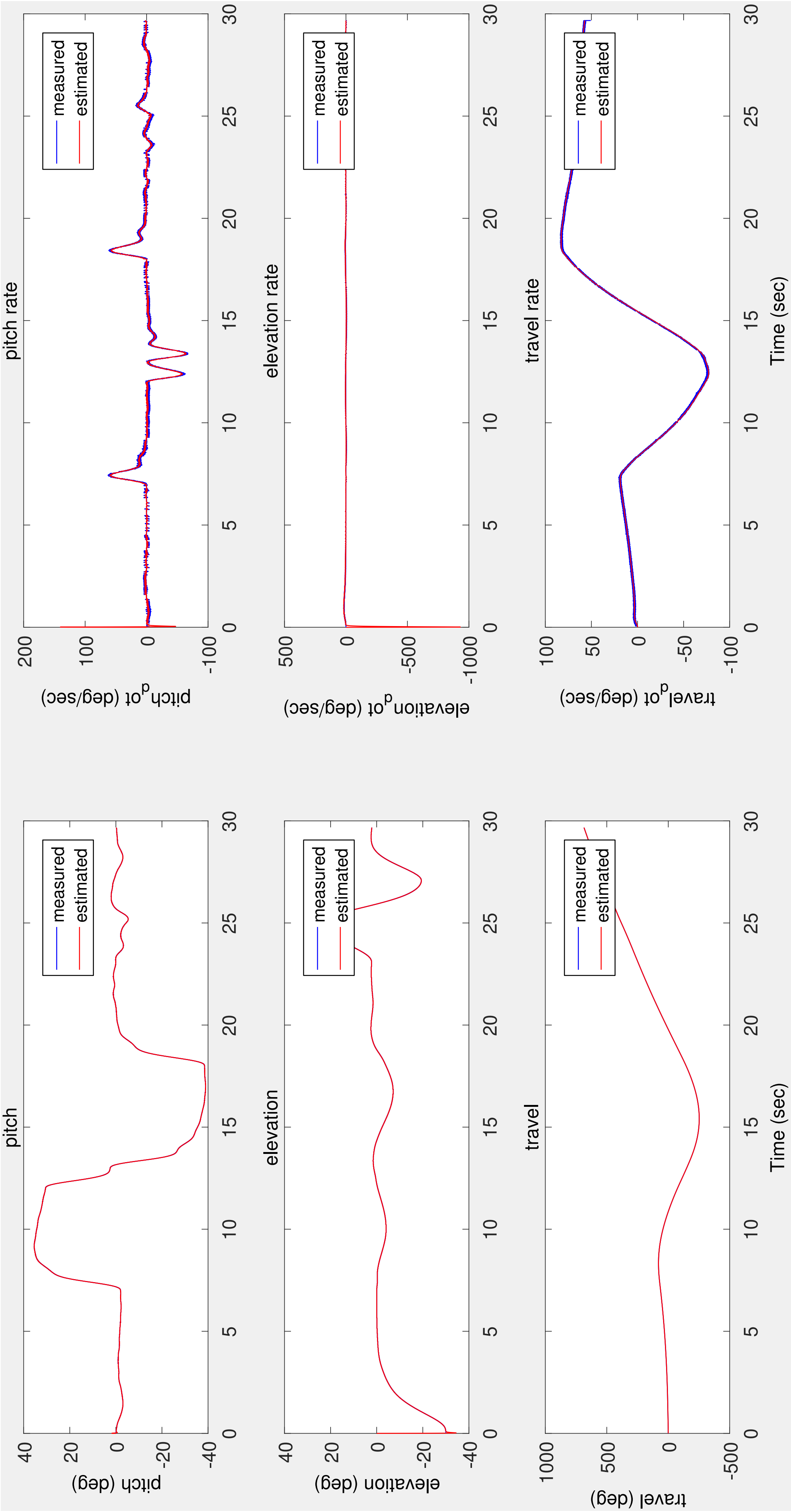
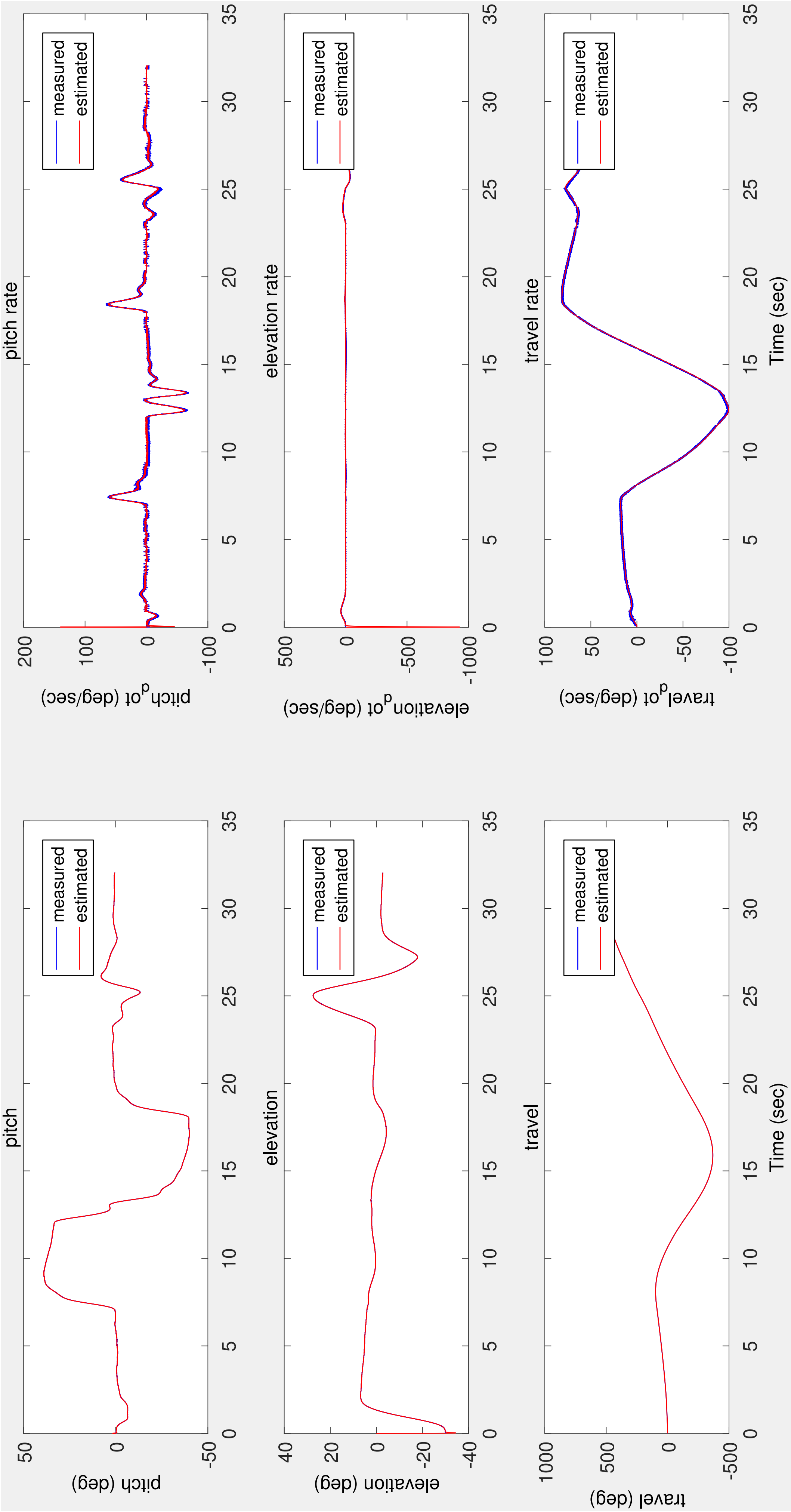


Figure 15: LQR Controller with Integral Effect Estimating p, e and λ



4.3 Problem 3

When only $\tilde{e}$ and $\tilde{\lambda}$ are measured, the output matrix becomes:

$$\boldsymbol{C} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

The observer matrix, found through $obsv(\boldsymbol{A}, \boldsymbol{C})$ is a 12x6 matrix with rank 6. Therefore, it is observable.

However, when only $\tilde{p}$ and $\tilde{e}$ are measured, the output matrix becomes:

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The observer matrix, found through $obsv(\boldsymbol{A}, \boldsymbol{C})$ is a 12x6 matrix with rank 4. Therefore, it is not observable.

This observer is particularly challenging to implement due to the way that pitch is estimated. The pitch estimate is based on the elevation and travel error of the observer, and the impact of the input $\boldsymbol{u}$ on our model and the derivative of the estimated travel rate. This last term of the pitch estimate becomes a double derivative of measured travel, which significantly amplifies measurement noise.

The observer desirably filters out noise if the poles' negative real value σ in fig. 11 is not too large. However, if this value is too low, the observer becomes too slow and cannot accurately estimate the states. When the double derivation introduces large amounts of noise, the challenge emerges as a balancing act of making σ large enough so that the observer accurately estimates the states, while not making it so large that the noise becomes undesirably amplified.

One must also wisely set θ as seen in fig. 11 to a high enough value to make the system the right amount of underdamped with not too much oscillation and good settling time, and spread the poles such that they are not close enough to introduce unstable behavior [1, p.290]. It can also help to reduce the poles of the controlled system itself to help make the observer slower in order to counteract the amplified noise while allowing the observer to keep up with the controlled system. This is also limited however, as the controller eventually becomes too slow for efficient control.

In the tuning process, Q and R began at their values from Part 5.4.2, but eventually were reduced to make the controller slower. Many different pole placements for the observer were tried, including on a circle as suggested by Chen, on the real line as was done in 5.4.2 with varying uniform lengths between each pole, on a vertical line with the spread between them decided by θ, and several others [1, p.290].

What ended up allowing for reasonable control of the system was spreading the poles as much as possible in between the real values of -50 and -3, while keeping θ as low as possible. We ended up using poles $\boldsymbol{\lambda} = [-3, -20-4i, -20+4i, -30-4i, -30+4i, -50]^T$, $Q = diag([30, 0.1, 50, 1, 15])$, and $R = diag([0.1, 0.1])$.

References

[1] Chi-Tsong Chen, *Linear System Theory and Design*, Oxford University Press, 4th edition, 2014

[2] Kristoffer Gryte et al., *Helicopter lab assignment*, Department of Engineering Cybernetics, NTNU, Version 4.5, 2015

[3] Morten D. Pedersen, *TTK4115 Lecture 5 State Feedback*, Department of Engineering Cybernetics, NTNU, 19.09.2016