Boat Lab Report

Group 8

Halvor Ødegård Teigen, 478422
Simen Krantz Knudsen, 478444
Ole-Jørgen Hannestad, 478457

November 22, 2018

Contents

Introduction

The main goal of this lab exercise is to simulate and control a ship using classical control and linear system theory. The exercise consists of five parts; identification of the model parameters by using transfer functions and Power Spectral Density, implementing an autopilot using classical control theory, checking the observability for different subsystems, and lastly implementing a discrete Kalman filter in MATLAB and Simulink.

Subjects covered include Bode plots, stochastic states and measurements, calculation of observability of different linear systems, and the discrete Kalman Filter. Throughout the report, our findings and the mathematical derivations are presented. The reader is assumed to have a basic understanding of calculus and control theory, as we skip some of the simplest steps in the derivations.

1 Assignment 1 Identification of the boat parameters

1.1 Problem 1 a)

The model used throughout the tasks is given by the following set of equations. ψ_w is the high-frequency component of the compass course due to the wave disturbance, ξ_w is the accumulated compass heading due to ψ_w, ψ is the average heading without wave disturbance, r is the rate of change for the average heading, and b is the bias to the rudder angle. The v denotes the measurement noise, w_w is a zero mean white noise process,and w_b is Gaussian white noise.

$$\dot{\xi_w} = \psi_w \tag{1a}$$
$$\dot{\psi}_w = -\omega_0^2 \xi_w - 2\lambda\omega_0\psi_w + K_w w_w \tag{1b}$$
$$\dot{\psi} = r \tag{1c}$$
$$\dot{r} = -\frac{1}{T}r + \frac{K}{T}(\delta - b) \tag{1d}$$
$$\dot{b} = w_b \tag{1e}$$
$$y = \psi + \psi_w + v \tag{1f}$$

To derive the transfer function from δ to ψ, denoted H_{ship}, eq. (1c) is inserted into eq. (1d) and the term b is set to zero; i.e. assuming no underwater current. Next, the Laplace transform is applied and the expression is rearranged. The Bode diagram of H_{ship} is shown in fig. 1, with a phase margin of 16.9° and a crossover frequency of about 0.04. As the phase angle never crosses the $-180°$ line, the gain margin is in theory infinite.

$$\begin{aligned} & \ddot{\psi} = -\frac{1}{T}\dot{\psi} + \frac{K}{T}\delta \\ \overset{\mathcal{L}\{\}}{\Longrightarrow} \quad & H_{ship}(s) = \frac{\psi}{\delta}(s) = \frac{K}{s(1+Ts)} \end{aligned} \tag{2}$$

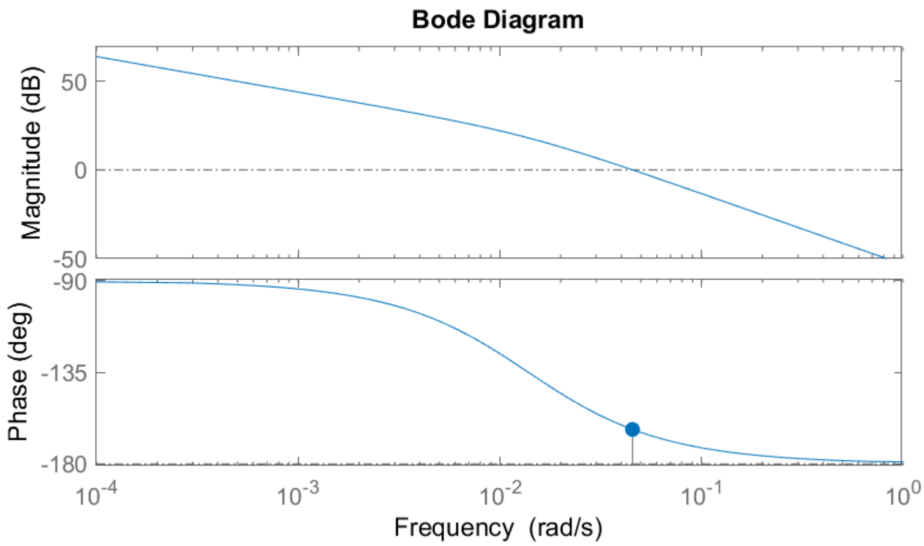

Figure 1: Bode diagram of the open loop model H_{ship}, without a controller.

1.2 Problem 1 b)

In this problem, the disturbances and the measurement noise are turned off. To identify the unknown parameters T and K in eq. (2), we applied two sine waves with an amplitude of 1, but with different frequencies, to the input δ. $|H_{ship}(j\omega_1)|$ and $|H_{ship}(j\omega_2)|$ are given by the amplitudes in the output, when the frequencies $\omega_1 = 0.005 \frac{rad}{s}$ and $\omega_2 = 0.05 \frac{rad}{s}$ are applied at the input. The plots of the output ψ are seen in figs. 2 and 3. The magnitude of H_{ship} is given by:

$$\left|\frac{\psi}{\delta}(j\omega)\right| = \frac{K}{\sqrt{T^2\omega^4 + \omega^2}} = \frac{K}{\omega} \cdot \frac{1}{\sqrt{1 + T^2\omega^2}} \tag{3}$$

When finding the maximum and minimum values of the response, we disregard the initial transient and just look at the harmonic oscillations. The maximum value of these oscillations in fig. 2 is approximately 63.3582 and the minimum value is 4.6417, this corresponds to an amplitude $A_{\omega_1} = 29.358$. The maximum value of the oscillations in fig. 3 is 4.2308 and the minimum value is 2.5693. This corresponds to an amplitude $A_{\omega_2} = 0.831$.

Next, the two frequencies ω_1 and ω_2 are inserted into eq. (3) to obtain two expression. After setting these expressions equal to the amplitudes A_{ω_1} and A_{ω_2}, and manipulating, we get two equations for the two unknowns K and T:

$$K^2 = (A_{\omega_1}\omega_1)^2(T^2\omega_1^2 + 1) \qquad\qquad T^2 = \frac{(A_{\omega_1}\omega_1)^2 - (A_{\omega_2}\omega_2)^2}{(A_{\omega_2}^2\omega_2^4 - A_{\omega_1}^2\omega_1^4)}$$

Solving these two expressions, by inserting the values mentioned in the paragraphs above, yield $K = 0.1561$ and $T = 72.4666$.

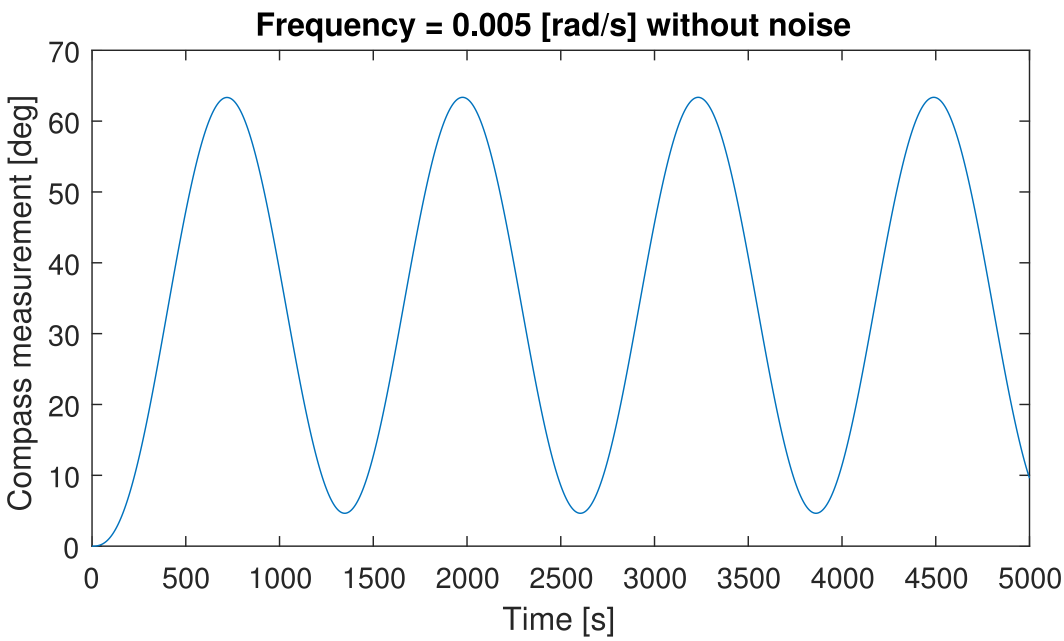

Figure 2: The output ψ, when applying a sine input with amplitude 1 and frequency $\omega_1 = 0.005$ on the input.

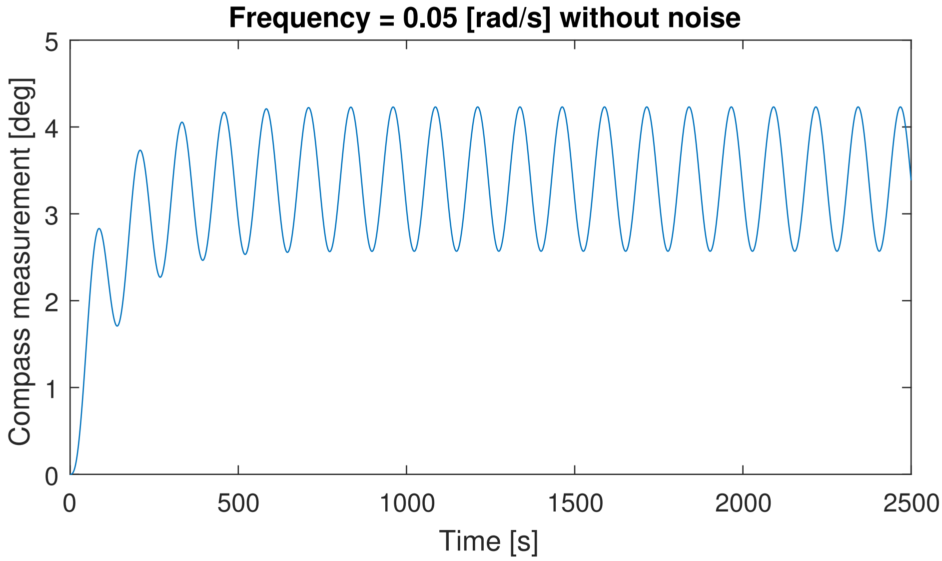

Figure 3: The output ψ, when applying a sine input with amplitude 1 and frequency $\omega_2 = 0.05$ on the input.

1.3 Problem 1 c)

Next, we turned the waves and measurement noise on and repeated the procedure for determining K and T, with the same frequencies as in section 1.2. Plots of these responses are seen in fig. 4 and fig. 5. We see that the noise is overpowering when the frequency gets too high, see fig. 5, because the amplitude of the high-frequency signal is relatively close to the amplitude of the measurement noise. This makes it difficult to find an accurate value for the amplitude of the high-frequency output, thus the parameters K and T are difficult to estimate in this case.

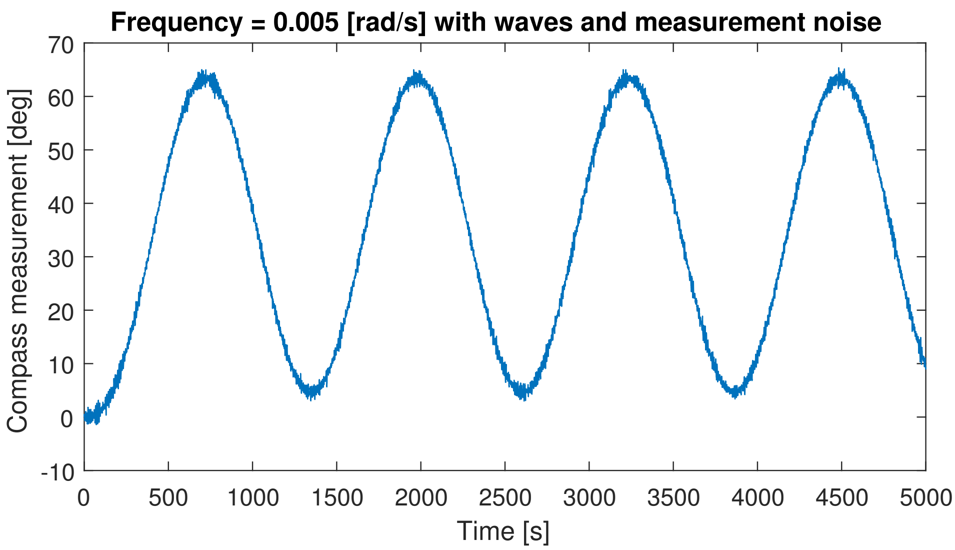

Figure 4: The output $|H(j\omega_1)|$, when applying a sine wave with amplitude 1 and $\omega_1 = 0.005$. Waves and measurement noise were active.

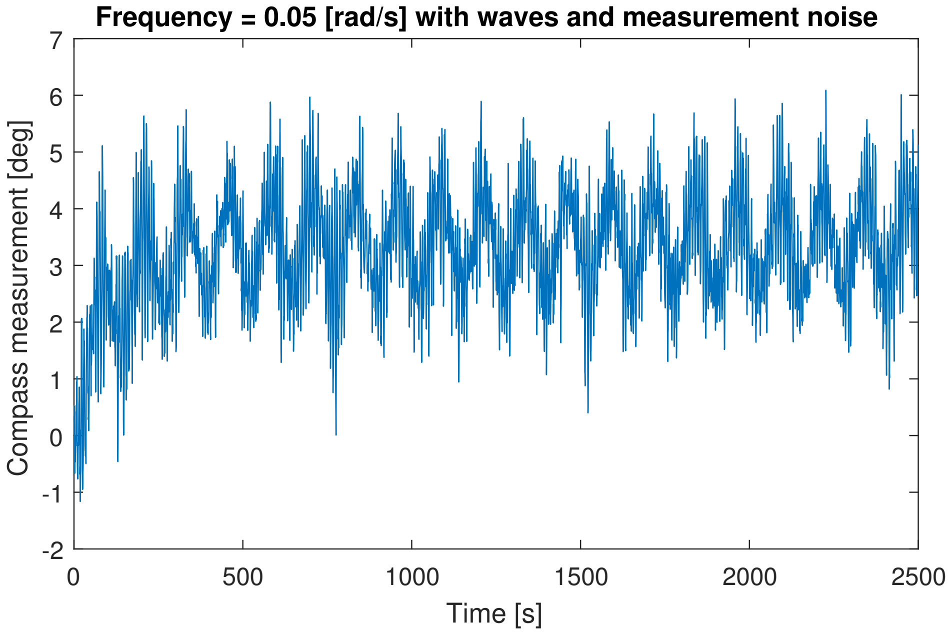

Figure 5: The output $|H(j\omega_2)|$, when applying a sine wave with amplitude 1 and $\omega_2 = 0.05$. Waves and measurement noise were active.

1.4 Problem 1 d)

To find out if the model of the ship, given by eq. (2), is a good approximation, a step of 1° is applied to the input δ. Figure 6 shows a small deviation between the model and the actual ship with increasing time. We see a noticeable difference from approximately 500 seconds, although this will be accounted for when the measurements are included as part of a controller later in the assignment. The model is a good approximation, as long as the rudder angle is not kept constant and different from zero for too long.

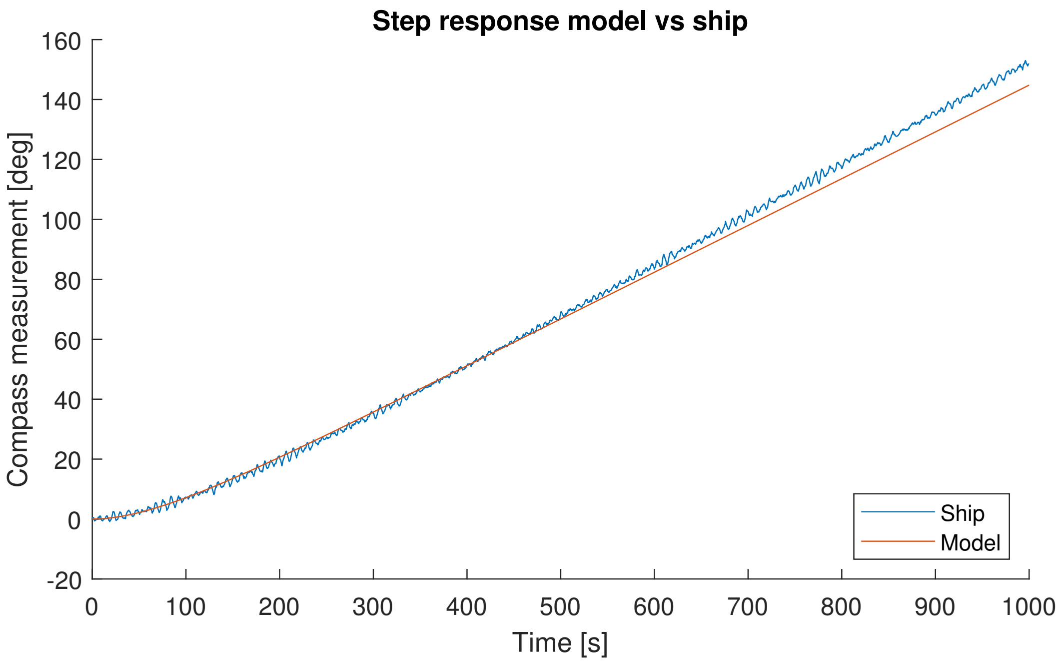

Figure 6: The step responses of the ship and the model, with 1° applied on the rudder input. The y-axis shows the compass measurement in degrees, and the plot lasted for about 16 minutes.

2 Assignment 5.2 Identification of the wave spectrum model

2.1 Problem 2 a)

In this assignment, the goal is to estimate the unknown parameters in eq. (1b); ω_0, λ, and K_w. To do so we estimate the Power Spectral Density (PSD) from measured data of ψ_w and fit it to an analytical expression of the PSD which contains these parameters.

The PSD shows a signal's distribution of power over its frequencies. To estimate the PSD of ψ_w, named $S_{\psi_w}(\omega)$, we utilized the MATLAB function `pwelch`, as shown in listing 1. This function uses the measured values for ψ_w and estimates the PSD based on this data. $S_{\psi_w}(\omega)$ is shown in fig. 7.

Listing 1: The MATLAB code for estimating the PSD of ψ_w

```
%% calculation
fs = 10;                      %Sampling frequency
window = 4096;                %Window size
x = psi_w(2,:) .* (pi/180); %Extract values and convert to rad

[pxx,f] = pwelch(x,window, [], [],fs); %Estimate PSD based on data
w = f .* (2*pi);              %Convert to rad/s
pxx = pxx ./ (2*pi);          %Convert to s/rad
```

2.2 Problem 2 b)

Inserting eq. (1a) into eq. (1b) and Laplace transforming leads to the following transfer function:

$$\dot{\psi}_w = -\omega_0^2 \int_0^t \psi_w(\tau)\,\mathrm{d}\tau - 2\lambda\omega_0\psi_w + K_w w_w$$

$$s\psi_w(s) = -\omega_0^2 \cdot \frac{1}{s}\psi_w(s) - 2\lambda\omega_0\psi_w(s) + K_w w_w(s)$$

$$\frac{\psi_w}{w_w}(s) = \frac{sK_w}{s^2 + 2\lambda\omega_0 s + \omega_0^2} \tag{4}$$

We now have a transfer function which takes in a white noise process w_w, and filters it so we get the ships reaction to the waves ψ_w. Because w_w is a white noise process with unity variance, its corresponding PSD, named $P_{w_w}(\omega)$, equals 1 for all frequencies [1, p. 120]. The formula for an analytic expression for the PSD of ψ_w is then given by:

$$P_{\psi_w}(\omega) = \left|\frac{\psi_w}{w_w}(j\omega)\right|^2 \cdot P_{w_w}(\omega) = \left|\frac{\psi_w}{w_w}(j\omega)\right|^2$$

Inserting eq. (4) into the expression above gives:

$$P_{\psi_w}(\omega) = \left|\frac{j\omega K_w}{-\omega^2 + 2\lambda\omega_0 j\omega + \omega_0^2}\right|^2 \quad \Longrightarrow$$

$$P_{\psi_w}(\omega) = \frac{\omega^2 K_w^2}{(\omega_0^2 - \omega^2)^2 + 4\lambda^2\omega_0^2\omega^2} \tag{5}$$

2.3 Problem 2 c)

Identifying ω_0 is the first step towards using the analytical expression in eq. (5). This is the system's resonance frequency, and will therefore be the frequency where S_{ψ_w} has its peak. By using the MATLAB function `max(pxx)`, where `pxx` is the estimated PSD of ψ_w, we get the peak value σ^2 and the index of the maximum value. Next, this index is used to look up the corresponding frequency of the maximum value. From this we get:

$$\omega_0 = 0.7823 \qquad \sigma^2 = 7.9191 \cdot 10^{-4} \tag{6}$$

2.4 Problem 2 d)

The constant K_w is defined as:

$$K_w = 2\lambda\omega_0\sigma \tag{7}$$

where λ denotes the damping factor of the damped harmonic oscillator in eq. (1b). By now, the values for the resonance frequency ω_0 and its corresponding intensity σ^2 are known, and we can apply the trial and error method with regards to the value of λ. Selecting different values for λ gives different values for K_w in eq. (7), and therefore different analytical expressions for the power spectral density in eq. (5). We concluded that $\lambda = 0.09$ resulted in the best fit between S_{ψ_ω} and P_{ψ_ω}, as shown in fig. 7. Calculating eq. (7) with the obtained values yields $K_w = 0.0396$.

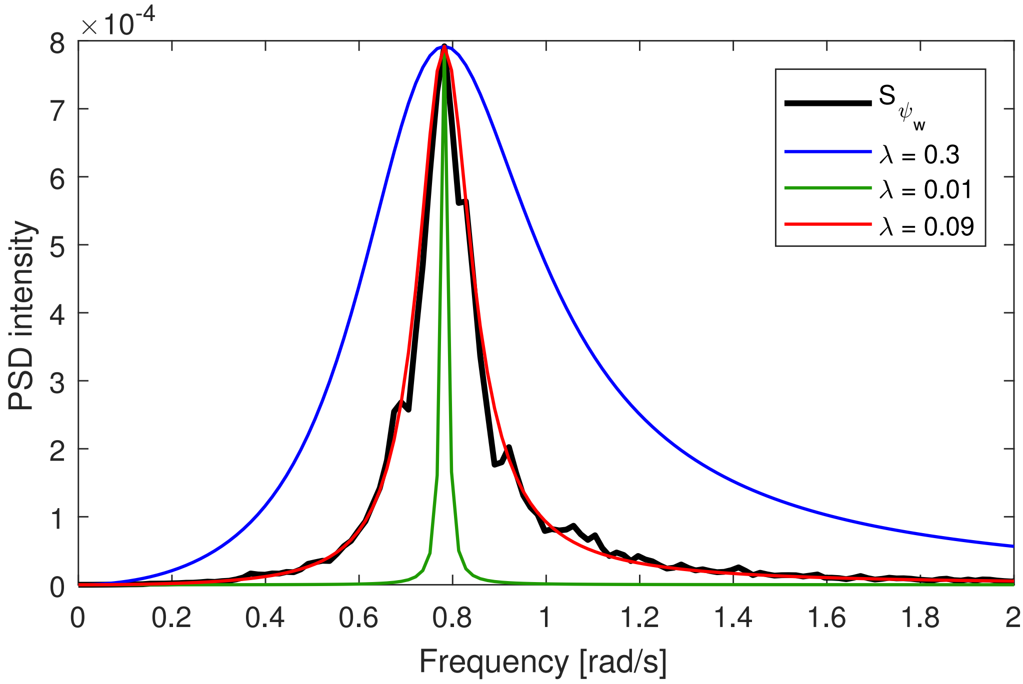

Figure 7: The estimate of the PSD of ψ_w, $S_{\psi_w}(\omega)$, plotted in black, with three plots of the analytic expression for the PSD function with different values of λ.

3 Assignment 5.3 Control system design

3.1 Problem 3 a)

In this subsection we will implement an autopilot for the ship utilizing classical control theory. For all of the tasks linked to implement this, the compass angle ψ will have to be held within the range of $\pm 35°$, provided by the given assignment. In addition, the measurement noise will be turned on for the rest of this report. The compass angle reference value is set to $\psi_r = 30°$, and the simulations in the following problems are done with an initial course angle of 0°. The following limited PD controller is given in the assignment:

$$H_{pd}(s) = K_{pd}\frac{1 + T_d s}{1 + T_f s} \tag{8}$$

Next, we use the transfer function in eq. (2) to get the open-loop transfer function $H_0 = H_{pd} \cdot H_{ship}$. We want the phase margin of the open-loop system to be 50° and a crossover frequency ω_c of about 0.10. In order to have the derivative time constant T_d cancel the transfer function time constant T, we set $T_d = T = 72.4666$.

$$H_0(s) = K_{pd}\frac{(1 + T_d s)}{(1 + T_f s)} \cdot \frac{K}{s(1 + Ts)} = \frac{K_{pd}K}{s(1 + T_f s)} \tag{9}$$

$$H_0(j\omega_c) = H_0(s)\Big|_{s=j\omega_c} = K_{pd}K \cdot \frac{-T_f\omega_c^2 - j\omega_c}{T_f\omega_c^4 + \omega_c^2} \tag{10}$$

The angle of the transfer function given in eq. (10) with a desirable phase margin $\varphi = 50°$, is given by the following equation:

$$\angle H_0(j\omega_c) = \arctan\left(\frac{-\omega_c}{-T_f\omega_c^2}\right) = -(180° - 50°)$$

Inserting the given crossover frequency $\omega_c = 0.10$ and solving with regards to T_f, leads to $T_f = 8.39$. To find the controller gain K_{pd}, the absolute value of the transfer function in eq. (10) is set equal to 1. This gives the following expression:

$$K_{pd} = \frac{\omega_c}{K}\sqrt{1 + T_f\omega_c^2}$$

Inserting the value for K found in section 1.2, $\omega_c = 0.10$ and T_f found in the paragraph above, gives $K_{pd} = 0.8361$. The Bode plot of the transfer function given by eq. (9) is shown in fig. 8. Both the phase margin and the crossover frequency is as desired, respectively $\varphi = 50°$ and $\omega_c = 0.10$. This is a vast improvement over the phase margin of 16.9° we have in the uncontrolled system, see fig. 1. Because the phase in fig. 8 never crosses the $-180°$ line, the system will in theory never be unstable with this PD-controller.

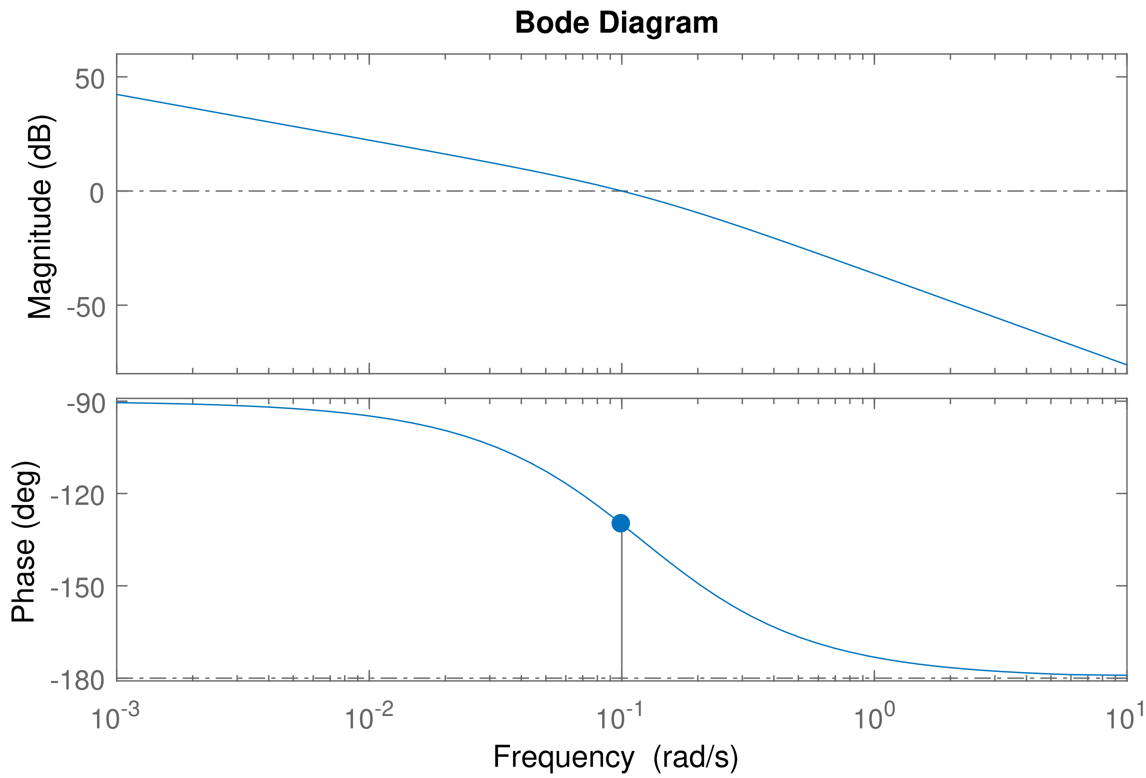

Figure 8: Bode diagram of the open loop system with a PD-controller, inserted the time constant $T_f = 8.39$ and the controller gain $K_{pd} = 0.8361$.

3.2 Problem 3 b)

The Simulink implementation of the PD-controller is shown in fig. 21, and a plot of the simulation without disturbances can be seen in fig. 9. The compass course reaches the reference value without any overshoot. Thus the autopilot is working, and the restriction of keeping the compass course $\psi \in \langle -35°, 35° \rangle$ is met.

A P-controller is in general slower than a PD-controller, when it's assumed that they are both tuned to give a critically damped response. Furthermore, a P-controller might give an overshoot in the compass course with more oscillation in rudder input. The rudder could therefore experience more wear and tear than with a PD-controller.

We see that there is a dip in the compass course at about 20°. This is caused by what the derivative term in the controller does to the rudder input. The derivative term sees the compass course approaching the reference too fast and compensates for this by starting to turn the rudder in the opposite direction. This causes the ship to start turning the other way. The P-term of the controller then corrects this and steers the ship to the reference value, which is reached after approximately 250 seconds. As the ship's moment of inertia is unknown, it's hard to tell whether this response is fast or not. A smaller phase margin would make the response more aggressive, but the system would be less robust. The jitter in the rudder input is because of measurement noise in compass course.

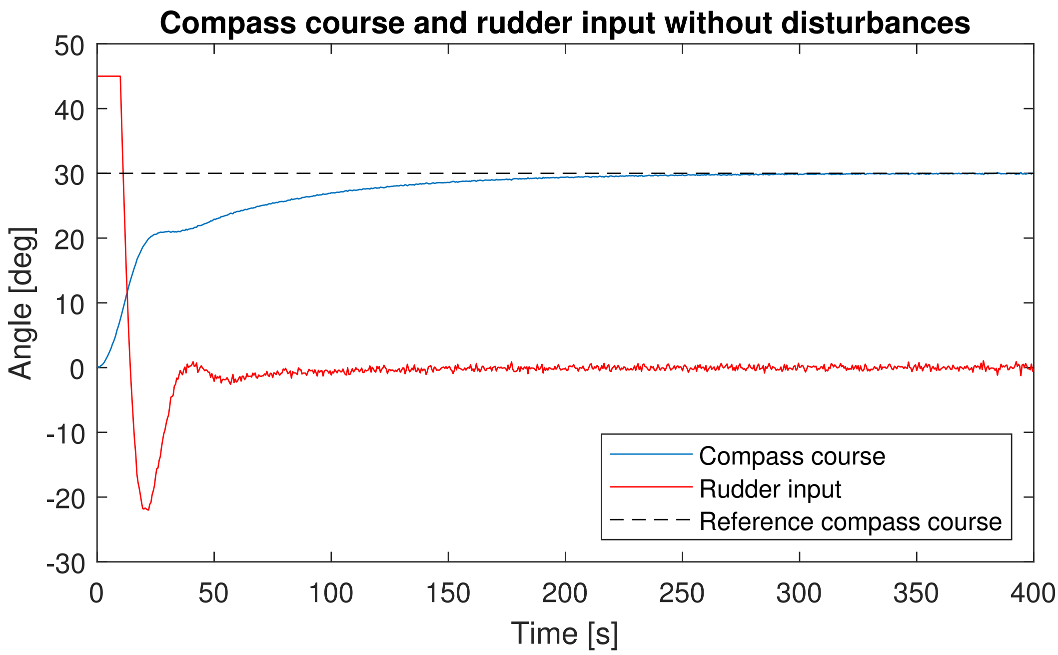

Figure 9: Plot of the compass course and the rudder angle, with a PD controller for the compass course ψ. The compass course reference angle is 30°, and no disturbances are affecting the system.

3.3 Problem 3 c)

A plot of the simulation with current disturbance and measurement noise can be seen in fig. 10. The response is approximately as fast as in fig. 9, but we do not reach the desired reference value. As expected from a PD-controller with constant disturbance applied, we observe a stationary deviation from the reference value, in this case 3°. Thus the autopilot is not working as desired.

Adding integral effect to the controller would fix this problem. Doing so, will push the phase in fig. 8 below $-180°$ at some finite frequency, which in turn can make the system unstable. This instability will depend on the controller gain, i.e. keeping the gain at a low enough value will keep the system stable, whereas a too high gain will lead to $\omega_c > \omega_{-180°}$ and instability. The crossover frequency ω_c is defined as the frequency where the amplitude of H_0 is 0 dB, and $\omega_{-180°}$ is the frequency where the phase crosses $-180°$.

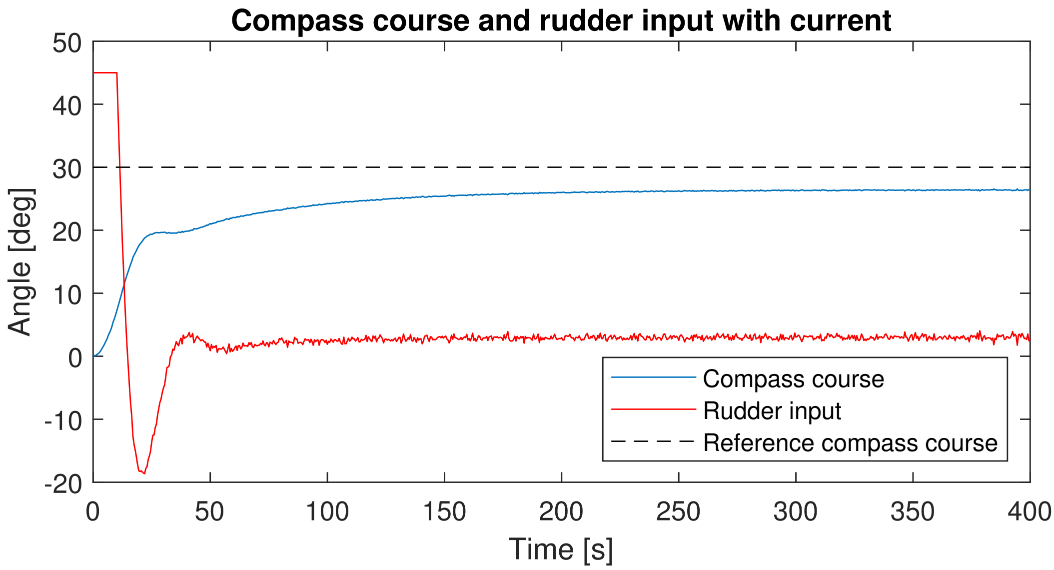

Figure 10: Plot of the compass course and the rudder angle, with a PD controller for the compass course ψ. The compass course reference angle is 30°, and the model is simulated with a current disturbance.

3.4 Problem 3 d)

A plot of the simulation with measurement noise and wave disturbance can be seen in fig. 11. The compass course reaches the reference value without any stationary deviation but oscillates around the reference value with an amplitude of about 2°. This is due to the wave disturbance's affection on the compass course, i.e. ψ_w. The PD-controller gives a desired response as expected. The derivative effect of the controller withstands the dynamic disturbance of the waves good, as in contrast of the stationary current.

Even though the compass course is relatively stable around the reference value, we can not say that the system is working satisfactory. The rudder input fluctuates between $-15°$ and $15°$ in a matter of seconds. This is unwanted with regards to wear and tear. A solution to this could be to low-pass filter the rudder input. This would lead to a more satisfactory rudder input, at the cost of a slower compass course response.

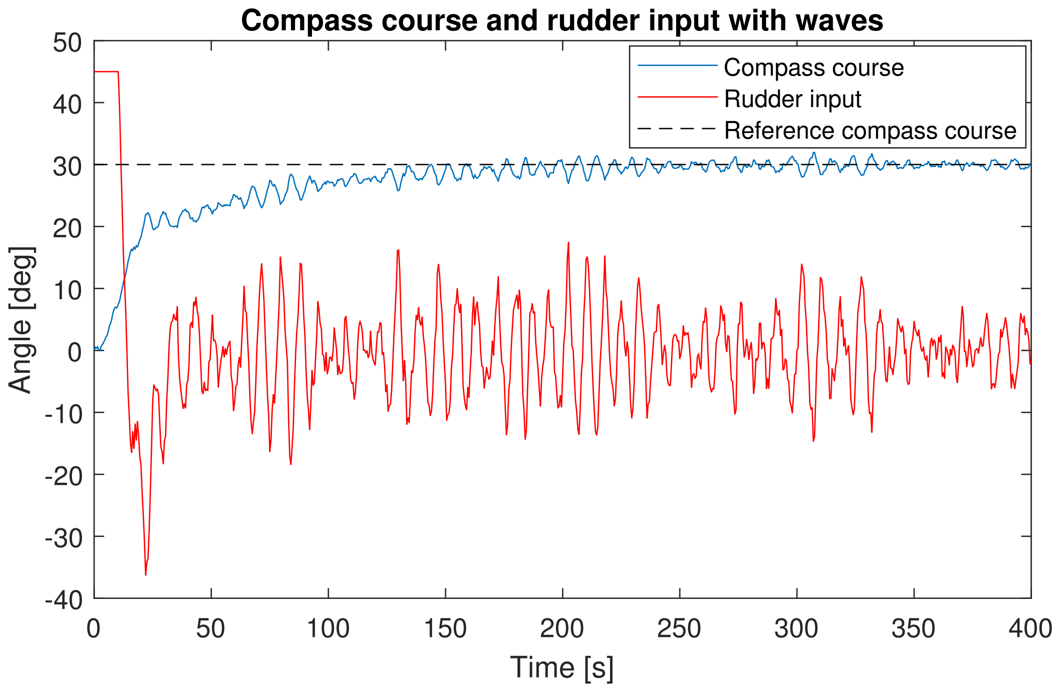

Figure 11: Plot of the compass course and the rudder angle, with a PD controller for the compass course ψ. The compass course reference angle is 30°, and the model is simulated with a wave disturbance.

4 Assignment 5.4 Observability

4.1 Problem 4 a)

In this part of the assignment, the observability matrix is calculated for different systems. The systems are different due to applying different disturbances. If a system is observable, it is possible to trace each state back to its initial condition. The general state-space form is given by:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u + \mathbf{E}\mathbf{w}, \quad y = \mathbf{C}\mathbf{x} + v \tag{11}$$

The matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{E}$ for the system given by eqs. (1a)–(1f), when $\mathbf{x}$ is defined as $\mathbf{x} = \begin{bmatrix}\xi_w & \psi_w & \psi & r & b\end{bmatrix}^T$ and $\mathbf{w} = \begin{bmatrix}w_w & w_b\end{bmatrix}^T$, are:

$$\begin{gathered}
\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\omega_0^2 & -2\lambda\omega_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1/T & -K/T \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ K/T \\ 0 \end{bmatrix} \\
\mathbf{E} = \begin{bmatrix} 0 & 0 \\ K_w & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \end{bmatrix}
\end{gathered} \tag{12}$$

4.2 Problem 4 b)

First the system is modelled without disturbances. This means that the system is only governed by the average heading ψ and its derivative r. Thus, the state vector is defined as $\mathbf{x} = \begin{bmatrix}\psi & r\end{bmatrix}^T$, and the matrices in eq. (11) are found to be:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -1/T \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ K/T \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad \mathbf{E} = \mathbf{0} \tag{13}$$

Using the MATLAB function `obsv(A,C)` to compute the observability matrix for the system given above, yields:

$$\mathcal{O}_\mathbf{1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The observability matrix has full rank, thus the system is observable.

4.3 Problem 4 c)

Next, the system is modelled with only the current disturbance. In this case, the states of the system are the average heading ψ, the derivative of the average heading r, and the bias to the rudder angle b. This bias is caused by the current affecting the rudder. The state vector is defined as $\mathbf{x} = \begin{bmatrix}\psi & r & b\end{bmatrix}^T$ and the matrices are given below:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1/T & -K/T \\ 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ K/T \\ 0 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \qquad \mathbf{E} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{14}$$

Using the same approach as in section 4.2, the following observability matrix is computed:

$$\mathcal{O}_\mathbf{2} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -0.0138 & -0.0022 \end{bmatrix}$$

The observability matrix has full rank, thus the system is observable.

4.4 Problem 4 d)

When the system is modelled with only wave as a disturbance, the states are the following: the average heading ψ, the derivative of the average heading r, the high-frequency heading component ψ_w due to the wave disturbance, and the accumulated high-frequency heading component ξ_w. The state vector is defined as $\mathbf{x} = \begin{bmatrix} \psi & r & \psi_w & \xi_w \end{bmatrix}^T$, and the matrices are given below:

$$\begin{gathered} \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -1/T & 0 & 0 \\ 0 & 0 & -2\lambda\omega_0 & -{\omega_0}^2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ K/T \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \qquad \mathbf{E} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ K_w & 0 \\ 0 & 0 \end{bmatrix} \end{gathered} \tag{15}$$

Using the same approach as in section 4.2, the corresponding observability matrix to this system is computed:

$$\mathcal{O}_\mathbf{3} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & -0.1408 & -0.6120 \\ 0 & -0.0138 & -0.5922 & 0.0862 \\ 0 & 0.0002 & 0.1696 & 0.3625 \end{bmatrix}$$

The observability matrix has full rank, thus the system is observable.

4.5 Problem 4 e)

Finally, the system is modelled with both wave and current disturbances. Using the matrices in eq. (12), the corresponding observability matrix is computed to be:

$$\mathcal{O}_\mathbf{4} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ -0.6120 & -0.1408 & 0 & 1 & 0 \\ 0.0862 & -0.5922 & 0 & -0.0138 & -0.0022 \\ 0.3625 & 0.1691 & 0 & 0.0002 & 0 \\ -0.1038 & 0.3386 & 0 & 0 & 0 \end{bmatrix}$$

The observability matrix has full rank, thus the system is observable.

We see that some of the numerical values in the observability matrices in sections 4.3–4.5 are close to zero. This implies that the system may be close to unobservable in practice, although this is not a problem as the compass course is kept within $\pm 35°$. As the system modelled with both wave and current disturbance is observable, we can utilize an observer to further improve the system response.

5 Assignment 5.5 Discrete Kalman filter

5.1 Problem 5 a)

In this part of the assignment, the goal is to discretize the matrices in eq. (12) to the general multivariable discretized form:

$$\mathbf{x}_k = \mathbf{A}_d \mathbf{x}_{k-1} + \mathbf{B}_d \mathbf{u}_{k-1} + \mathbf{E}_d \mathbf{w}_{k-1} \tag{16a}$$
$$\mathbf{y}_k = \mathbf{C}_d \mathbf{x}_k + \mathbf{v}_k \tag{16b}$$

where the discrete matrices are defined as:

$$\mathbf{A}_d = \exp(\mathbf{A}T) \qquad \mathbf{B}_d = \int_0^{T_s} \exp(\mathbf{A}\alpha)\mathbf{B}\,\mathrm{d}\alpha \qquad \mathbf{C}_d = \mathbf{C} \qquad \mathbf{E}_d = \int_0^{T_s} \exp(\mathbf{A}\beta)\mathbf{E}\,\mathrm{d}\beta \tag{17}$$

In order to derive the expressions in eq. (17), a few assumptions were made. We assume zero-order hold on the input $\mathbf{u}$, i.e. the input is considered to be constant during the sample period T_s. The same goes for the disturbances to calculate the disturbance matrix $\mathbf{E_d}$. The matrix $\mathbf{C_d}$ contains the noiseless correlation between the state and measurement vectors.

Listing 2 shows the MATLAB code implementing the exact discretization used to find the matrices in eq. (17). The function `ss(A,B,C,D)` returns the system on a state-space form. The function `c2d(sys, Ts, 'zoh')` returns the discretized matrices given by eq. (17), given zero-order hold (zoh) on the input.

Listing 2: Exact discretization implemented in MATLAB.

```
f_s = 10;        %Sample frequency [Hz]
T_s = 1/f_s;     %Sample time

sys = ss(A,B,C,0);                 %system
sys_disturb = ss(A,E,C,0);         %system w/disturbance as control input

sys_disc = c2d(sys,T_s, 'zoh'); %discrete system

%discrete distubance system
sys_disturb_disc = c2d(sys_disturb,T_s, 'zoh');

A_d = sys_disc.A;
B_d = sys_disc.B;
C_d = C;
E_d = sys_disturb_disc.B;
```

5.2 Problem 5 b)

To estimate the variance of the measurement noise, we set the rudder angle δ to 0 and measure the compass course. This isolates the measurement noise and we get the correct variance.
The MATLAB function `var()` was used to find the variance of the signal, $\mathbf{R}$, resulting in:

$$\mathbf{R} = 6.132 \cdot 10^{-7}\ rad^2 \tag{18}$$

5.3 Problem 5 c)

The conceptual idea of the Kalman filter is to estimate what a dynamic process will do in the next time instance, when the states are undeterministic. The states are stochastic because the system is influenced by random disturbances. The measurement y is also stochastic, because it's affected by white measurement noise v. In this assignment, the discrete Kalman filter was utilized. This implies that the measurement y is taken in discrete time intervals, and the estimate $\hat{\mathbf{x}}_k$ is computed likewise. Futhermore, it's assumed that the input is deterministic and constant between two time instances t and $t + T_s$. The following matrices were given in the assignment:

$$\mathbf{w} = \begin{bmatrix} w_w & w_b \end{bmatrix}^T \qquad \mathbf{Q} = E\{\mathbf{w}\mathbf{w}^T\} = \begin{bmatrix} 30 & 0 \\ 0 & 10^{-6} \end{bmatrix}$$

$$\mathbf{P}_0^- = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.013 & 0 & 0 & 0 \\ 0 & 0 & \pi^2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2.5 \cdot 10^{-3} \end{bmatrix} \qquad \hat{\mathbf{x}}_0^- = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{19}$$

The matrix $\mathbf{P}_0^-$ contains the initial a priori state estimation error covariances. As only the diagonal elements are non-zero, all the covariances in the system are considered to be zero, while the variances are given by the diagonal. $\mathbf{Q}$ is the process noise covariance matrix and is constant throughout sections 5.3, 5.4 and 5.5. $\hat{\mathbf{x}}_0^-$ is the initial a priori state estimate.

At time instance kT_s, $k \in \mathbb{N}_0$, the state vector is denoted $\mathbf{x}_k$ and is unknown due to stochastic disturbances in the system. The estimate of the state before the measurement is incorporated is denoted $\hat{\mathbf{x}}_k^-$, and the updated state estimate is denoted $\hat{\mathbf{x}}_k$. Furthermore, the estimation error is defined as

$$\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k \tag{20}$$

The equations given below are implemented in MATLAB to realize the discrete Kalman filter [1, ch. 4.2]. The MATLAB code of the implementation is seen in listing 10.

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{C}_d^T (\mathbf{C}_d \mathbf{P}_k^- \mathbf{C}_d^T + \mathbf{R}_v)^{-1} \tag{21a}$$
$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{C}_d \hat{\mathbf{x}}_k^-) \tag{21b}$$
$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{C}_d) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{C}_d)^T + \mathbf{K}_k \mathbf{R}_v \mathbf{K}_k^T \tag{21c}$$
$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{A}_d \hat{\mathbf{x}}_k + \mathbf{B}_d \mathbf{u}_k \tag{21d}$$
$$\mathbf{P}_{k+1}^- = \mathbf{A}_d \mathbf{P}_k \mathbf{A}_d^T + \mathbf{E}_d \mathbf{Q} \mathbf{E}_d^T \tag{21e}$$

In eq. (21a) the Kalman gain $\mathbf{K}_k$ is computed. This blending factor minimizes the expected value of $||\mathbf{e}_k||^2$, where the estimation error $\mathbf{e}_k$ is defined in eq. (20).

Equations (21b) and (21c) update the estimate with measurement and the error covariance matrix, respectively. A simple explanation of eq. (21b) is: if the measurement is equal to the estimated measurement, $\hat{\mathbf{x}}_\mathbf{k}$ is set to the model estimate. If there is a deviation between the measurement and the model prediction, the gain matrix $\mathbf{K}_k$ is used to combine the prediction and the measurement in an optimal way.

Equations (21d) and (21e) predict the a priori state estimation and estimation error covariance for the next time step. $\mathbf{R}_v$ is the measurement noise variance given by $\mathbf{R}/T_s$, where $\mathbf{R}$ is defined in eq. (18) and T_s is the sample time. $\mathbf{y}_k$ is defined in eq. (16b).

5.4 Problem 5 d)

From fig. 12 we see that the feed forward of the rudder bias eliminates the stationary deviation seen in fig. 10. This gives us the desired response, and a much better result than in section 3.3. As before, it meets the requirement of keeping ψ within the range of $\pm 35°$.

To eliminate stationary deviations we have two options: feed forward or integral effect in the controller. From classical control theory we know that feed forward does not influence the stability of the closed loop system, as opposed to integral effect. This was discussed in detail in section 3.3. This is desirable, and a good reason to choose feed forward instead of a PID controller.

Zooming in on fig. 12 it can be seen that the estimated rudder bias stabilizes at approximately 3°. This is the reason why feed forward with the bias removes the stationary deviation, it cancels the constant current disturbance pushing on the rudder. When the compass course reaches its reference, the error in the feedback loop becomes zero and $\delta = b$, see fig. 23. This is also recognized in fig. 12.

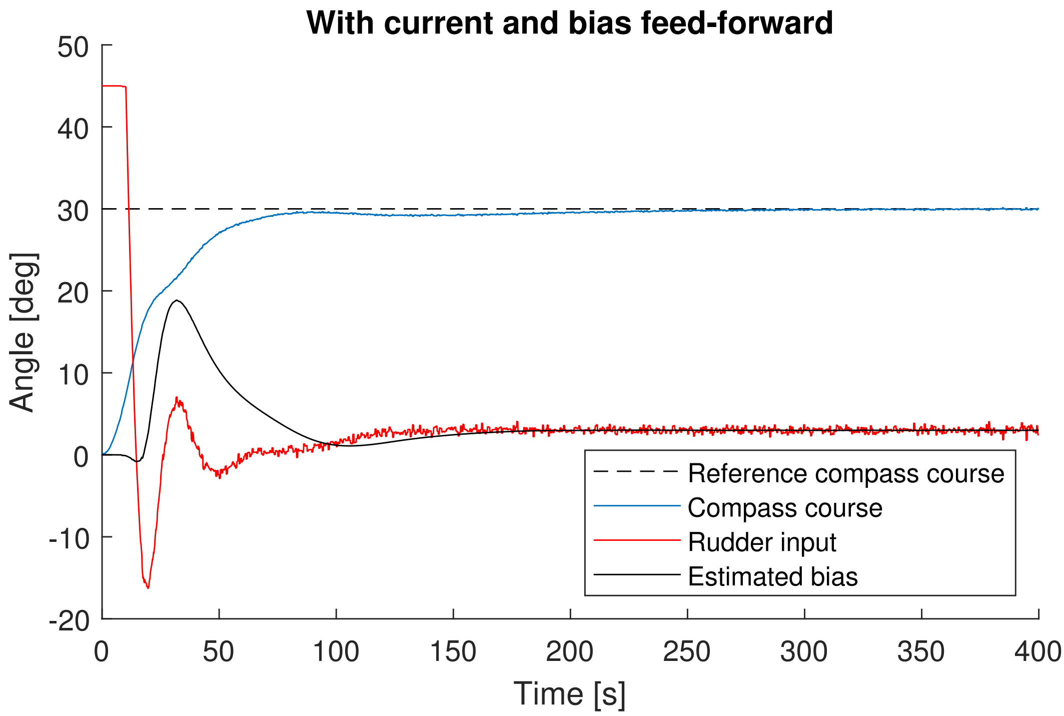

Figure 12: Compass course ψ plotted with the rudder input δ and the estimated bias from the Kalman filter. The system is simulated with a current disturbance, and the reference compass heading ψ_r is 30°.

5.5 Problem 5 e)

In this part, the wave filtered compass course ψ provided by the Kalman filter is used in the autopilot. In addition, the model is simulated with both wave and current disturbances. Figure 13 shows the actual compass course plotted with the filtered compass course. Figure 14 shows the estimated bias and rudder input, and fig. 15 shows the estimated and actual wave influence.

When comparing figs. 11 and 13, there's a significant improvement in the ship's compass course response, particularly with regards to the time it takes to reach the reference angle. In fig. 11

the measured compass course reaches 30° after about 200 seconds, while in fig. 13 the reference compass angle of 30° is reached after about 50 seconds. It is worth noticing that the filtered compass course is used in the feedback loop. This faster response brings along slightly more oscillation, as seen clearly in the filtered compass course in fig. 13. Considering the model restriction of keeping the compass course $\psi \in \langle -35°, 35° \rangle$, this holds in both fig. 11 and fig. 13.

Comparing the rudder input in fig. 11 with fig. 14, the rudder input in the latter is more desirable. In both cases the rudder input is oscillatory, but in fig. 11 it's much more exposed to wear and tear due to the high-frequency components of the input. This is the result of the rudder input trying to compensate for the waves by turning the rudder. The use of the filtered compass course in the feedback loop explains the low-frequency oscillation in the rudder input in fig. 14. In order to compensate for the current disturbance, the estimated bias and the rudder input stabilizes at about 3° in fig. 14.

The actual and estimated wave influence is shown in fig. 15. To estimate the filtered compass course ψ, the Kalman filter uses the estimated wave influence of the ship ψ_w. This is one of the direct reasons for the better response in fig. 13 compared to section 3.4. Before 100 seconds has passed, the system has not yet reached the reference compass course. We observe that the wave influence is estimated poorly before 100 seconds, this may be because of difficulties determining if it's the controller or the disturbance that is contributing to the change in ψ. To get a good estimate from the start, we could disconnect it from the controller. This way the transient would not affect the estimate, but the estimate shown in fig. 15 is the one which was used in the feedback loop.

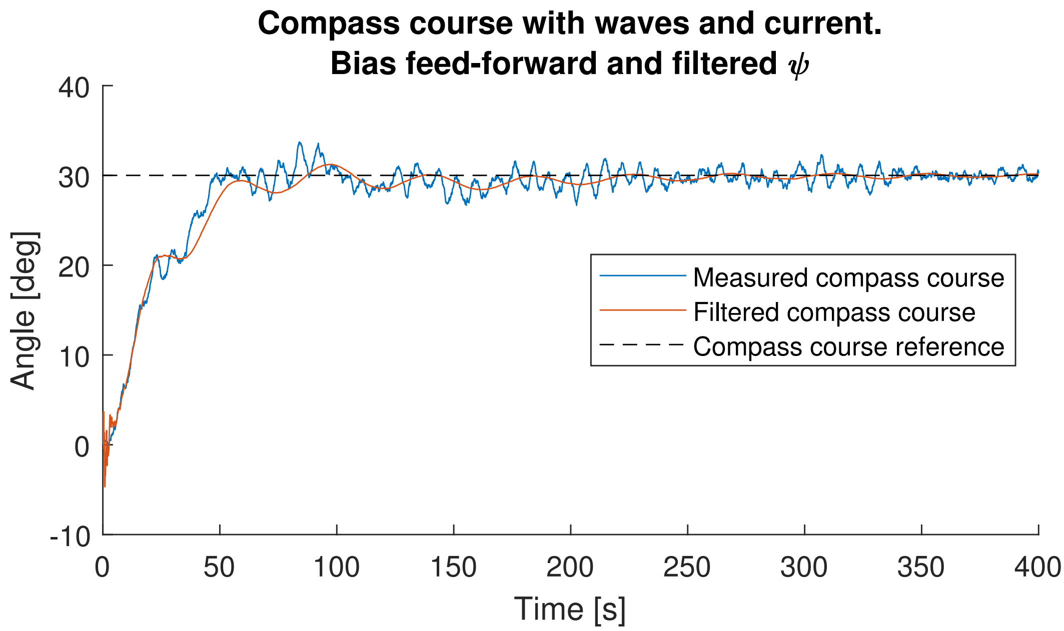

Figure 13: Plot of the measured and the filtered compass course ψ, when the system is simulated with wave and current disturbances. The estimated ψ is provided by the Kalman filter and used in the feedback loop.

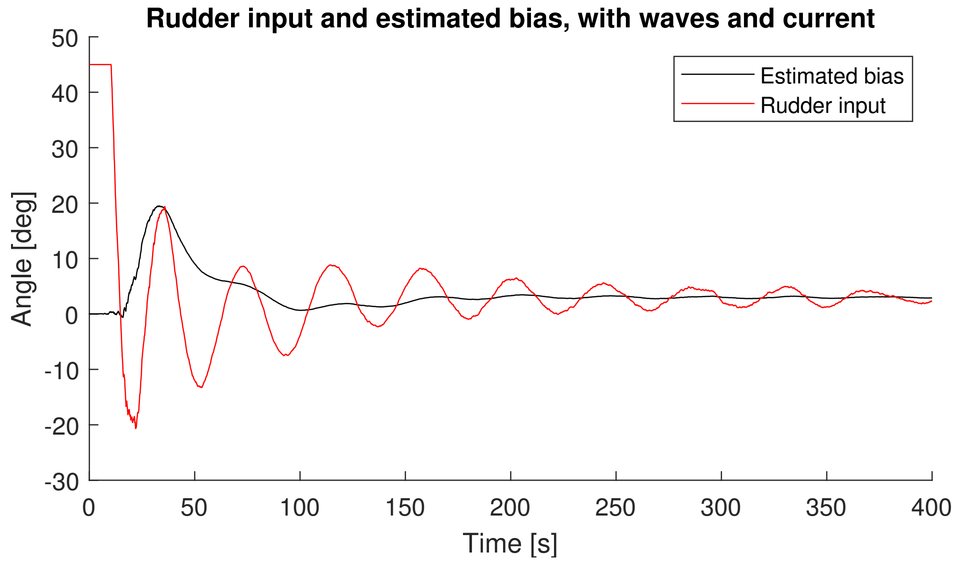

Figure 14: The estimated bias, caused by the current and wave disturbances, plotted with the rudder input δ.

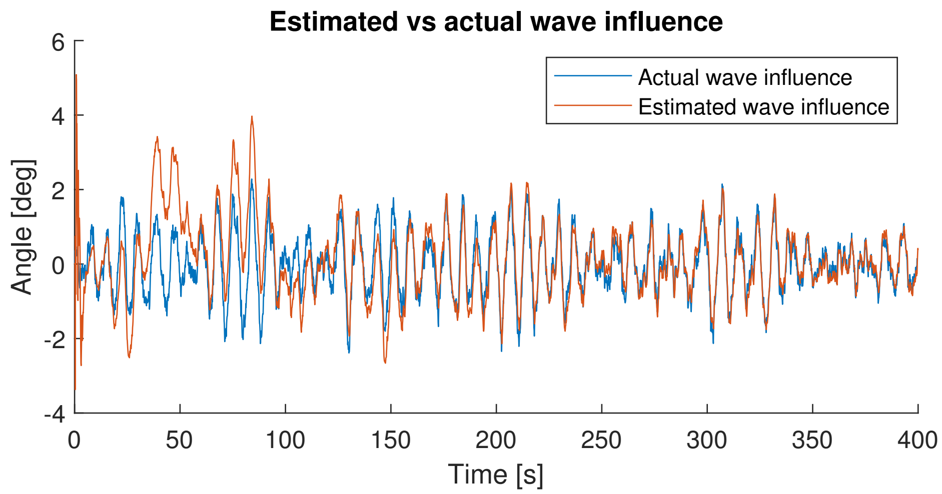

Figure 15: The estimated wave influence on the ship, provided by the Kalman filter, plotted against the actual wave influence.

5.6 Problem 5 f)

In this part, the goal is to observe how changes in the $\mathbf{Q}$ matrix affect the responses in sections 5.4 and 5.5. The elements $\mathbf{Q_{11}}$ and $\mathbf{Q_{22}}$ are the process noise variances. In practice they can be explained as how certain we are of the attributes, e.g. amplitude, of the disturbances affecting the system. A low value implies that the model is considered to be nearly perfect with regards to the corresponding disturbance, thus the Kalman filter relies heavily on the model. A high value implies that the Kalman filter relies more on the measurement, rather than the model, of the disturbance. In our case, the current is considered to be easier to model than the wave disturbance, i.e. it's more uncertain how the waves will affect the ship, as opposed to the current.

The elements $\mathbf{Q_{12}} = \mathbf{Q_{21}} = E\{w_w w_b\}$ denote the covariance between the wave and current disturbance. Since these elements are zero, the model assumes that there is no correlation between the wave and the current. This is hardly the case in real life, as you would expect that if the current affects the rudder from one direction, the waves would more or less hit the ship from the same direction. The current would affect the ship itself, and not just the rudder bias.

In the plot in fig. 12, the compass course is plotted with the rudder input and the estimated bias. The latter is provided by the Kalman filter, and is a result of a simulation with current disturbance. Changing $\mathbf{Q_{22}}$ from 10^{-6} to 10^{-3}, while keeping $\mathbf{Q_{11}}$ unchanged at 30, gives the response seen in fig. 16. The resulting response of the increased $\mathbf{Q_{22}}$ shows a more oscillatory rudder input, see fig. 12 compared to fig. 16. We also know that an increase in $\mathbf{Q_{22}}$ will lead to a higher value in the Kalman gain, which makes the Kalman filter trust the measurements more than the model. An increase in an element in the $\mathbf{Q}$ matrix seems to make the corresponding estimate more oscillatory. This leads to our hypothesis that the observer becomes slower as the Kalman gain increases, because a slower observer can result in a more oscillatory estimate.

Furthermore, when comparing the plots of the compass course, we see a significant difference in responses. These plots are seen in figs. 13 and 14 as well as fig. 17, respectively. In fig. 13 the filtered compass course is more oscillatory than in fig. 17. This supports our hypothesis, as $\mathbf{Q_{11}}$ is higher in fig. 13 than in fig. 17. Again, a decrease in $\mathbf{Q}$ leads to less oscillations which may be because of the estimator getting faster.

As seen in fig. 18, a decrease of $\mathbf{Q_{11}}$ from 30 to 0.03 while keeping $\mathbf{Q_{22}}$ unchanged at 10^{-6}, gives a estimated wave influence lower than the actual wave influence. This is a result of the Kalman filter relying too much on the model of the wave influence, i.e. it will behave more like a damped harmonic oscillator, which in turn leads to a poor estimation. This confirms the discussion in the first paragraph, the wave influence on the compass course is difficult to model properly. The most noticeable difference of the decrease in $\mathbf{Q_{11}}$ is the rudder input, which oscillates with a high frequency and an amplitude of 3° to 5°. This oscillation will affect the wear and tear of the rudder more than the rudder input in fig. 14.

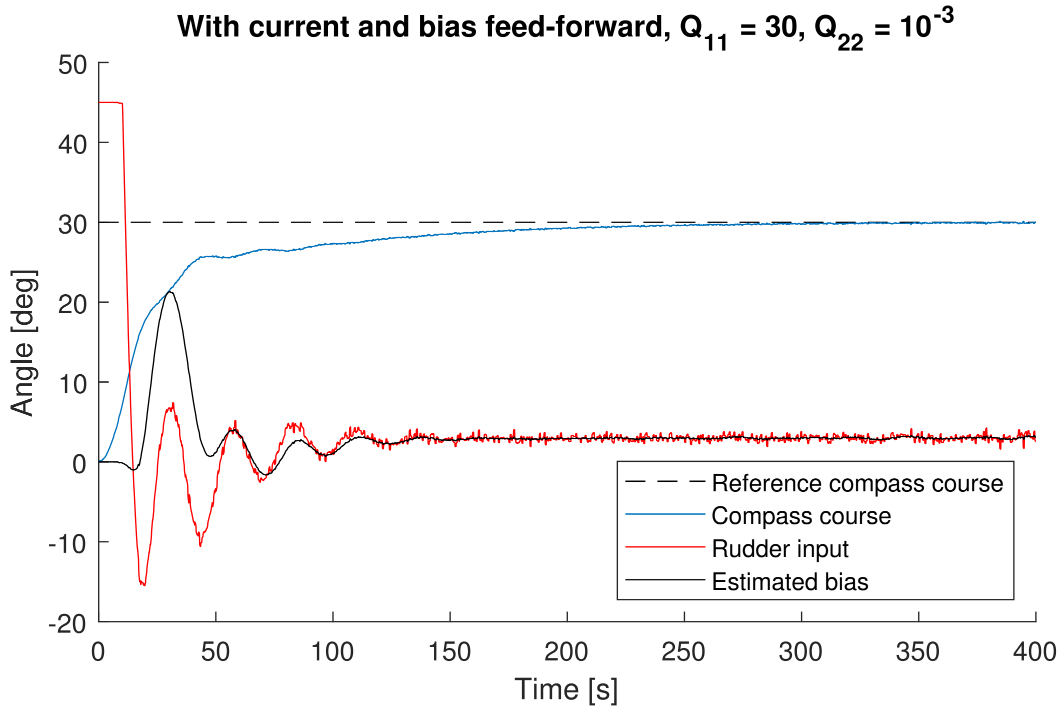

Figure 16: Plot of measured compass course, rudder input δ and the estimated bias provided by the Kalman filter. The system was simulated with current disturbance, with $\mathbf{Q}_{22} = 10^{-3}$ while keeping $\mathbf{Q}_{11}$ unchanged at 30.

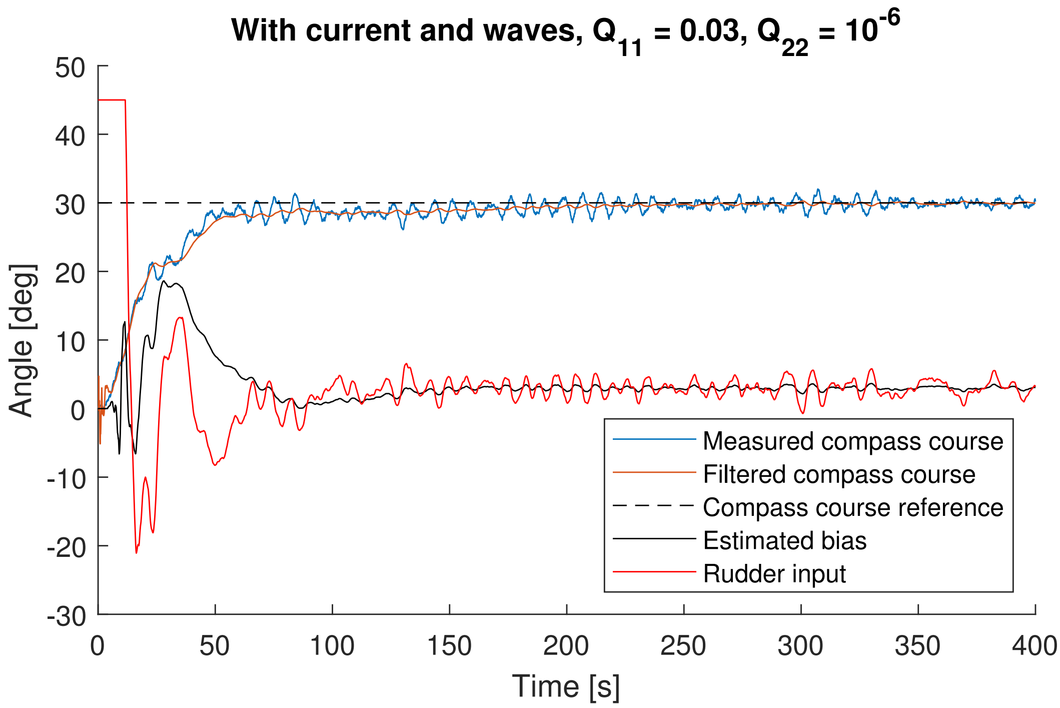

Figure 17: Plot of measured compass course, estimated bias and the rudder input δ. The system was simulated with wave and current disturbances, and the wave filtered compass course estimated by the Kalman filter was used in the autopilot. The element $\mathbf{Q}_{11}$ was set to 0.03 while $\mathbf{Q}_{22}$ remained unchanged at 10^{-6}.

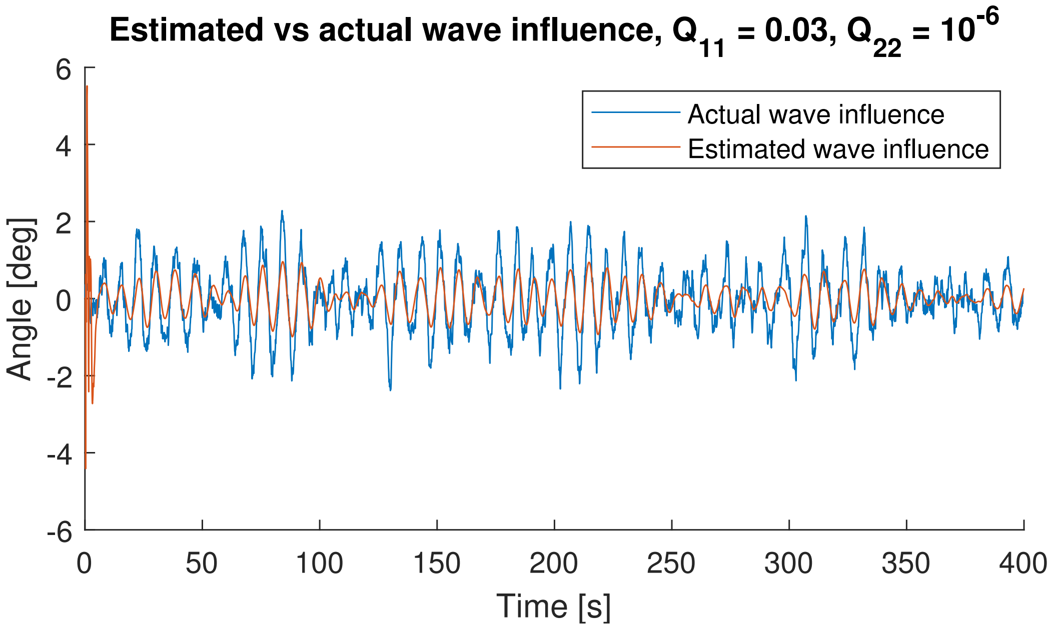

Figure 18: Plot of actual and estimated wave influence. The system was simulated with wave and current disturbances, with $\mathbf{Q}_{11} = 0.03$ and $\mathbf{Q}_{22} = 10^{-6}$.

A Matlab code

Listing 3: The MATLAB code in part 1b) Identification of the boat parameters

```matlab
clear all
sine_freq = 0.05; %set frequency of sine input
sim('p5p1b_sim.mdl')
figure()
plot(ship.time, ship.signals.values);
title(['Frequency = ', num2str(sine_freq) , ...
    ' [rad/s] without distubances']);
ylabel('Compass measurement [deg]');
xlabel('Time [s]');

sine_freq = 0.005; %set frequency of sine input
sim('p5p1b_sim.mdl')
figure()
plot(ship.time, ship.signals.values);
title(['Frequency = ', num2str(sine_freq) , ...
    ' [rad/s] without distubances']);
ylabel('Compass measurement [deg]');
xlabel('Time [s]');

K = 0.15613;
T = 72.4666;

save('common files\task_51', 'K', 'T')
```

Listing 4: The MATLAB code in part 1c) Identification of the boat parameters

```
clear all
sine_freq = 0.05; %set frequency of sine input
sim('p5p1c_sim.mdl')
figure()
plot(ship.time, ship.signals.values);
title(['Frequency = ', num2str(sine_freq) , ...
    ' [rad/s] with waves and measurement noise']);
ylabel('Compass measurement [deg]');
xlabel('Time [s]');

sine_freq = 0.005; %set frequency of sine input
sim('p5p1c_sim.mdl')
figure()
plot(ship.time, ship.signals.values);
title(['Frequency = ', num2str(sine_freq) , ...
    ' [rad/s] with waves and measurement noise']);
ylabel('Compass measurement [deg]');
xlabel('Time [s]');
```

Listing 5: The MATLAB code in part 1d) Identification of the boat parameters

```
clear all
load('task_51.mat')
sim('p5p1d_sim.mdl')
figure()
hold on
plot(ship.time, ship.signals.values);
plot(model.time, model.signals.values);
title('Step response model vs ship');
ylabel('Compass measurement [deg]');
xlabel('Time [s]');
legend({'Ship','Model'},'Location','Southeast')
```

Listing 6: The MATLAB code in part 2 Identification of wave spectrum model

```
load('wave.mat')
%% calculation
fs = 10;                    %Sampling frequency
window = 4096;              %Window size
x = psi_w(2,:) .* (pi/180); %Extract values and convert to rad

[pxx,f] = pwelch(x,window, [], [],fs); %Estimate PSD based on data
w = f .* (2*pi);            %Convert to rad/s
pxx = pxx ./ (2*pi);        %Convert to s/rad

figure()
plot(w, pxx);
title('Estimated Power Spectral Density');
xlabel('Frequency [rad/s]');
ylabel('PSD intensity');
xlim([0 1.6]);

%% find w_0 and sigma^2
[sigma2,max_index] = max(pxx);
w0 = w(max_index);

%% find lambda
figure()
p1 = plot(w, pxx, 'black');
hold on
p2 = plot(w, func(0.3, w, pxx),'b');
p3 = plot(w, func(0.01, w, pxx),'Color',[31/255 155/255 3/255]);
p4 = plot(w, func(0.09, w, pxx),'r');

set(p1,'LineWidth',2);
set([p2 p3 p4],'LineWidth',1.1);

xlabel('Frequency [rad/s]');
ylabel('PSD intensity');
legend('S_{\psi_w}','\lambda = 0.3','\lambda = 0.01','\lambda = 0.09')
xlim([0, 2]);

%% save variables
save('common files\task_52','fs','psi_w','sigma2','w0')

%% function definition
function P = func(lambda, Wx, Sx)
    [sigma2,max_index] = max(Sx);
    w0 = Wx(max_index);

    Kw = 2*lambda*w0*sqrt(sigma2);
    P = ((Wx.^2 * Kw^2)./((w0^2-Wx.^2).^2 + 4*lambda^2 * w0^2 * Wx.^2));
end
```

Listing 7: The MATLAB code in part 3b) Control system design

```
clear all
load('task_51.mat')
psi_r = 30;
K_pd = 0.836;
T_d = T;
T_f = 8.39;

H_pd = tf(K_pd*[T_d 1],[T_f 1]);
H_ship = tf([K],[T 1 0]);
H_0 = H_pd*H_ship;

sim('p5p3b_sim.mdl')

plot(ship.time, ship.signals.values);
hold on
plot(rudder_angle.time, rudder_angle.signals.values,'r');
plot(ship.time,30*ones(size(ship.time)),'black--')

title('Compass course and rudder input without disturbances')
legend({'Compass course', 'Rudder input', ...
    'Reference compass course'}, 'Location','Southeast');
ylabel('Angle [deg]');
xlabel('Time [s]');

save('common files\task_53', 'psi_r', 'H_0', 'H_pd', ...
    'H_ship', 'K_pd', 'T_d', 'T_f')
```

Listing 8: The MATLAB code in part 3c) Control system design

```
clear all
load('task_53.mat')
sim('p5p3c_sim.mdl')

plot(ship.time, ship.signals.values);
hold on
plot(rudder_angle.time, rudder_angle.signals.values,'r');
plot(ship.time,30*ones(size(ship.time)),'black--')

title('Compass course and rudder input with current')
legend({'Compass course', 'Rudder input', ...
    'Reference compass course'}, 'Location','Southeast');
ylabel('Angle [deg]');
xlabel('Time [s]');
```

Listing 9: The MATLAB code in part 3d) Control system design

```
clear all
load('task_53.mat')
sim('p5p3d_sim.mdl')

plot(ship.time, ship.signals.values);
hold on
plot(rudder_angle.time, rudder_angle.signals.values,'r');
plot(ship.time,30*ones(size(ship.time)),'black--')

title('Compass course and rudder input with waves')
legend({'Compass course', 'Rudder input', ...
    'Reference compass course'}, 'Location','Southeast');
ylabel('Angle [deg]');
xlabel('Time [s]');
```

Listing 10: The MATLAB code for the implementation of Discrete Kalman Filter function

```
function [b, psi, psi_w] = kalman_filter_func(u, y)
    persistent init_flag A B C E Q R_v P_ x_ I

    if isempty(init_flag)
        init_flag = 1;
        %initialize the matrices
        matrices = load('task55.mat');
        A = matrices.A_d;
        B = matrices.B_d;
        C = matrices.C_d;
        E = matrices.E_d;
        Q = matrices.Q;
        R_v = matrices.R_v;
        P_ = matrices.P_;
        x_ = matrices.x_;
        I = matrices.I;
    end

    %update with measurement
    K = P_*C'*inv(C*P_*C' + R_v);           %Kalman gain
    x = x_ + K*(y-C*x_);                    %Linear observer
    P = (I-K*C)*P_*(I-K*C)' + K*R_v*K'; %Update error covariance

    %predict for next step
    x_ = A*x + B*u;
    P_ = A*P*A' + E*Q*E';

    b = x(5);
    psi = x(3);
    psi_w = x(2);
end
```

Listing 11: The MATLAB code in part 5c) Discrete Kalman Filter

```
clear all
load('task_53.mat')
load('task_54.mat')
f_s = 10;       %Sample frequency [Hz]
T_s = 1/f_s;    %Sample time

sys = ss(A,B,C,0);              %system
sys_disturb = ss(A,E,C,0);      %system w/disturbance as control input

sys_disc = c2d(sys,T_s, 'zoh'); %discrete system

%discrete distubance system
sys_disturb_disc = c2d(sys_disturb,T_s, 'zoh');

A_d = sys_disc.A;
B_d = sys_disc.B;
C_d = C;
E_d = sys_disturb_disc.B;

%% task 5.5 b)
sim('p5p5b_sim.mdl')

R = var(ship.signals.values)*(pi/180)^2; %unit: [rad^2]
R_v = R/T_s;

%% task 5.5 c)

Q = [30 0;
     0 10^-6];

P_ = [1 0 0 0 0;
      0 0.013 0 0 0;
      0 0 pi^2 0 0;
      0 0 0 1 0;
      0 0 0 0 0.0025];

x_ = [0; 0; 0; 0; 0];

I = eye(5);

save('common files\task_55','A_d','B_d','C_d','E_d', ...
    'Q','R_v','P_','x_','I', 'T_s')
```

Listing 12: The MATLAB code in part 5d) Discrete Kalman Filter

```
clear all
load('task_52.mat')
load('task_53.mat')
load('task_55.mat')
sim('p5p5d_sim.mdl')

figure()
hold on
plot(ship.time,30*ones(size(ship.time)),'black--')
plot(ship.time, ship.signals.values);
plot(rudder_angle.time, rudder_angle.signals.values,'r');
plot(est_bias.time, est_bias.signals.values, 'black');

title('With current and bias feed-forward')
legend('Reference compass course','Compass course', ...
    'Rudder input', 'Estimated bias', 'Location', 'Southeast');
ylabel('Angle [deg]');
xlabel('Time [s]');
```

Listing 13: The MATLAB code in part 5e) Discrete Kalman Filter

```
clear all
load('task_52.mat')
load('task_53.mat')
load('task_55.mat')
sim('p5p5e_sim.mdl')

figure()
hold on
plot(ship.time, ship.signals.values);
plot(est_psi.time, est_psi.signals.values);
plot(ship.time,30*ones(size(ship.time)),'black--')

title({'Compass course with waves and current.' ...
    'Bias feed-forward and filtered \psi'})
legend({'Measured compass course', 'Filtered compass course', ...
    'Compass course reference'}, 'Location','East');
ylabel('Angle [deg]');
xlabel('Time [s]');

figure()
hold on
plot(est_bias.time, est_bias.signals.values, 'black');
plot(rudder_angle.time, rudder_angle.signals.values,'r');

title('Rudder input and estimated bias, with waves and current')
legend('Estimated bias', 'Rudder input');
ylabel('Angle [deg]');
xlabel('Time [s]');

figure()
hold on
plot(psi_w(1,1:4000),psi_w(2,1:4000));
plot(est_psi_w.time, est_psi_w.signals.values);

title('Estimated vs actual wave influence')
legend('Actual wave influence','Estimated wave influence');
ylabel('Angle [deg]');
xlabel('Time [s]');
```

B Simulink diagrams

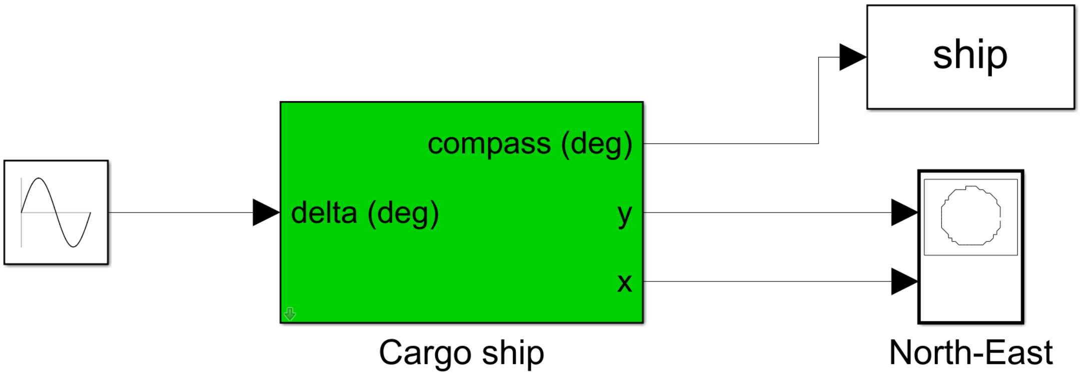

Figure 19: Simulink diagram used in problem 1 b) and c)

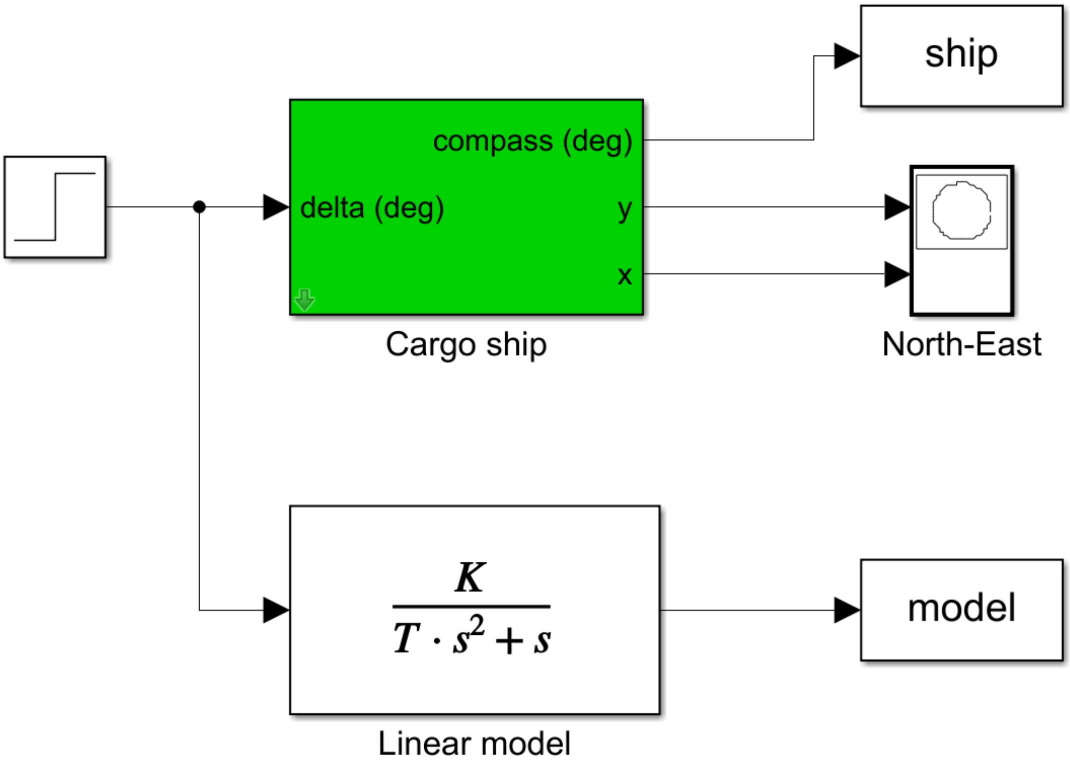

Figure 20: Simulink diagram used in problem 1 d)

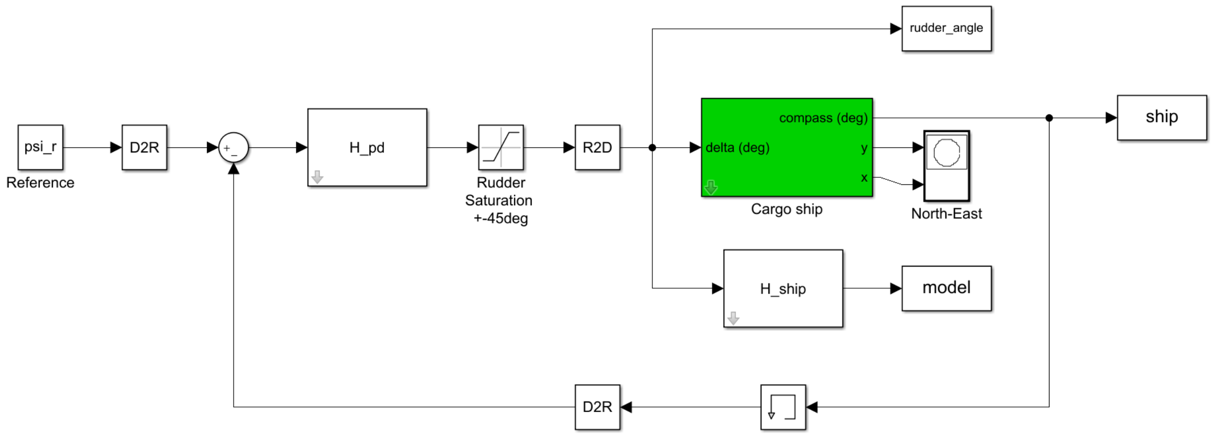

Figure 21: Simulink diagram used in problem 3

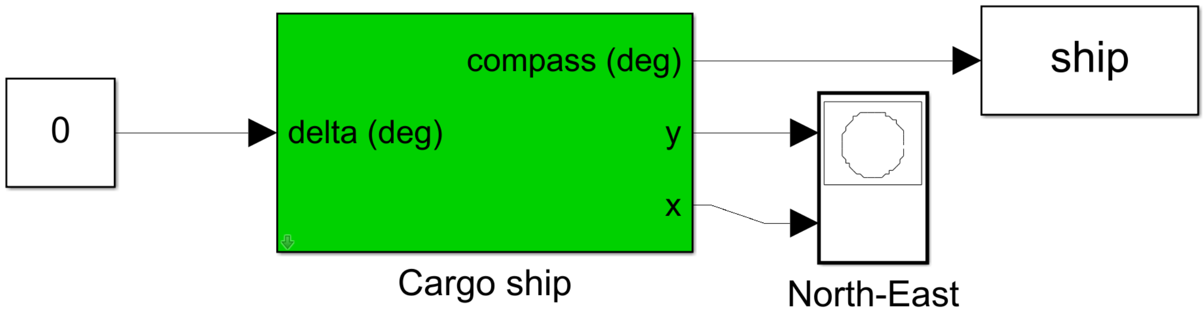

Figure 22: Simulink diagram used in problem 5 b)

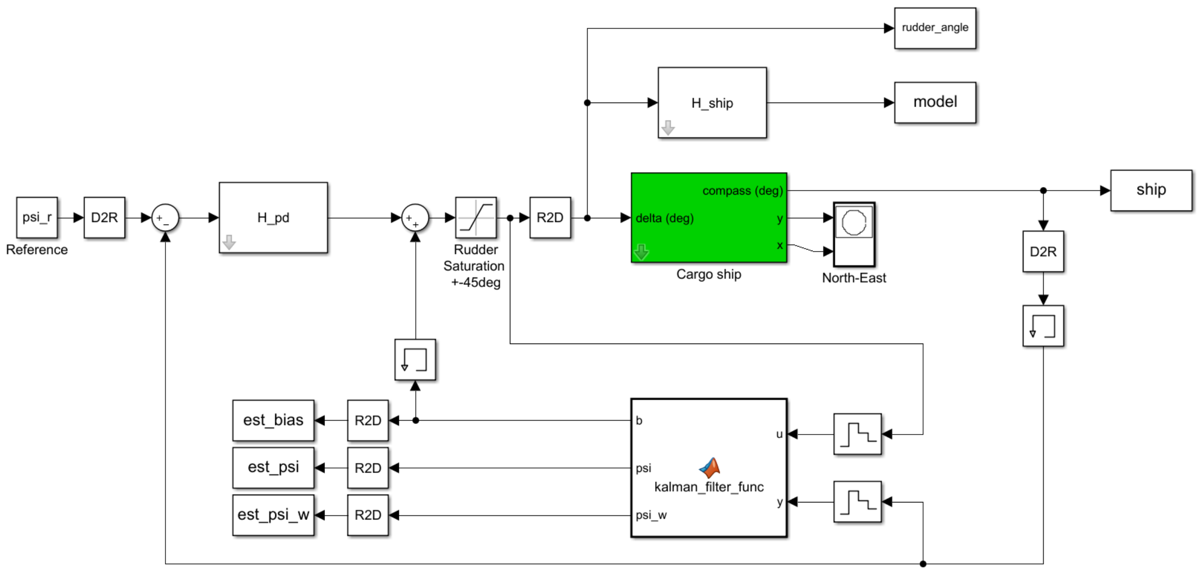

Figure 23: Simulink diagram used in problem 5 d)

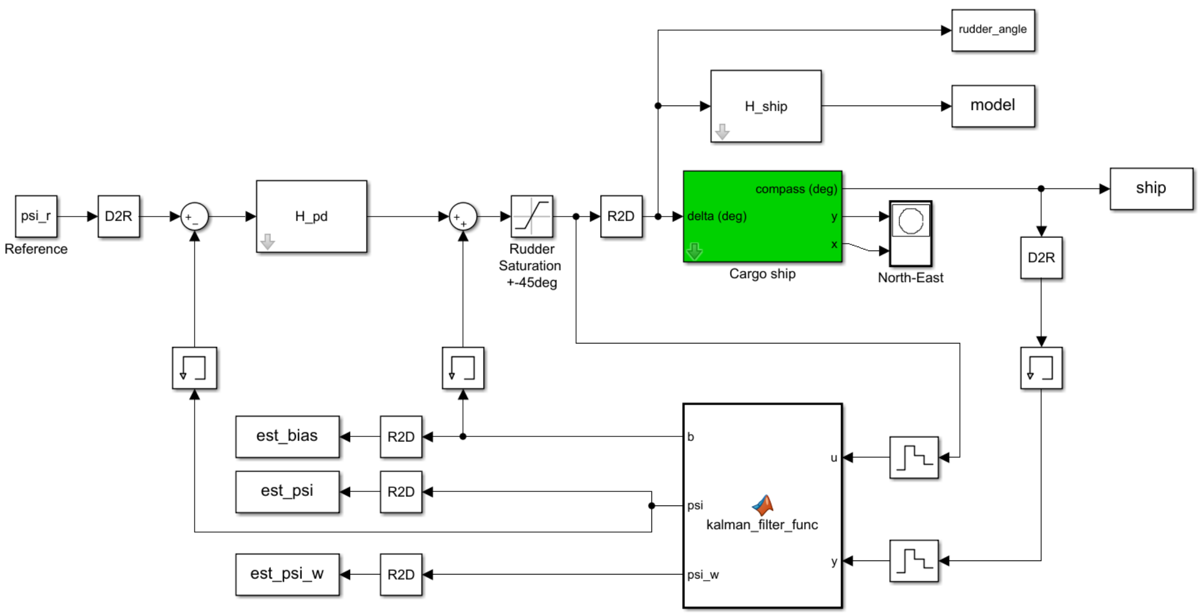

Figure 24: Simulink diagram used in problem 5 e)

References

[1] Patrick Y. C. Hwang Robert G. Brown. *Introduction to Random Signals and Applied Kalman Filtering.* John Wiley & Sons, Inc., 2012.