Linear Systems TTK4115

Boat Lab Report

Group 33:

Bernt Johan Damslora (nr. 759477)
Didrik Rokhaug (nr. 759528)

November 19, 2016

NTNU – Trondheim
Norwegian University of
Science and Technology

Contents

List of Figures

1 Identification of the boat parameters

In the assignment text we were given the following model of a ship:

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

Where ψ is the average heading, ψ_w is a high frequency component due to the wave disturbance, r the yaw rate and b is a bias to the rudder angle δ, w_w and w_b are white noice disturbances, v is meassurement noise and K, T, λ and ω_0 are model parameters.

1.a Transfer function of model

Using this model we can find the transfer function from δ to ψ. Taking the laplace transform of eq. (1c) and eq. (1d) yields

$$s\psi = r \tag{2a}$$
$$sr = -\frac{1}{T}r + \frac{K}{T}(\delta - b) \tag{2b}$$

Assuming no disturbances $b = 0$ and combining eq. (2a) with eq. (2b) gives

$$H_{ship}(s) = \frac{\psi}{\delta}(s) = \frac{K}{s^2T + s} \tag{3}$$

1.b Finding K and T

To find K and T we use the frequency response of eq. (3) with the frequencies $\omega_1 = 0.005$rad/s and $\omega_2 = 0.05$rad/s, both with an amplitude of 1. Using the provided "ship" we found the amplitudes experimentally (see fig. 1 and fig. 2). This yields the equations

$$|H_{ship}(j\omega_1)| = \frac{K}{\sqrt{T^2\omega_1^4 + \omega_1^2}} = 31.9787 \tag{4a}$$
$$|H_{ship}(j\omega_2)| = \frac{K}{\sqrt{T^2\omega_2^4 + \omega_2^2}} = 0.7847 \tag{4b}$$

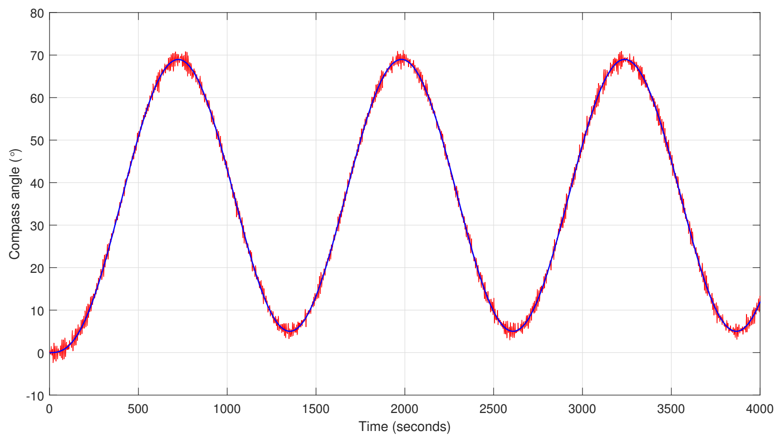

Figure 1: Response of the ship given a input of $\sin 0.005$. Blue line is without any noise, red line is with both waves and meassurement noise.

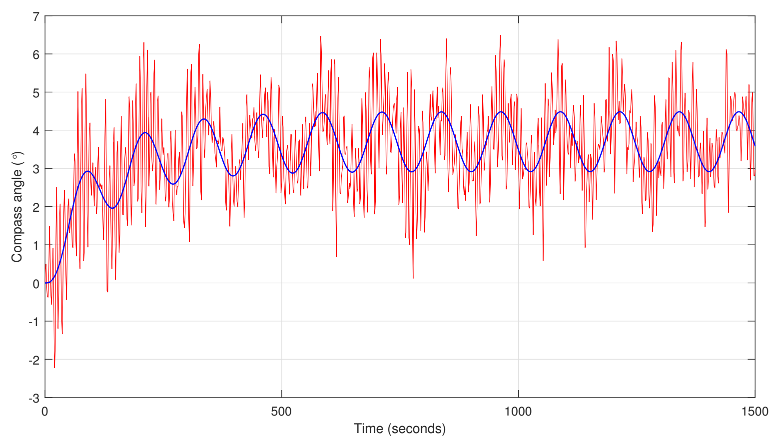

Figure 2: Response of the ship given a input of $\sin 0.05$. Blue line is without any noise, red line is with both waves and meassurement noise.

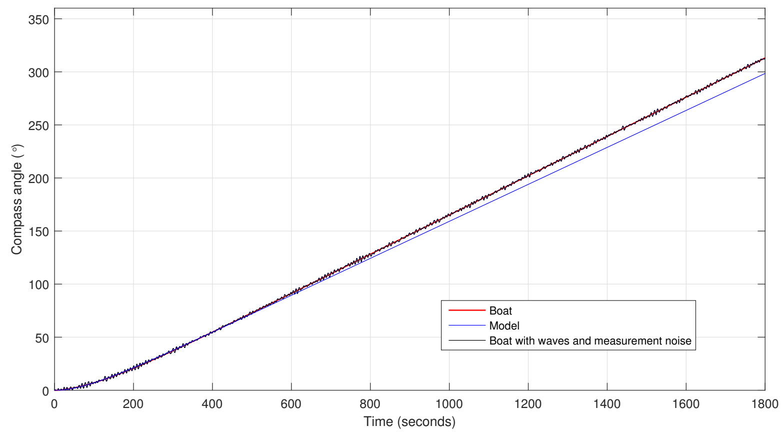

Figure 3: Position of the ship and the model given a step input

Using Matlab function `solve` to solve this set of equations numerically yields

$$K = 0.1742 \tag{5a}$$
$$T = 86.5246 \tag{5b}$$

1.c Not finding K and T in a noisy environment

If we simulate the ship with disturbance we see from fig. 1 and fig. 2 that we get a noise component on top of the sinusoidal response. This makes it harder, if not impossible to find the true amplitude of the response, as we cannot simply use the half difference between the maximum and minimum.

1.d Judging the model

If we apply a step input of 1 degree at $t = 0$ we get the response seen in fig. 3. As we can see the model is close to the "actual" ship at the beginning, but as the time increases, so does the errors of the model. This means that for small ts the model is a good approximation.

Listing 1: The Matlab code used in part 2

```
%% 2a - find Pxx estimate
psi_w_data = psi_w(2,:); % Load data
psi_w_data = psi_w_data * pi / 180; % Convert data to radians
[pxx, f] = pwelch(psi_w_data,4096,[],[],10); % Estimate Pxx
pxx = pxx / (2*pi); % Convert Pxx to s/rad
f = f*2*pi; % Convert frequencies to rad/s

%% 2c - find omega_0
[pxx_max,omega_0_ind]=max(pxx) % Find maximum frequency index
omega_0 = f(omega_0_ind) % Extract said frequency

%% 2d - find lambda
sigma = sqrt(pxx_max); % Sigma is needed in PSD
lambda_0 = 1; % Initial guess for lambda
PSD = @(l, xdata) arrayfun(@(o) (2*l*omega_0*sigma)^2 * o^2 /
    (o^4 + 2*omega_0^2 * (2*l^2 - 1) * o^2 + omega_0^4)) ,
    xdata) % PSD function (o is omega, l is lambda)
lambda = lsqcurvefit(PSD,lambda_0,f,pxx) % Estimate lambda by
     fitting PSD to the Welch estimate from 2a
```

2 Identification of wave spectrum model

2.a Estimating PSD from data

To estimate the Power Spectral Density, we use Welch's estimate, which is implemented in Matlab as seen in listing 1. The result is included in fig. 4.

2.b Defining PSD of model

To find the values of ω_0 and λ we can compare the estimated PSD of the waves with an analytical, found by using the model of the waves. In order to do this we first need to find the transfer function from w_w to ψ_w. This can be done by taking the laplace transform of eq. (1a) and eq. (1b) yielding

$$G(s) = \frac{\psi_w}{w_w}(s) = K_w \frac{s}{s^2 + 2\lambda\omega_0 s + \omega_0^2} \tag{6}$$

To find the power spectral density function we use that $S_y(\omega) = |H(j\omega)|^2 S_u(\omega)$. This means

$$P_{\psi_w}(\omega) = |G(j\omega)|^2 = K_w^2 \frac{\omega^2}{\omega^4 + 2\omega_0^2(2\lambda^2 - 1)\omega^2 + \omega_0^4} \tag{7}$$

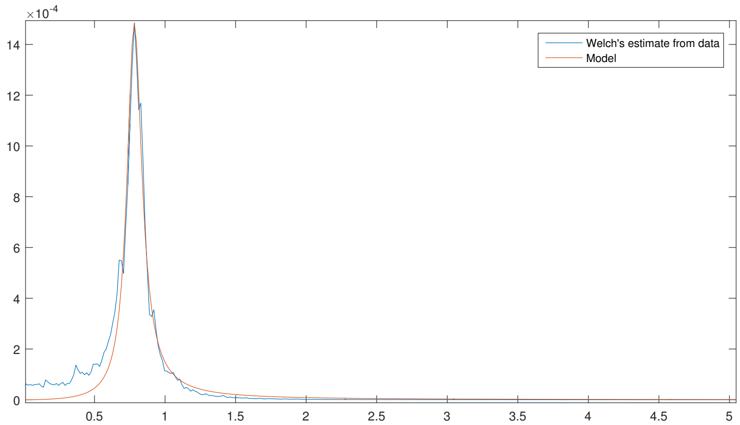

Figure 4: Estimated PSD from data and PSD of model.

2.c Finding ω_0

ω_0 was estimated as the maximum frequency in the PSD estimate, as shown in listing 1. This frequency is 0.7823Hz.

2.d Finding λ

λ was estimated using the Matlab function `lsqcurvefit`, which iterates toward the λ that yields the curve closest to the PSD estimate. The resulting λ is 0.0827.

The PSD of the model, using these parameters, together with the PSD estimate from the data, is plotted in fig. 4.

3 Control system design

3.a PD controller design

We want to design a limited PD controller with the transfer function

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

to control the heading ψ of the ship, by setting the rudder angle δ. We want the cross frequency and phase margin of the open loop system to be 0.10rad/s and 50 degrees respectively. We also want the time constant T_d to equal the time constant of the system. In order to choose K_{pd}, T_d and T_f we first find the transfer function of the open loop system

$$H_{sys}(s) = H_{pd}(s) \cdot H_{ship}(s) = KK_{pd}\frac{1+T_d s}{s(Ts+1)(T_f s+1)} \tag{9}$$

As we want T_d to equal the time constant of the model we set $T_d = T = 86.5246\text{s}$. Next we use that the phase margin of a system is equal to $\angle H(j\omega_c) + 180°$.

$$\begin{aligned}
\angle H_{sys}(j\omega_c) + 180° &= 50° && \text{(10a)}\\
\angle \frac{KK_{pd}}{(j\omega_c)^2 T_f + j\omega_c} &= -130° && \text{(10b)}\\
\angle KK_{pd} - \angle((j\omega_c)^2 T_f + j\omega_c) &= -130° && \text{(10c)}\\
\angle((j\omega_c)^2 T_f + j\omega_c) &= -130° && \text{(10d)}\\
\tan^{-1}\frac{\omega_c}{\omega_c^2 T_f} &= 50° && \text{(10e)}\\
T_f &= \frac{1}{\omega_c \tan(50°)} && \text{(10f)}\\
T_f &= 8.3910\text{s} && \text{(10g)}
\end{aligned}$$

Last we find K_{pd} by using the definition of the cross frequency: $|H(j\omega_c)| = 1$. This yields

$$\begin{aligned}
|H_{sys}(j\omega_c) &= 1 && \text{(11a)}\\
|H_{sys}^2(j\omega_c) &= 1^2 && \text{(11b)}\\
\left|\frac{KK_{pd}}{s^2 T_f + s}\right|^2 &= 1 && \text{(11c)}\\
\frac{(KK_{pd})^2}{(\omega_c^2 T_f)^2 + \omega_c^2} &= 1 && \text{(11d)}\\
K_{pd} &= \frac{\sqrt{(\omega_c^2 T_f)^2 + \omega_c^2}}{K} && \text{(11e)}
\end{aligned}$$

Inserting eq. (10g) into eq. (11e) yields a $K_{pd} = 0.7494$. The controller was implemented in Simulink as seen in fig. 5.

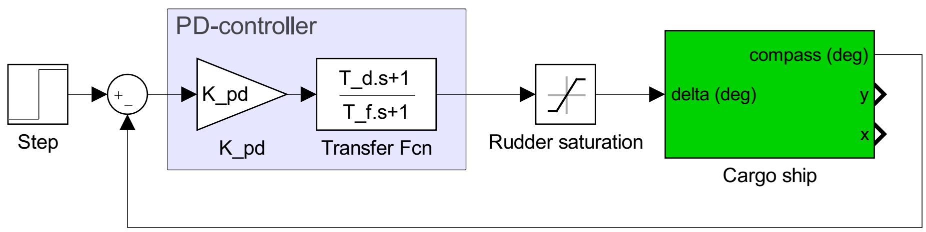

Figure 5: Simulink implementation of the PD-controller.

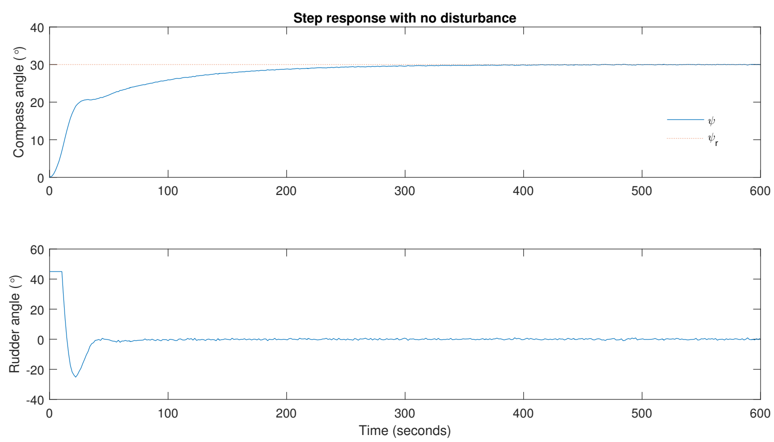

Figure 6: Step response of the controller without any disturances.

3.b Performance without disturbances

As we can see in fig. 6 the controller does a good job when there are no disturbances. The ship gets close to reference ψ_r of 30° about 2 minutes, and the rudder angle is relatively constant over time.

3.c Performance with a current disturbance

In fig. 7 we see the response of the controller with current disturbance. As we can see the controller is unable to counteract the constant disturbance of the current, which leads to a large constant deviation from the setpoint.

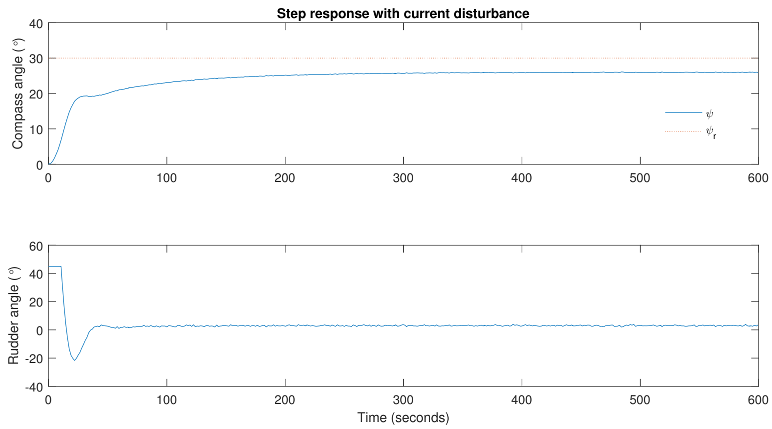

Figure 7: Step response of the controller with current disturbance.

3.d Performance with wave disturbance

When we apply wave current (and turn off current disturbance), we see in fig. 8 that the controller does its best to remove the noise but is unable to do much about it. In its attempt to remove the noise it applies a lot of rudder input, which if the boat were real would stress the physical system.

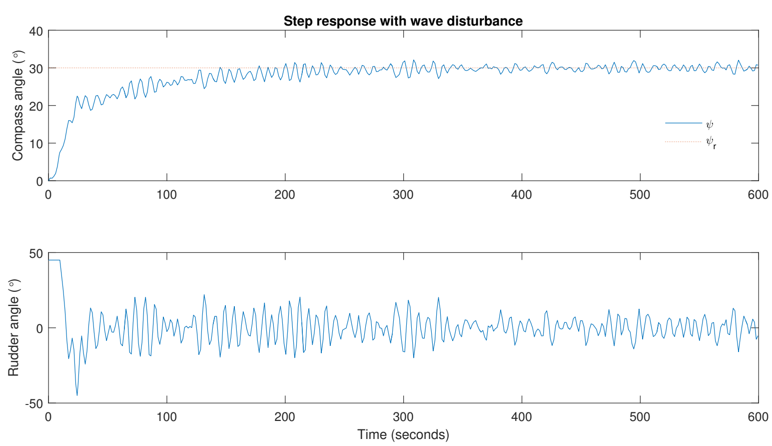

Figure 8: Step response of the controller with wave disturbance.

4 Observability

4.a Defining the matrices of the model system

The model in eq. (1) can be written on the form $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u + \mathbf{E}\mathbf{w}$, $y = \mathbf{C}\mathbf{x} + v$, with $\mathbf{x} = [\xi_w\ \psi_w\ \psi\ r\ b]^T$, $u = \delta$ and $\mathbf{w} = [w_w\ w_b]^T$. Then the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{E}$ are

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\omega_0^2 & -2\lambda\omega_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1/T & -K/T \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{12a}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ K/T \\ 0 \end{bmatrix} \tag{12b}$$

$$\mathbf{E} = \begin{bmatrix} 0 & 0 \\ K_w & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{12c}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \end{bmatrix} \tag{12d}$$

4.b Observability without disturbances

We first checked observability without any disturbance. To do this we removed the disturbances from the system. the resulting $\mathbf{A}$ and $\mathbf{C}$ matrices were

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -1/T \end{bmatrix} \tag{13a}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{13b}$$

To find out whether the system is observable we checked if `rank(obsv(A, C)) == size(A,1)`. This code calculates the rank of the observability matrix $\mathcal{O}$, and compares it to the dimension of the matrix $\mathbf{A}$. Using this code we found that the system is observable without any disturbance.

4.c Observability with a current disturbance

Next we checked observability with the current disturbance. The **A** and **C** matrices used were

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1/T & -K/T \\ 0 & 0 & 0 \end{bmatrix} \tag{14a}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{14b}$$

Using the code in section 4.b we found that the system is observable with current disturbance.

4.d Observability with wave disturbance

Next we checked observability with the wave disturbance. The **A** and **C** matrices used were

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega^2 & -2\lambda\omega_0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1/T \end{bmatrix} \tag{15a}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix} \tag{15b}$$

Using the code in section 4.b we found that the system is observable with wave disturbance.

4.e Observability with both disturbances

Next we checked observability with both current and wave disturbance. The **A** and **C** matrices used were

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\omega_0^2 & -2\lambda\omega_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1/T & -K/T \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{16a}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \end{bmatrix} \tag{16b}$$

Using the code in section 4.b we found that the system is observable with both wave and current disturbance.

5 Discrete Kalman filter

5.a Discretization of the model

To make a discrete Kalman filter we need to discretize the model found in section 4.a. The discretized system can be found using the equations

$$
\begin{align}
A_d &= e^{AT_s} \tag{17a}\\
B_d &= \int_0^{T_s} e^{A\tau} B d\tau \tag{17b}\\
E_d &= \int_0^{T_s} e^{A\tau} E d\tau \tag{17c}\\
C_d &= C \tag{17d}
\end{align}
$$

This can be done using he Matlab function `[Ad, Bd] = c2d(A, B, Ts)` to get the discretized matrices A_d and B_d discretized with a timestep of `Ts`. To get the model discretized with a sampling frequency of 10Hz we set `Ts` to 0.1. In order to get E_d we use the same function, but swap `Bd` and `B` with `Ed` and `E`. The resulting A_d, B_d and E_d matrices are

$$
\begin{align}
A_d &= \begin{bmatrix} 0.1000 & 0.0993 & 0 & 0 & 0 \\ -0.0607 & 0.9841 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.0999 & -1.0063 \cdot 10^{-5} \\ 0 & 0 & 0 & 0.1000 & -0.0002 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{18a}\\
B_d &= \begin{bmatrix} 0 \\ 0 \\ 1.0062 \cdot 10^{-5} \\ 0.0002 \\ 0 \end{bmatrix} \tag{18b}\\
E_d &= \begin{bmatrix} 2.4880 \cdot 10^{-5} & 0 \\ 0.0005 & 0 \\ 0 & -3.3545 \cdot 10^{-7} \\ 0 & -1.0063 \cdot 10^{-5} \\ 0 & 0.1000 \end{bmatrix} \tag{18c}
\end{align}
$$

5.b Estimating the variance of the measurement noise

Another thing we need to make a Kalman filter is the variance of the measurement noise. The variance was found by simulating long timeseries of the ship with a input of zero, and with measurement noise turned on. The variance was found using the Matlab function `ts_var = var(ts)`, where `ts` is a timeseries, and `ts_var` is the variance of the timeseries. The variance of the measurement noise was found to be 0.002. However this is in degrees squared, whereas we want the variance in radians (squared). This gives us a variance of $6.0923 \cdot 10^{-6}$.

5.c Implementation of the Kalman filter

A Kalman filter was implemented using Simulink with a Matlab function block. The custom Kalman filter subsystems structure is shown in fig. 9. The rudder angle and measured compass angle are converted to radians, discretized by a zero-order hold block, and passed into the Kalman filter function. The estimated state from the filter is passed through a memory block to avoid an algebraic loop, converted back to degrees, and finally split up to expose the different states.

The Kalman filter function itself is listed in listing 2. The first time the function is called, it initializes the a priori states to the provided values. After that, it calculates the Kalman gain, state estimate and covariance estimate using the update equations

$$K_k = P_k^- C_d^T (C_d P_k^- C_d^T + R)^{-1} \tag{19a}$$
$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - C_d \hat{x}_k^-) \tag{19b}$$
$$P_k = (I - K_k C_d) P_k^- (I - K_k C_d) + K_k R K_k^T \tag{19c}$$

and finally updates the a priori estimates using the prediction equations

$$\hat{x}_{k+1}^- = A_d \hat{x}_k \tag{20a}$$
$$P_{k+1}^- = A_d P_k A_d^T + E_d Q E_d^T \tag{20b}$$

5.d Performance with current disturbance

The estimated bias from the Kalman filter was added to the rudder angle input, as shown in fig. 10.

As we can see from fig. 11, the estimated bias feed-forward tackles the current disturbance, eliminating the constant deviation. In this respect it is clearly superior to the simple PD controller, which is unable to reach the reference compass angle, as seen in fig. 7.

Listing 2: Matlab implementation of the discrete Kalman filter

```
function x_est = kalman_m( psi_meas, delta )
% Definition of system
Ad = [0.9970,   0.09925, 0,     0,          0;
        -0.06074, 0.9841,  0,     0,          0;
         0,        0,       1,     0.09994,  -1.006e-05;
         0,        0,       0,     0.9988,   -0.0002012;
         0,        0,       0,     0,          1];
Bd = [0; 0; 1.006e-05; 2.0121e-04; 0];
Cd = [0, 1, 1, 0, 0];
Ed = [2.488e-05, 0;
      4.963e-04, 0;
      0,         -3.355e-07;
      0,         -1.006e-05;
      0,          0.1000];
R = 6.0923e-07;
Q = [30 0; 0 1e-6];

% Initialization of a priori states
persistent x_est_pre P_est_pre init_flag
if isempty(init_flag)
    init_flag = 1;
    x_est_pre = [0; 0; 0; 0; 0];
    P_est_pre = diag([1 0.013 pi^2 1 2.5e-4]);
end

% Update step
K = P_est_pre*transpose(Cd)/(Cd*P_est_pre*transpose(Cd) + R);
x_est = x_est_pre + K * (psi_meas - Cd*x_est_pre);
P_est = (eye(5)-K*Cd)*P_est_pre*transpose(eye(5)-K*Cd) + K*R*transpose(K);

% Prediction step
x_est_pre = Ad*x_est + Bd*delta;
P_est_pre = Ad*P_est*transpose(Ad) + Ed*Q*transpose(Ed);
```

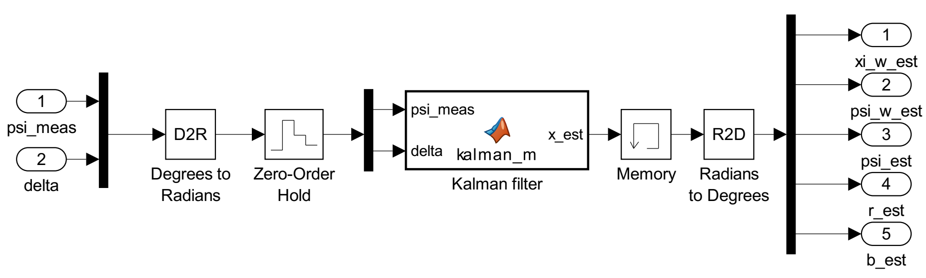

Figure 9: Implementation of the Kalman filter subsystem in Simulink.

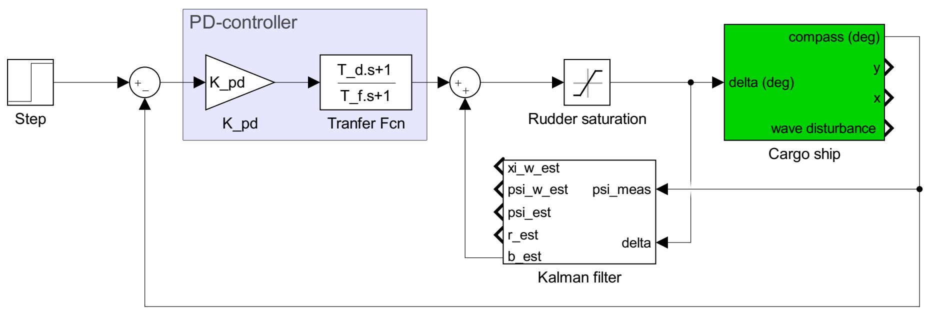

Figure 10: Simulink implementation of PD control with estimated bias feed forward.

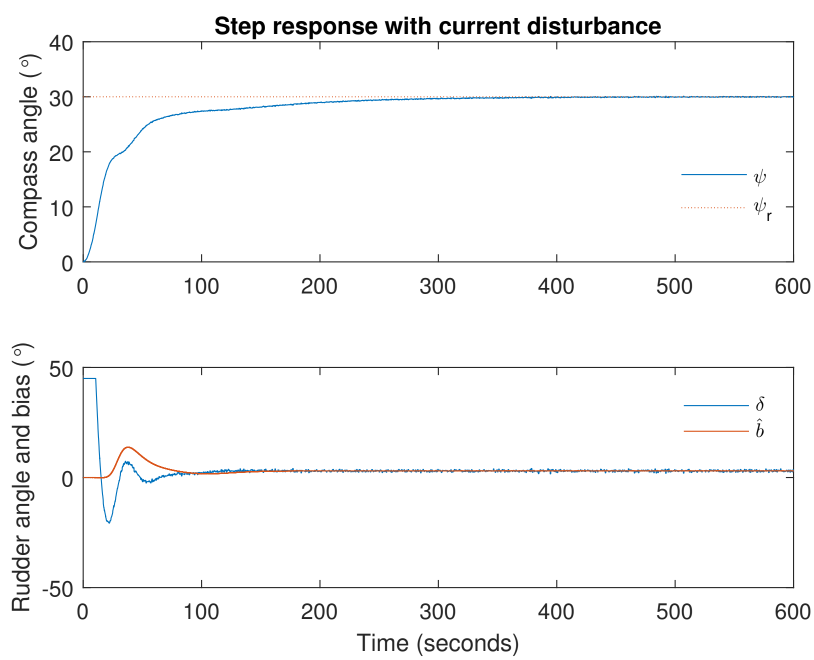

Figure 11: Step response with estimated bias feed-forward.

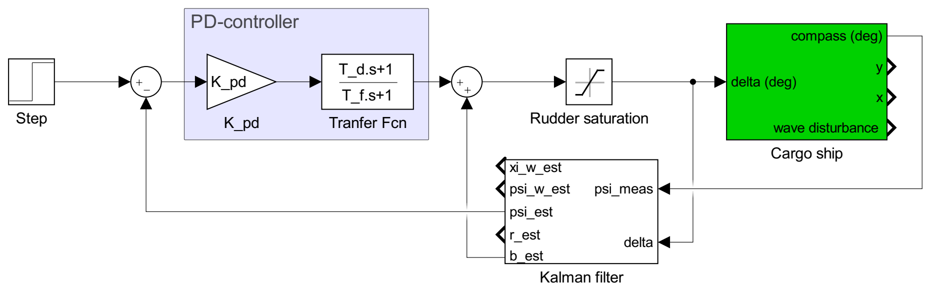

Figure 12: Simulink implementation of PD control with estimated bias feed forward, and filtered feedback.

5.e Performance with both disturbances

The PD controller feedback was changed from the measurment to the wave-filtered estimate from the Kalman filter, as seen in fig. 12.

As we can see from fig. 13, the new controller based on the wave-filtered output keeps the compass angle about as well as the simple PD controller (fig. 8) when exposed to wave disturbance, but with considerably smaller adjustments to the rudder angle over time. Avoiding stress of components, this is a clear improvement.

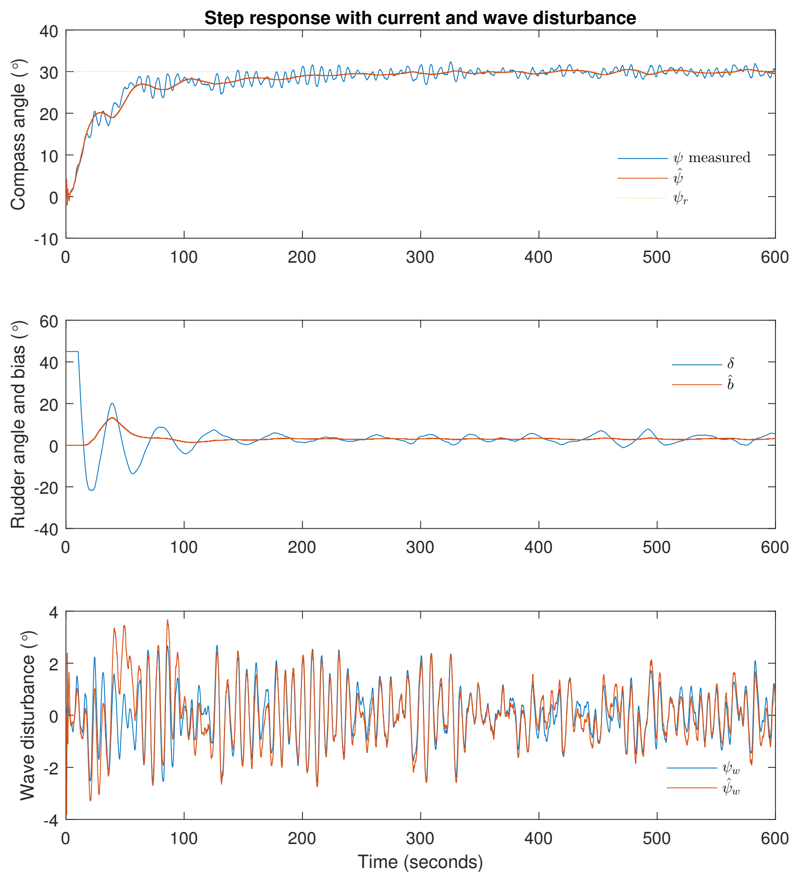

Figure 13: Step response with estimated bias feed-forward and wave-filtered feedback.