TTK4190 Guidance and Control of Vehicles

Assignment 2, Pt. 5

Written Fall 2023

Seknaya, Sacit Ali
sacitas@stud.ntnu.no

Kopperstad, Håvard Olai
haavarok@stud.ntnu.no

Almenningen, Elias Olsen
eliasoa@stud.ntnu.no

November 15, 2023

Problem 1 - Kalman Filter Design

a)

The system equation when using a first-order Nomomto model with rudder bias is stated in [1] as Equation (13.44 - 13.46)

$$\dot{\psi} = r \tag{1}$$

$$\dot{r} = -1\frac{1}{T}r + \frac{K}{T}(\delta - b) + w_1 \tag{2}$$

$$\dot{b} = w_2 \tag{3}$$

The states are chosen as $x = [\psi,\, r,\, b]^\top$ and the Kalman filter matrices becomes

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{1}{T} & -\frac{K}{T} \\ 0 & 1 & 0 \end{bmatrix} \tag{4}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 \\ \frac{K}{T} \\ 0 \end{bmatrix} \tag{5}$$

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{6}$$

$$\boldsymbol{D} = 0 \tag{7}$$

$$\boldsymbol{E} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{8}$$

The state-space model is then

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\delta + \boldsymbol{E}\boldsymbol{w} \tag{9}$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\delta + \varepsilon \tag{10}$$

where ε is zero-mean Gaussian measurment noise.

b)

Discretizing the matrices found in a) is done by first-order discretization as stated in p. 409 in [1]:

$$\boldsymbol{A}_d[k] = \boldsymbol{I} + \boldsymbol{A} * h = \begin{bmatrix} 0 & h & 0 \\ 0 & -\frac{h}{T} & -h\frac{K}{T} \\ 0 & h & 0 \end{bmatrix} \tag{11}$$

$$\boldsymbol{B}_d[k] = \boldsymbol{B} * h = \begin{bmatrix} 0 \\ h\frac{K}{T} \\ 0 \end{bmatrix} \tag{12}$$

$$\boldsymbol{C}_d[k] = \boldsymbol{C} \tag{13}$$

$$\boldsymbol{D}_d[k] = \boldsymbol{D} \tag{14}$$

$$\boldsymbol{E}_d[k] = \boldsymbol{E} * h = \begin{bmatrix} 0 & 0 \\ h & 0 \\ 0 & h \end{bmatrix} \tag{15}$$

c)

To check if the system is observable, MatLab is used, and symbolic values for K, T is used.

```
syms T K h
A = [ 0   1    0;
```

```
       0 -1/T -K/T;
       0   0    0  ];
B = [ 0;
     K/T;
       0  ];
C = [1 0 0];

D = 0;

E = [ 0 0;
       1 0;
       0 1];

Ad = eye(3) + h * A;
Bd = h * B;
Cd = C;
Dd = D;
Ed = h * E;

Ob  = [C;C*A;C*A*A]
unobsv = length(A) - rank(Ob)

Ob_d = [Cd;Cd*Ad;Cd*Ad*Ad]
unobsv_d = length(Ad) - rank(Ob_d)
```

which yields

```
Ob =
    [1,    0,    0]
    [0,    1,    0]
    [0, -1/T, -K/T]

unobsv =
     0

Ob_d =
    [1,               0,          0]
    [1,               h,          0]
    [1, h - h*(h/T - 1), -(K*h^2)/T]

unobsv_d =
     0
```

Both the continious and discrete system has zero unobservable states, meaning that both systems are observable.

Problem 2 - Implementation

a)

The noisy measurements is generated in matlab using *normrnd(.)* as shown below:

```
psi_noise = normrnd(psi_deg,0.5);
r_noise   = normrnd(r,0.1);
```

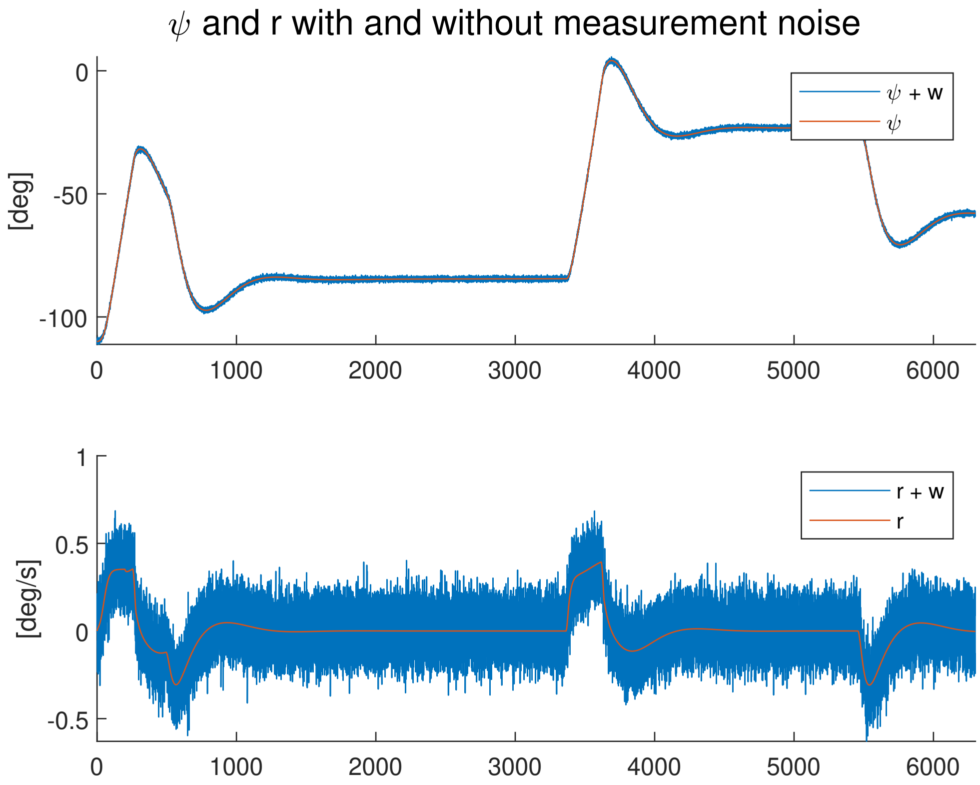


Figure 1: Noisy measurements and true states

b)

The discrete-time Kalman filter is implemented in MatLab as

```
    % Kalman Filter:
    psi_meas                    = normrnd(x(6),deg2rad(0.5));
    r_meas                      = normrnd(x(3),deg2rad(0.1));
    [x_pst,P_pst,x_prd,P_prd]   = KF(x_prd,P_prd,Ad,Bd,Ed,Cd,Dd,Qd,Rd,psi_meas,x(7));
```

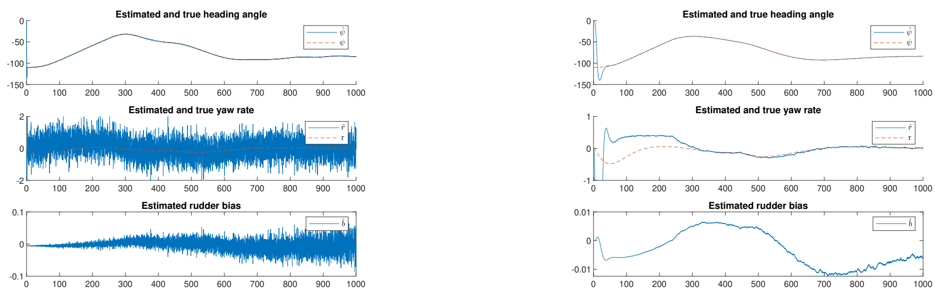
where the function KF is the Kalman filter using the pseudocode in [1] on page 416:

```
function [x_pst,P_pst,x_prd,P_prd] = KF(x_prd,P_prd,Ad,Bd,Ed,Cd,Dd,Qd,Rd,psi_meas,Δ)
% initialization
n       = length(Ad);
u       = Δ;

% KF gain: K[k]
K = P_prd * Cd' * inv( Cd * P_prd * Cd' + Rd );
```

```
IKC = eye(n) - K * Cd;

% Measurement: y[k]
y = psi_meas;

% Corrector: x_hat[k] and P_hat[k]
x_pst = x_prd + K * ( y - Cd * x_prd - Dd * u );
P_pst = IKC * P_prd * IKC' + K * Rd * K';

% Predictor: x_prd[k+1] and P_prd[k+1]
x_prd = Ad * x_pst + Bd * u;
P_prd = Ad * P_pst * Ad' + Ed * Qd * Ed';
end
```

When tuning $P(0)$, Q, R we tried to see what made the estimated states coincide with the true states, see Figure 2. $\hat{\psi}$ goes converges towards ψ, regardless of tuning, but this makes sense, as it is the state that is measured. With $Q >> R$, $\hat{r}$, $\hat{b}$ are noisy and do not seem to converge. When $R >> Q$, $\hat{r} \rightarrow r$ and b is much less noisy. The initial value of P does not seem to matter. The final values of our filter is then

```
% Kalman init
x_prd = [0 0 0]';
P_prd = diag([0 0 0]);
Qd    = diag([1 1]);
Rd    = 1000;
```



(a) $Q >> R$

(b) $Q << R$

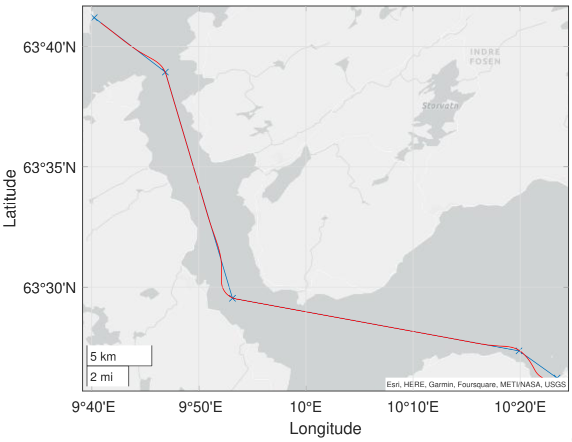
Figure 2: The tuning of Q and R

c)

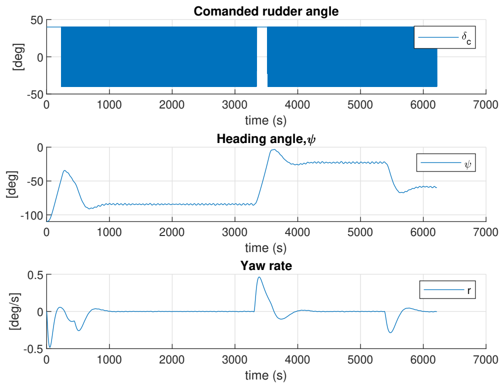
When simulating the path following system using crab angle compensation, and using the noisy measurements directly into the heading contoller, see Figure 3a, the ship manages to follow the trajectories, but the commanded rudder angle is oscillating rapidly between maximum and minimum saturation. This is not a good thing and would likey destroy it in the real world.

d)

When filtering the measurements, before using them in the controller, see Figure 3b that the commanded rudder angle is much less oscillatory. It is still noisy and would not be a good thing on a real ship, but it shows that the Kalman filter is doing its job. The ship still manages to follow the path. The gains of the PID controller were also adjusted to improve δ_c. To make it even better, more tuning of the Kalman filter and/or PID controller is necessary.

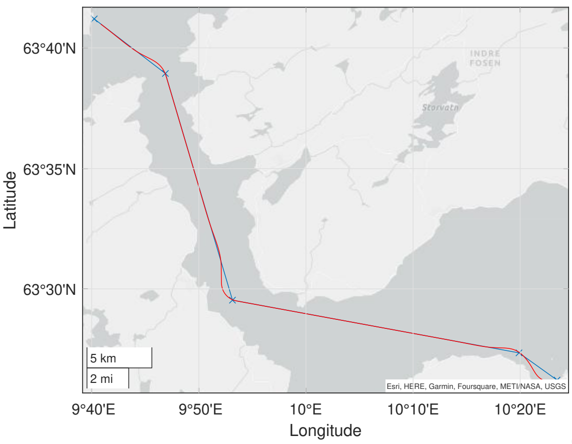


(a) The position using the noisy measurements directly into the heading controller

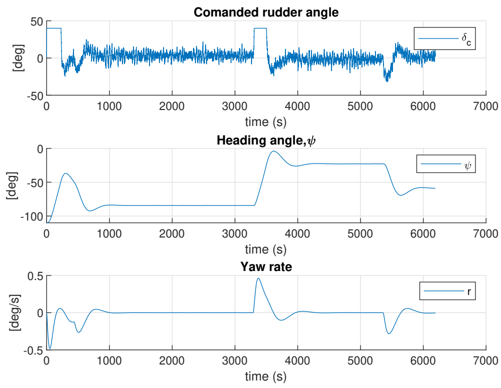


(b) The commanded rudder, heading angle and yaw rate using the noisy measurements directly into the heading controller

Figure 3: Noisy measurements directly into the heading controller



(a) The position using the noisy measurements directly into the heading controller



(b) The commanded rudder, heading angle and yaw rate using the Kalman Filter measurements directly into the heading controller

Figure 4: Kalman Filter measurements into the heading controller

Problem 3 - Navigation Systems

a)

A wave filter is usually a model-based state estimator that separates the position and heading measurements into low-frequency and wave-frequency position and heading signals. Wave filtering is crucial in ship motion control systems since wave-frequency part of the motion should not be compensated for by the control system unless wave-induced vibration damping is an issue. If the wave-frequency part of the motion enters the feedback loop, this will cause unnecessary use of the actuators (thrust modulation) and reduce the tracking performance, which, again results in increased fuel comsumption [1].

The linear wave model under can be used to simulate the wave response and can be further augmented with the rest of the ship model.

$$\begin{aligned}\dot{\xi}_w &= \psi_w \\ \dot{\psi}_w &= -\omega_0^2 \xi_w - 2\lambda\omega_0\psi_w + K_w w_3\end{aligned}$$

b)

In a model-based navigation system, the craft position, velocity and attitude are states in the estimator, while linear acceleration and angular rates are generated using a mathematical model. Alternatively, in a Inertial Navigation System (INS) the craft model can be avoided by using accelerometers and angular rate (ARS) measurements as inputs and integrate the kinematic equations.

The drawback of the model-based approach to aided INS is model uncertainty for the craft. One obvious advantage is that additional sensors such as the inertial measurement unit (IMU) are avoided. Another benefit is that the mathematical model can be used for fault detection and isolation, as well as fault recovery [1].

References

[1] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, 2011.