Project assignment in

TTK4115

Linear Systems

Discrete Kalman filter applied to a ship autopilot.

Norges teknisk- naturvitenskapelige universitet
Institutt for teknisk kybernetikk
TRONDHEIM

Versjon 1.0, October 27, 2003, JS

Contents

1 Practical information

- The assignment is mandatory, but will not be graded.
- The assignment shall be carried out in groups, using the same groups as the Helicopter project.
- The amount of time to complete the assignment is estimated to 20 hours. Each group will get 8 hours of guidance.
- The time schedule will be put on the course homepage.
- The file boat.zip can be downloaded from the course home page.

2 Purpose of the assignment

The purpose of the assignment can be summarized as:

- The students will partly model and simulate a continuous system influenced by stochastic signals.
- All parameters in the system are not explicitly given, so the students will have to use basic identification techniques to have a complete model.
- Using basic control theory, a simple autopilot shall be designed.
- A discrete Kalman filter for wave filtering and estimation of disturbances is to be implemented using MATLAB and Simulink.

3 Background material

3.1 Coordinate systems

3.1.1 Reference frames

In navigation several reference frames are used. By a reference frame we mean which coordinate system a vector is described relatively to. We only consider two coordinate system in this assignment, NED and BODY.

- NED is a coordinate system in which the x-axis point to the north, the y-axis point east and the z-axis towards the center of the earth (down).

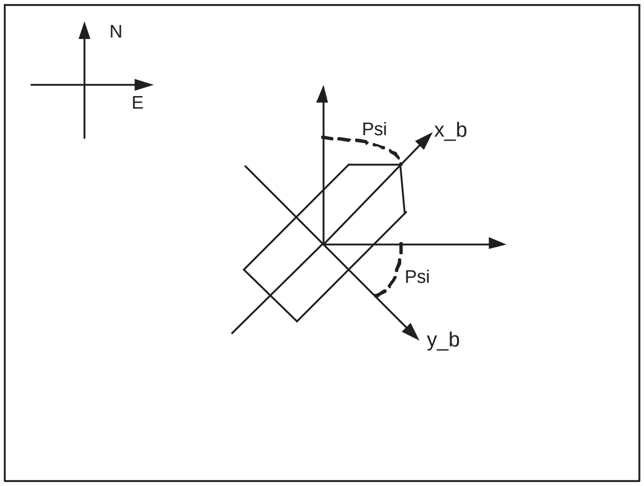

Figure 1: Body and NED reference frames

- BODY is a coordinate system where the x-axis is along the longitudinal axis of the boat(from aft to fore). The y-axis is along the transversal axis(to starboard) and the z-axis along the normal axis(from top to bottom).

Figure 1 illustrates the BODY and NED reference frames. In figure 1 the heading ψ is depicted. ψ is the compass measurement.

3.1.2 Transforming vectors between different ref. frames

This section is not intended as a complete description of the subject, but a short introduction to how transformations are carried out. Given a vector $\mathbf{v}^b$, where the superscript b denotes that $\mathbf{v}$ is described in the BODY reference frame. We can transform this vector such that it is described in the NED reference frame. In the horizontal plane this becomes:

$$\mathbf{v}^n = \mathbf{R}_b^n(\psi)\mathbf{v}^b \tag{1}$$

$$\mathbf{R}_b^n = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \tag{2}$$

The transformation can also be reversed:

$$\mathbf{v}^b = \mathbf{R}_n^b(\psi)\mathbf{v}^n = \mathbf{R}_b^n(\psi)^T\mathbf{v}^n \tag{3}$$

since $\mathbf{R}^T = \mathbf{R}^{-1}$ when $\mathbf{R}$ is a rotation matrix.
To clarify we will give an example:

Example 1 (Rotation of a vector) *Let $\psi = \frac{\pi}{4}$ and $\mathbf{v}^b = [0\ 1]^T$. Carrying out the calculation as described in (1) yields a vector $\mathbf{v}^n = [-0.7071\ 0.7071]^T$. Figure 2 depicts the two vectors. The transformation from NED to BODY can be carried out to see that $\mathbf{v}^b = [0\ 1]^T$.*

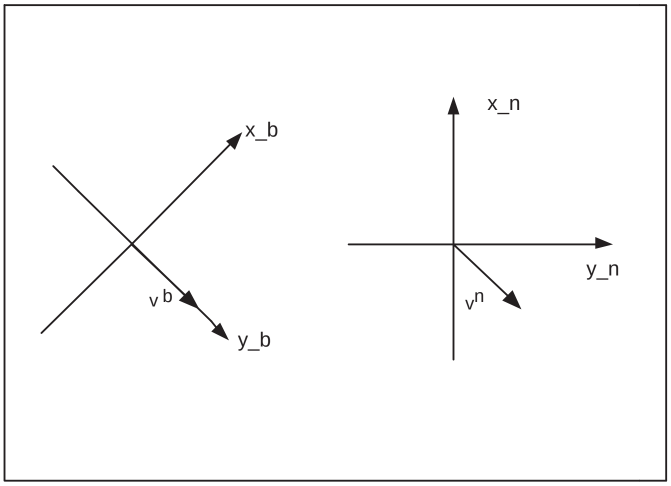

Figure 2: The same vector in different frames

3.2 System description

We shall look at a model of a ship, influenced by both current and waves. The three following sections presents the necessary background material to complete the project. We will model the ship as a system which is not affected by waves and try to find an estimate of the course angle without the wave disturbance.

3.2.1 The ship

A nonlinear dynamical model of a ship can be represented as:

$$\dot{\eta} = \mathbf{R}(\psi)\nu \tag{4}$$

$$\mathbf{M}\dot{\nu} + \mathbf{C}(\nu)\nu + \mathbf{D}(\nu)\nu = \tau + \mathbf{w} \tag{5}$$

where

- **M** - system inertia matrix
- **C** - Coriolis-centripetal matrix

- $\mathbf{D}$ - damping matrix
- τ - vector of control inputs
- $\mathbf{w}$ vector of environmental disturbances
- η - NED positions $[x, y, \psi]$. Where x is the north-direction, y is the position in the east-direction, and ψ is the angle between the north direction and the x_b axis. ψ is positive clockwise.
- ν - BODY velocities $[u, v, r]$. Where u is the velocity in the x-direction, v the velocity in the y-direction and r is rotation velocity about the z-axis.

Assuming that the speed is low some of the nonlinear terms are negligible and hence the equation reduces to:

$$\dot{\eta} = \mathbf{R}(\psi)\nu \tag{6}$$

$$\mathbf{M}\dot{\nu} + \mathbf{C}\nu + \mathbf{D}\nu = \tau + \mathbf{w} \tag{7}$$

We now have a linear equation for the BODY velocities. But equation (6) is still non-linear. This will be simplified later. Assuming that forward speed u is constant, we simplify the model by only considering the sway(v)-yaw(r) dynamics. Also letting $\tau = \mathbf{B}\delta$, where δ is the rudder angle, relative to the BODY frame, we get the equation:

$$\mathbf{M}\dot{\nu} + \mathbf{N}(u_0)\nu = \mathbf{B}\delta + \mathbf{w}_{waves} + \mathbf{w}_{current} \tag{8}$$

where $\nu = [v \ r]^T$ and $\mathbf{N}(u_0) = \mathbf{C}(u_0) + \mathbf{D}(u_0)$.

If we are only interested in ψ, we get the following equation for η:

$$\dot{\eta} = \dot{\psi} = r \tag{9}$$

3.2.2 Waves

The waves are considered as high frequent disturbances. The response to the waves can be modelled as a damped harmonic oscillator:

$$\begin{bmatrix} \dot{x}_{w1} \\ \dot{x}_{w2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -w_0^2 & -2\lambda w_0^2 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \end{bmatrix} + \begin{bmatrix} 0 \\ K_w \end{bmatrix} w_w \tag{10}$$

$$y_w = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \end{bmatrix} \tag{11}$$

where w_w is a zero mean white noise process with unity variance. This representation of the waves corresponds to a spectral factorization of the wave spectrum. How to add this to the model is shown in section 3.2.4.

3.2.3 Current

The current is a slowly varying disturbance. We will assume that only effect of the current is a rudder angle bias b. This bias is modelled as:

$$\dot{b} = w_b \tag{12}$$

where w_b is Gaussian white noise.

3.2.4 The complete system

In the model of the system (not the actual process) that will be used in the rest of the assignment the state vector given by: $[\xi_w \ \psi_w \ \psi \ r \ b]^T$, where

- ψ - Is the heading, as described above.
- ψ_w - Is a high frequent component due to the wave disturbance.
- $\dot{\xi}_w = \psi_w$
- r - As described above.
- b - Bias to the rudder angle.

The model which will be used can be stated as:

$$\begin{aligned}
\dot{\xi}_w &= \psi_w \\
\dot{\psi}_w &= -w_0^2\xi_w - 2\lambda w_0\psi_w + K_w w_w \\
\dot{\psi} &= r \\
\dot{r} &= -\frac{1}{T}r + \frac{K}{T}(\delta - b) \\
\dot{b} &= w_b \\
y &= \psi + \psi_w + v
\end{aligned} \tag{13}$$

where y is the measured heading (compass measurement), w_b, w_w, v are white noise processes.

Clearly, the system can be written as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u + \mathbf{E}\mathbf{w}, \quad y = \mathbf{C}\mathbf{x} + v \tag{14}$$

The purpose of this model is to estimate the course angle without the wave disturbance. Hence, we model the ship as a system not affected by waves and include the disturbance only in the measurement. Further, the current only affects the rudder angle in the model. This is of course not the case for an actual ship, but is simplifies the Kalman filter design.

4 Exercises

Unzip the files from boat.zip to a desired directory.The files in boat.zip are:

- ship.mdl: Simulink file representing a cargo ship.
- wave.mat: The wave disturbance, ψ_w.
- Sfunctionshell.m: A half-finished discrete Kalman filter.
- vec2mat.m and mat2vec.m: Files which converts a vector to matrix and vice versa.

Open the mdl file "Ship.mdl". You will se a block which represents the actual boat(a cargo ship). The output of the model is *compass*, x, and y. The ship is assumed to have a constant forward speed u. The input to the ship is the rudder set-point, which is given in degrees. Clicking on the ship model opens a dialog box in which currents, waves and measurement noise can be turned on and off. Note that the rudder angle is constrained to ± 45 degrees.

4.1 Identification of the boat parameters.

a) Assume that there are no disturbances. Calculate the transfer function from δ to ψ, $H(s)$, parameterized by T and K.

b) Turn off all disturbances in the model. We want to identify the boat parameters T and K. Apply a sine input with amplitude 1 and frequency $w_1 = 0.005$(rad/s). Then apply a sine input with amplitude 1 and frequency $w_2 = 0.05$(rad/s). The amplitude of the sine waves on the output equals $|H(jw)|$. This gives us two equations with two unknowns. Repeat the exercise with with waves and measurement noise. This correspond to identifying the parameters in clear and poor weather conditions, respectively. Is it possible to get good estimates of the boat parameters in the latter case?

Apply a step input of 1 degree to the ship at $t = 0$ and compare the step response of the model with the response of the ship. Is the model a good approximation?

In the rest of the assignment the measurement noise shall be turned on.

4.2 Identification of wave spectrum model

a) Load the wave.mat file. The second row in the resulting matrix psi_w contains the influence the waves have on the compass measurement, that is ψ_w. NB! ψ_w is given in degrees in the file. The first row is the time instants the elements of ψ_w is applied to the system. Find an estimate of the Power Spectral Density function (PSD) of ψ_w, $S_{\psi_w}(\omega)$. The sampling frequency is $10Hz$. (Hint: Matlab function "psd", you only have to use the second row.)

b) Find an analytical expression for the transfer function of the wave response model (from w_w to ψ_w). Also find an analytical expression for the Power spectral density function of ψ_w, that is $P_{\psi_w}(\omega)$.

c) Find w_0 from the estimated $S_{\psi_w}(\omega)$.

d) To have a complete model for the wave response we need to identify the damping factor λ. Define $K_w = 2\lambda\omega_0\sigma$ where $\sigma = 0.2716$. Find λ by fitting the $P_{\psi_w}(\omega)$ to the estimate of the PSD, $S_{\psi_w}(\omega)$. Use trial and error.

4.3 Control system design

In this section we want to design an autopilot for the ship. That is, we want to be able to give a desired course angle ψ_r, and get the ship to follow this course.

a) Design a PD controller, $H_{pd}(s) = K_{pd}\frac{1+T_d s}{1+T_f s}$, based on the transfer function from δ to ψ without disturbances. Let the w_c and the phase margin of the open loop system, $H_{pd}(s) \cdot H_{ship}(s)$, be approximately 0.10 (rad/s) and 50 degrees, respectively. Choose the derivative time constant, T_d, such that it cancels the transfer function time constant.

b) Simulate the system without disturbances (only measurement noise). Does the autopilot work?

c) Simulate the system with a current disturbance(and without wave disturbance). Does the autopilot work in this case?

d) Simulate the system with wave disturbance(and without the current disturbance), does the system work satisfactory?

4.4 Observability

a) Find the matrices A, B, C and E in equation (14).

b) Is the system observable without disturbances?

c) Is the system observable with the current disturbance?

d) Is the system observable with the wave disturbance?

e) Is the system observable with both current and wave disturbance?

4.5 Discrete Kalman filter

In this section we shall implement a discrete Kalman filter to estimate the bias b, the heading ψ and the high frequency wave induced motion on the heading ψ_w. ψ_w must be removed from the control loop to avoid tear and wear on the actuator system. That is, we do not want the rudder to compensate for ψ_w. Hence, we use only the estimated ψ in the control law. This is referred to as wave filtering.

a) Discreteize the model found in exercise 4.4 a) using exact discretization. Use a sample frequency of $10Hz$

b) Find an estimate of the variance of the measurement noise.

c) Let now:

$$\mathbf{w} = [w_w \; w_c]^T, \; E\{\mathbf{w}\mathbf{w}^T\} = \mathbf{Q} = \begin{bmatrix} 30 & 0 \\ 0 & 10^{-6} \end{bmatrix}, \tag{15}$$

$$\mathbf{P}_0^- = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.013 & 0 & 0 & 0 \\ 0 & 0 & \pi^2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2.5 \cdot 10^{-4} \end{bmatrix}, \; \hat{\mathbf{x}}_0^- = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Since the process is sampled, $E\{vv\} = R$ equals the variance found in b) divided by the sample interval. Implement a discrete time Kalman filter using an m-file(or c) s-function. (Hint: Let the compass measurement and the rudder input be input to the s-function and the output be the a posteriori estimate of ψ and b. Also put a zero order hold on the compass measurement and the rudder command, since the process and controller is continuous. Use the same sampling frequency as above.)

d) Make a feed forward from the estimated bias such that the bias is cancelled. Simulate the system with the current disturbance. Does the autopilot have a better performance than the equivalent simulation in exercise 4.3 c)?

e) Use the wave filtered ψ instead of the measured heading in the autopilot. Simulate the system with wave and current disturbance. Does the autopilot have a better performance than the equivalent simulation in exercise 4.3 d)?

A S-function hints

We start by looking at the structure of an s-function, and we will only consider the elements which are of relevance to this assignment. The s-function is called by simulink with parameters x, u, t and $flag$, where x is the current state, u is the input, t is the time and $flag$ decides which part of the s-function that shall be executed. The execution flow for the s-function in this assignment is depicted in figure 3.

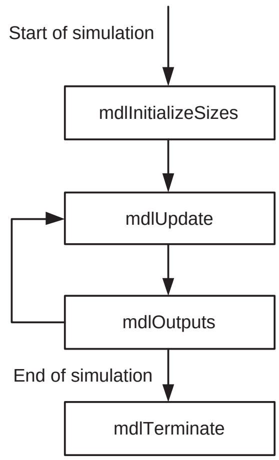

Figure 3: Execution flow of an s-function

- mdlInitializeSizes: Simulink calls this function only at the start of the simulation. Here the system struct is initialized. That is, the number of discrete/contiuous states, inputs, outputs are defined. Sample times and initial conditions are also specified.
- mdlUpdate: The discrete states are updated when simulink calls this function. We can look at the update function as:

$$\mathbf{x}(t + \Delta t) = f(\mathbf{x}(t), \mathbf{u}(t), t) \tag{16}$$

  where Δt is the sampling interval.

  Example: For a linear time invariant system we have $\mathbf{x}(t + \Delta t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$. In the s-function syntax we get:

```
sys=A*x+B*u;
```

- mdlOutputs: When simulink calls this function, the output of the s-function is calculated. We can look at the output function as:

$$\mathbf{y}(t) = h(\mathbf{x}(t), \mathbf{u}(t), t) \tag{17}$$

  Example: For a linear time invariant system this the function reduces to: $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)$. In the s-function syntax we get:

```
sys=C*x+D*u;
```

- mdlTerminate: Simulink only calls this function when the simulation terminates.

It also possible to define additional parameters as input to the s-function.

Clearly, we can write the discrete kalman filter with use of the update and output function. There are many ways of implementing the filter, we will describe two approaches below.

1. One can define all the matrices, that is, $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{E}, \mathbf{Q}, \mathbf{R}$ and $\mathbf{P}$ as global variables. Then let the a priori and the a posteriori estimates be state variables. Define the output to be the a posteriori estimates and the input to be the compass measurement and the rudder input. However, usually we want to avoid use of global variables.

2. Let a struct which contains the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{E}, \mathbf{Q}, \mathbf{R}, \mathbf{P_0}^-$ and $\hat{\mathbf{x}}_0^-$ be a parameter sent to the s-function. The matrices can then be used in all the sub-routines of the s-function. Let the state vector be given by the a priori estimates, the a posteriori estimates and the elements of the error covariance matrix, that is:

$$\mathbf{x} = [\hat{\mathbf{x}}^-,\ \hat{\mathbf{x}},\ P_{11}, \ldots P_{15},\ P_{21}, \ldots, P_{55}]^T \tag{18}$$

   The files vec2mat.m and mat2vec.m can be used to convert P^- from a matrix to a vector and vice versa.