

NTNU

TMR4243

---

## Case C - Internal Dynamics

---

Authors:

Shiv Jeet RAI  
Eirik ØSTEBY  
Even ØDEGAARD  
Jan Olav ØKSNES



# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Theory from individual work</b>	<b>3</b>
2.1	Control Plant Model . . . . .	3
2.1.1	Configuration and Workspace . . . . .	3
2.1.2	Fully actuated or underactuated? . . . . .	3
2.2	Rotation Matrix Properties . . . . .	3
2.2.1	Derivative of the rotation matrices . . . . .	3
2.2.2	Skew Symmetric property . . . . .	4
2.2.3	Nullity / NULL SPACE . . . . .	4
2.3	Guidance . . . . .	4
2.3.1	Coefficients . . . . .	4
2.3.2	Position, velocity and acceleration . . . . .	4
2.4	Use case description . . . . .	5
2.5	Motion Control . . . . .	5
2.6	Simulink implementation and simulation . . . . .	5
2.6.1	Lever arm (0,0) . . . . .	6
2.6.2	Lever arm (-0.5,0) . . . . .	7
2.6.3	Lever arm (0.5,0) . . . . .	8
<b>3</b>	<b>Processor-in-the-loop simulation</b>	<b>9</b>
3.1	Motivation . . . . .	9
3.2	Overview of the system . . . . .	9
3.3	Graphical user interface . . . . .	10
3.4	Observer gains . . . . .	10
3.5	PD controller gains . . . . .	10
3.6	Simulations . . . . .	10
3.6.1	Lever arm (0,0) . . . . .	11
3.6.2	Lever arm (-0.5,0) . . . . .	13
3.6.3	Lever arm (0.5,0) . . . . .	15
<b>4</b>	<b>Scale model test</b>	<b>17</b>
4.1	Overview of the system . . . . .	17
4.2	Graphical user interface . . . . .	18
4.3	Observer gains . . . . .	18
4.4	PD controller gains . . . . .	18
4.5	Simulations . . . . .	18
<b>5</b>	<b>Conclusion</b>	<b>21</b>
<b>Appendices</b>		<b>21</b>
<b>Appendix A Nonlinear Passive Observer</b>		<b>22</b>
A.1	Observer Model . . . . .	22

# 1 Introduction

This exercise investigates the pose dynamics of CS Enterprise I (CSEI) when only the position is controlled, the heading is not controlled. The rotational behaviour of the vessel is studied for different points within the vessel hull used as control variable. The derivations in this exercise is based on the CSEI mathematical model in the Marine Cybernetics Laboratory handbook. Instead of controlling the customary pose  $\eta = [x, y, \psi]^T$ , the controlled output  $p_c$  is the vessel position offset by a lever arm configurable to any desired position in the vessel,

$$p_c = \begin{bmatrix} x_c \\ y_c \end{bmatrix} = p + R_2(\psi)l, \quad (1)$$

where

- $p = \begin{bmatrix} x \\ y \end{bmatrix}$  is the position in the basin frame,
- $R_2(\psi)$  is the two dimensional rotation matrix, and
- $l = \begin{bmatrix} l_x \\ l_y \end{bmatrix}$  is the lever arm in the body frame.

The lever arm tip  $p_c(t)$  should track a time-varying desired position  $p_d(t) = \begin{bmatrix} p_{d,x}(t) \\ p_{d,y}(t) \end{bmatrix}$  which moves along a straight line path. This yields a control law for  $\tau_{uv}$  ensuring

$$\lim_{t \rightarrow \infty} |p_c(t) - p_d(t)| = 0 \quad (2)$$

In the assignment text, full state feedback is assumed available, but as the end of this assignment includes a scale model test, we have included a nonlinear observer in our model.

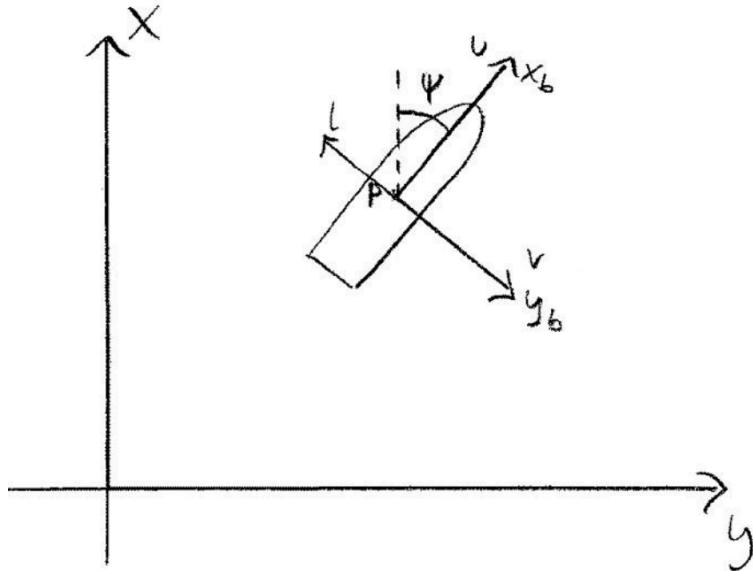


Figure 1: CSEI in basin

A PD control law,

$$\tau_{uv} = -K_p R_2(\psi)^T (p_c - p_d(t)) - K_d (\nu_2 - R_2^T(\psi)[\dot{p}_d(t) - R_2(\psi)S_2(r)l]) \quad (3)$$

is defined, and the system is implemented in SIMULINK, as a model consisting of the following modules:

- Guidance system
- Motion controller
- CSEI dynamics

The system was simulated and tuned with several different lever arm configurations, before Processor-in-the-loop (PIL) simulations were performed. After additional tuning on the PIL-simulations, we finally proceeded to perform a scale model test, using the CSEI in the Marine Cybernetics Laboratory, testing scenarios corresponding to the PIL-simulation.

Notice that we have chosen to comment the plots from the different simulation tasks in the end of each section, as the plots are quite repetitive and self-explanatory.

## 2 Theory from individual work

### 2.1 Control Plant Model

#### 2.1.1 Configuration and Workspace

Since the motion of the vessel is in three degrees of freedom, the configuration space of the vessel is 3. Namely:  $[x \ y \ \psi]^T$

Given by the task description, we know that  $\psi$  is not controllable, thus the workspace is 2. Namely:  $[x \ y]^T$

#### 2.1.2 Fully actuated or underactuated?

Since the degree of the workspace is larger than the configuration space, the craft is underactuated.

### 2.2 Rotation Matrix Properties

Given by the task:

$$S_2(r) = \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix} \quad S_3(r) = \begin{bmatrix} S_2(r) & 0 \\ 0 & 0 \end{bmatrix} \quad R_2(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \quad R_3(\psi) = \begin{bmatrix} R_2(\psi) & 0 \\ 0 & 1 \end{bmatrix}$$

#### 2.2.1 Derivative of the rotation matrices

$$R_2(\psi)S_2(r) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix} = \begin{bmatrix} -r\sin(\psi) & -r\cos(\psi) \\ r\cos(\psi) & -r\sin(\psi) \end{bmatrix}$$

$$\dot{R}_2 = \frac{d\psi}{dt}R_2 = \dot{\psi} \begin{bmatrix} -\sin(\psi) & -\cos(\psi) \\ \cos(\psi) & -\sin(\psi) \end{bmatrix} = r \begin{bmatrix} -\sin(\psi) & -\cos(\psi) \\ \cos(\psi) & -\sin(\psi) \end{bmatrix} = \underline{\underline{R_2(\psi)S_2(r)}}$$

$$R_3(\psi)S_3(r) = \begin{bmatrix} R_2(\psi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} S_2(r) & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R_2(\psi)S_2(r) & 0 \\ 0 & 0 \end{bmatrix}$$

$$\dot{R}_3 = \frac{d\psi}{dt}R_3 = \begin{bmatrix} \dot{R}_2(\psi) & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R_2(\psi)S_2(r) & 0 \\ 0 & 0 \end{bmatrix} = \underline{\underline{R_3(\psi)S_3(r)}}$$

### 2.2.2 Skew Symmetric property

For a matrix to be skew symmetric, the following must hold:  $R^T = -R$

$$S_2^T(\psi) = \begin{bmatrix} 0 & r \\ -r & 0 \end{bmatrix} = \underline{\underline{-S_2(r)}}$$

$$S_3^T(\psi) = \begin{bmatrix} S_2^T & 0 \\ 0 & 0 \end{bmatrix} = \underline{\underline{-S_3(r)}}$$

### 2.2.3 Nullity / NULL SPACE

$$v^T S_2(r)v = [v_1 \ v_2] \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -v_2r \\ v_1r \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = -v_1v_2r + v_1v_2r = \underline{\underline{0}}$$

$$v^T S_3(r)v = [v_1 \ v_2 \ v_3] \begin{bmatrix} S_2(r) & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = [v_1 \ v_2] S_2(r) \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \underline{\underline{0}}$$

## 2.3 Guidance

Given:

$$p_d(s) = \begin{bmatrix} a_x s & b_x \\ a_y s & b_y \end{bmatrix} \quad \text{with} \quad p_d(0) = p_{d,0} = (x_0, y_0)^T \quad p_d(1) = p_{d,1} = (x_1, y_1)^T$$

$$|\dot{p}_d| = \sqrt{a_x^2 \dot{s}^2 + a_y^2 \dot{s}^2} = |c| \sqrt{a_x^2 + a_y^2} = |u_d|$$

### 2.3.1 Coefficients

The coefficients were found as:

$$a_x = x_1 - x_0 \tag{4}$$

$$b_x = x_0 \tag{5}$$

$$a_y = y_1 - y_0 \tag{6}$$

$$b_y = y_0 \tag{7}$$

$$c = \frac{u_d}{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}} \tag{8}$$

### 2.3.2 Position, velocity and acceleration

Since  $c$  is constant, will  $t^* = ct$  just be a scaling.

$$p_d(ct) = \begin{bmatrix} a_x c t + b_x \\ a_y c t + b_y \end{bmatrix} \tag{9}$$

$$\dot{p}_d(ct) = \begin{bmatrix} a_x c \\ a_y c \end{bmatrix} \tag{10}$$

$$\ddot{p}_d(ct) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{11}$$

## 2.4 Use case description

- The steps the guidance system perform before actuating the vessel are finding the coefficients by an intial position and a desired speed.
- The guidance must set the reference  $p_d(t)$  to be equal to  $p_{d,1}$ . An easy way to do this is to set the coefficient c to be zero when  $p_{d,1}$  is reached. This method will also ensure the velocity given by the guidance system to be zero.

## 2.5 Motion Control

A PD control law was given by the task description.

The task wanted us to derive the velocity  $\dot{p}_c$  from  $p_c = p + R_2(\psi)l$  in terms of  $\nu_2$

$$\dot{p}_c = \dot{p} + \underline{\dot{p} R_2(\psi)l} = \underline{\underline{R_2^T(\psi)\nu_2 + R_2(\psi)S_2(r)l}}$$

The equation above can also be written, if  $p_c$  is defined as  $p_d(t)$ :

$$\nu_2 = \underline{\underline{R_2^T(\psi)[\dot{p}_d(t) - R_2(\psi)S_2(r)l]}}$$

## 2.6 Simulink implementation and simulation

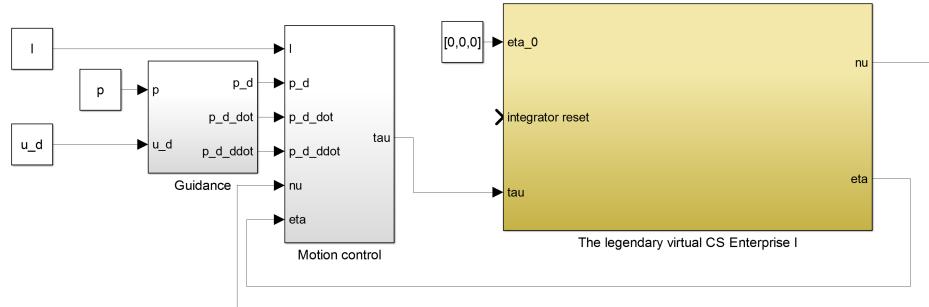


Figure 2: Overview

$$K_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad K_d = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.7 \end{bmatrix}$$

$$P_{d,0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad P_{d,1} = \begin{bmatrix} 8 \\ 6 \end{bmatrix} \quad U_d = 0.2m/s$$

### 2.6.1 Lever arm (0,0)

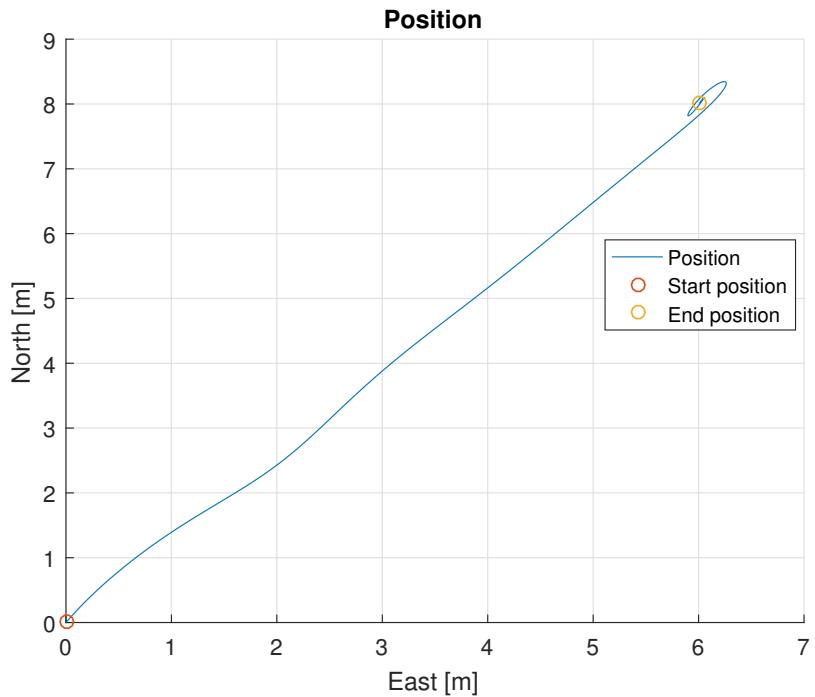


Figure 3: Position plot

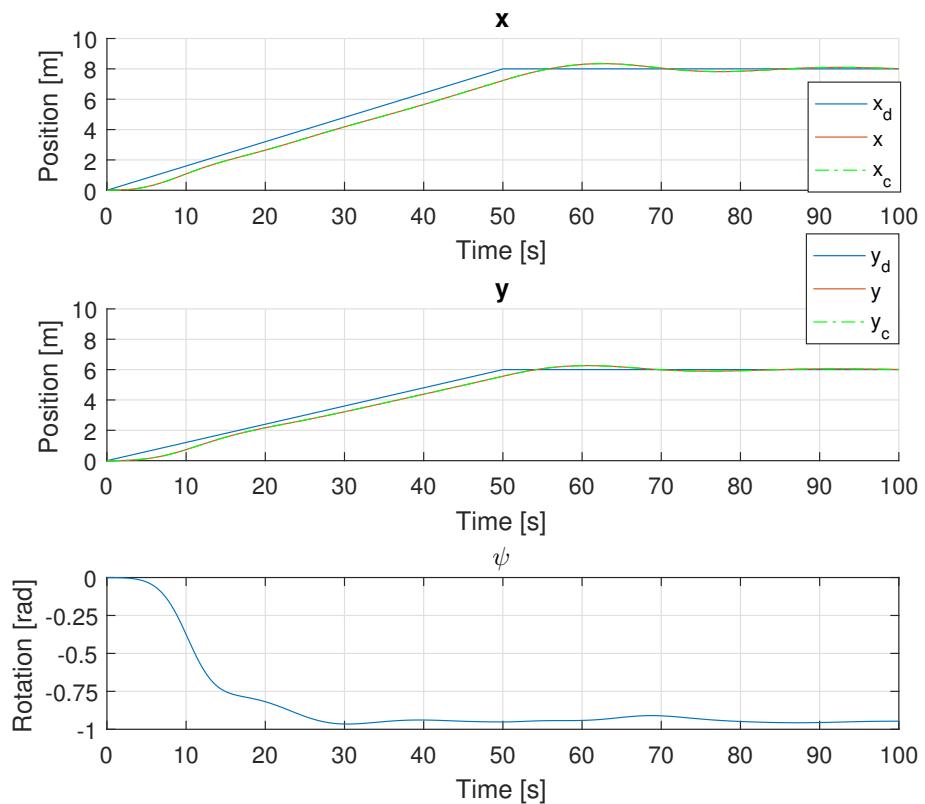


Figure 4: Eta plot

### 2.6.2 Lever arm (-0.5,0)

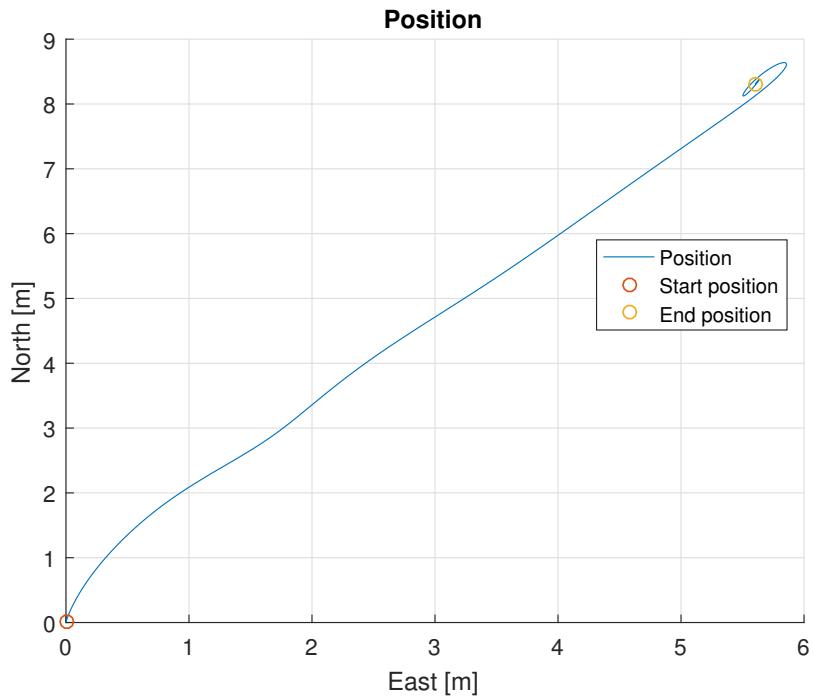


Figure 5: Position plot

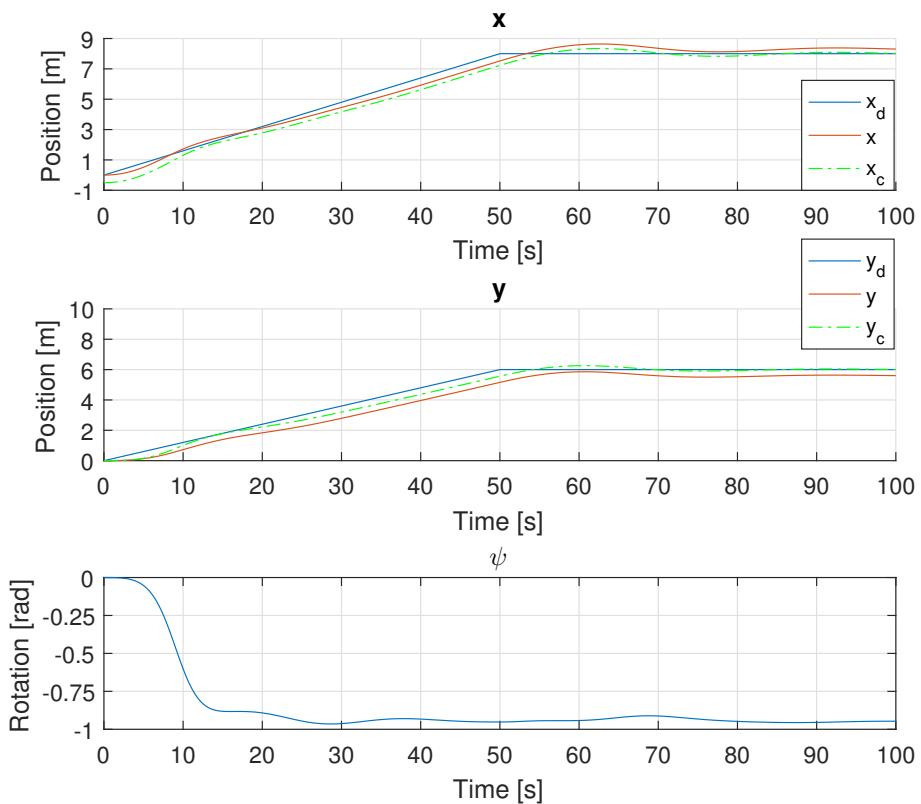


Figure 6: Eta plot

### 2.6.3 Lever arm (0.5,0)

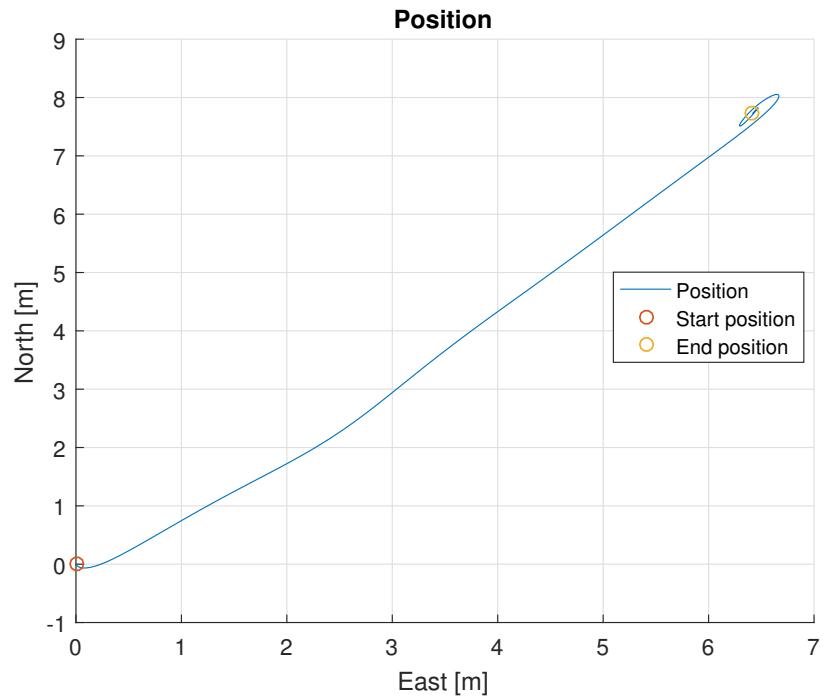


Figure 7: Position plot

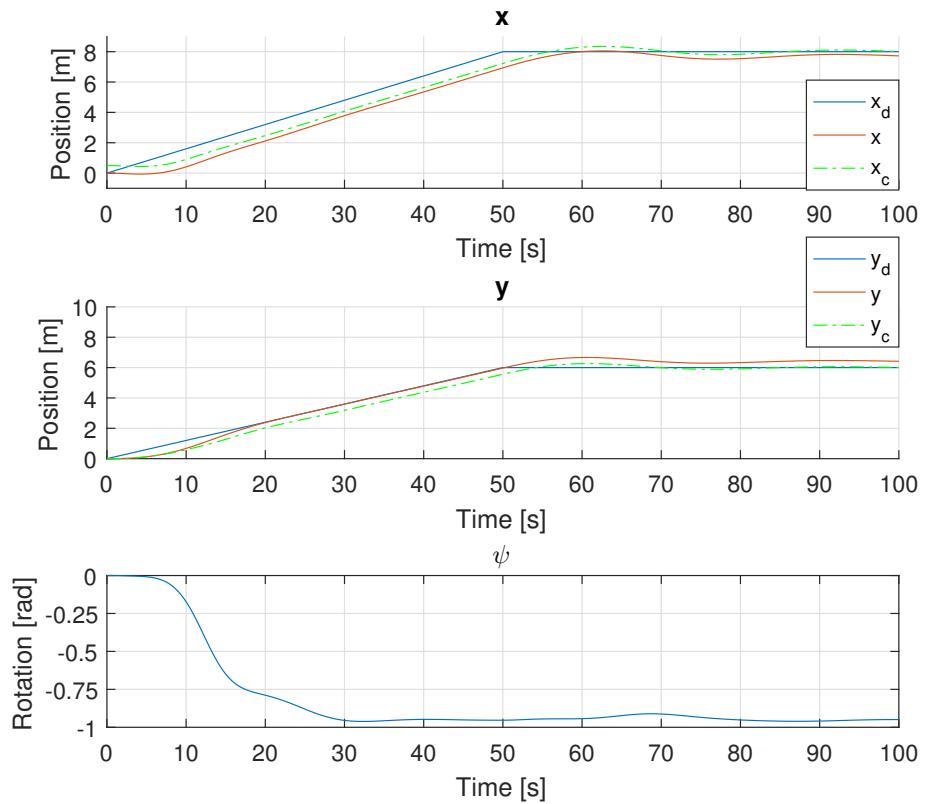


Figure 8: Eta plot

**Comment:** Notice that saturation on the thrusters have been used. The reason is to have simulations with thruster-output as we will have from the scale model test. This will make the system perform slower than needed in Simulink and PIL simulations, but will ease the transition to scale model-simulations.

All three lever arm configurations are reached smoothly, and only have a tiny overshoot. Since we're not controlling the yaw angle, the ship direction rapidly drifts to the left, and stabilizes at  $\psi = -55^\circ$ . Then the ship moves orthogonal to the path, only giving trust in sway.

### 3 Processor-in-the-loop simulation

#### 3.1 Motivation

Before testing our implementation of the guidance system and the motion control that used a control plant model of the CSE1 enterprise, we tested it in a processor-in-the-loop simulator (P-I-L) by using CRIO. The reasons for doing that were:

- In a P-I-L simulation the controller is deployed on representative microprocessor, problems such as insufficient computing resources or related to real time simulation can be exposed.
- Developing a suitable user interface for later testing in the basin, namely scale model test.

Instead of the control plant model a generated outputfile was used, namely CSE1\_tau.out. The file was precompiled from Simulink and had the same outputs and inputs as the control plant model. We made the neccessary adjustments in our Simulink file replacing the desired inputs and outputs with corresponding Veristand input/output blocks. The file was builded, and both our outputfile and the CSE1\_tau.out was added to Veristand. The neccessary mapping was done.

#### 3.2 Overview of the system

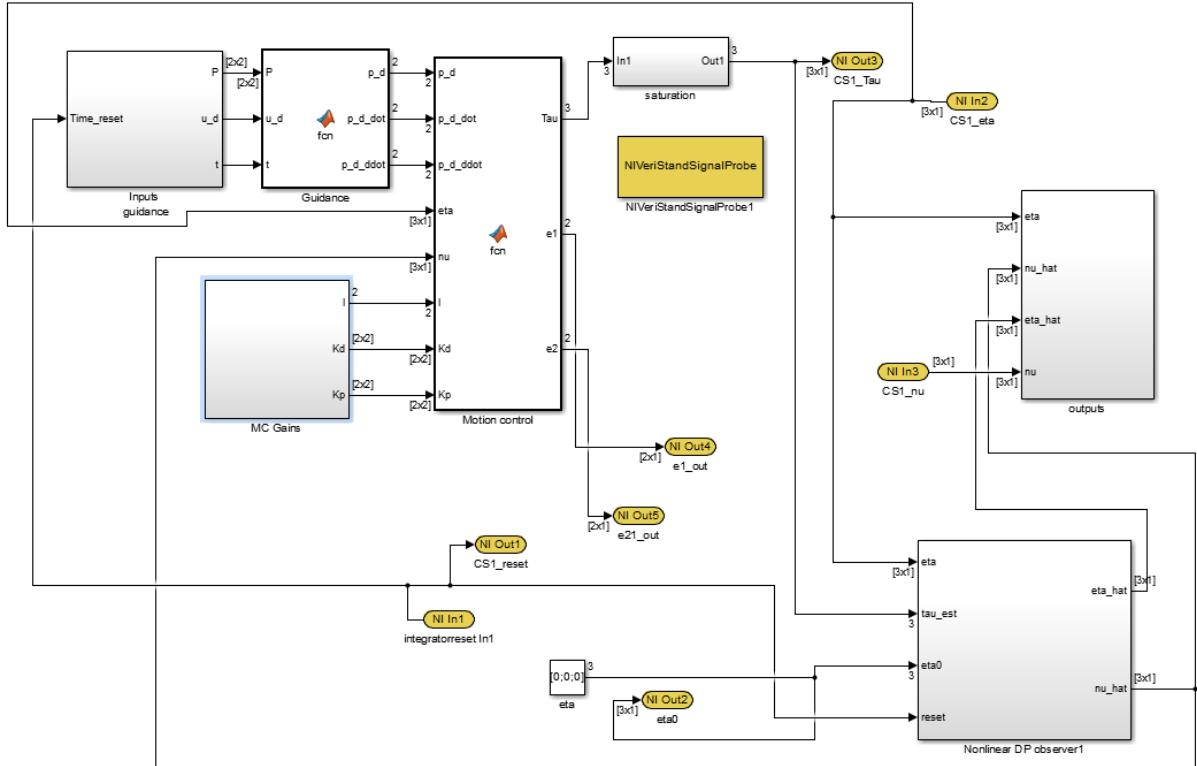


Figure 9: Simulink overview

### 3.3 Graphical user interface



Figure 10: Graphical user interface

### 3.4 Observer gains

The observer gains for the Nonlinear Passive Observer was chosen as:

$$K_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad K_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad K_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (12)$$

#### Reason:

Since the bias is supposed to counteract the environmental forces, and we in P-I-L didn't assumed any, the  $K_3$  was set to 0. Due to not having waves, the wave frequency,  $\omega_c$  was chosen by try and error.  $\omega_c$  was chosen as 1 and gave  $K_2$  as the identity matrix.  $K_4$  was set to zero as well. See the appendix for further details about the observer.

### 3.5 PD controller gains

The controller gains for the PD control law (equation 3) used in PIL:

$$K_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad K_d = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.7 \end{bmatrix} \quad (13)$$

### 3.6 Simulations

As specified by the task, we ran the same simulations as in subsection 2.6 on page 5. The plots are seen below:

$$P_{d,0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad P_{d,1} = \begin{bmatrix} 8 \\ 6 \end{bmatrix} \quad U_d = 0.2m/s$$

### 3.6.1 Lever arm (0,0)

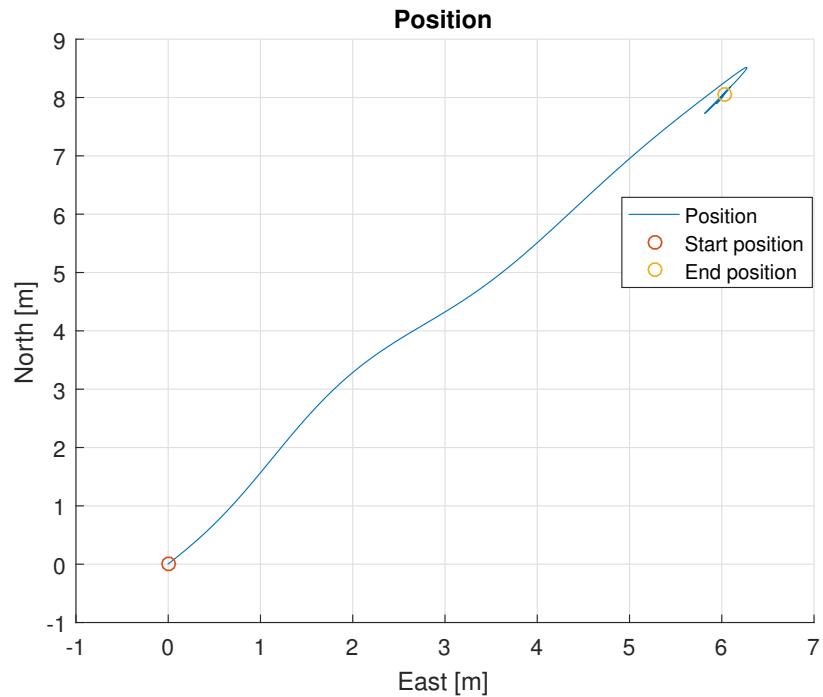


Figure 11: Position plot

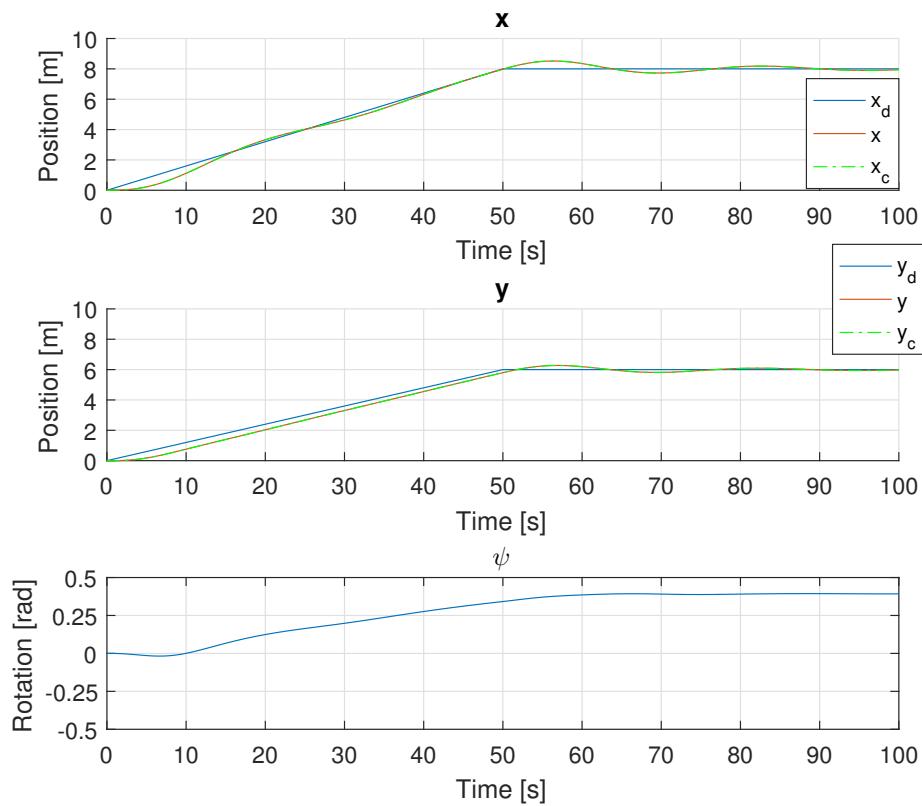


Figure 12: Eta plot

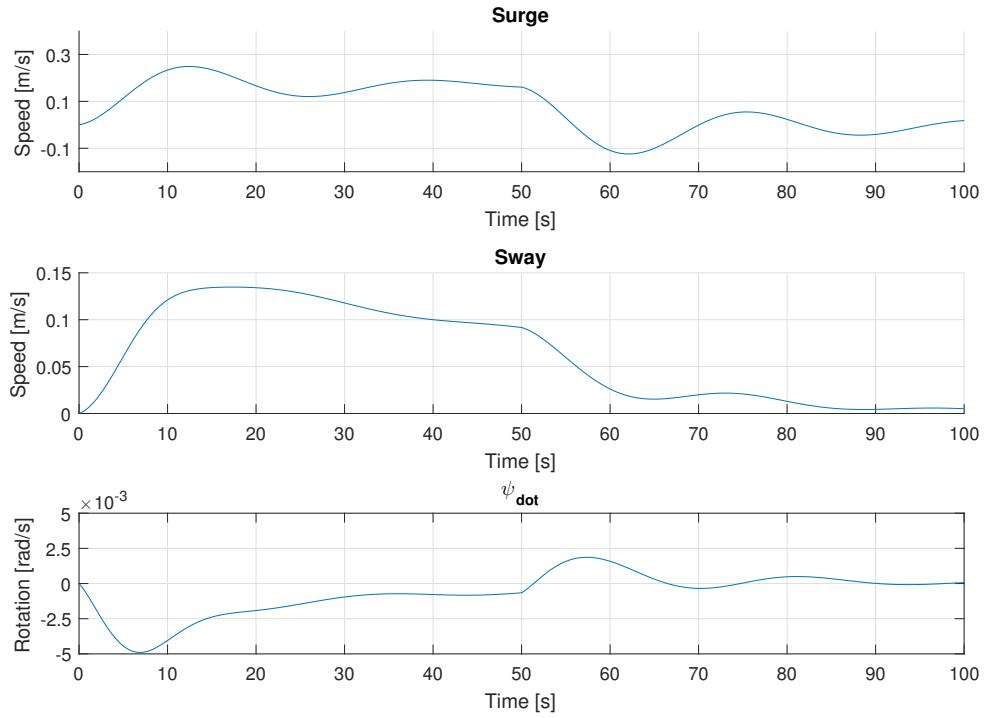


Figure 13: Nu plot

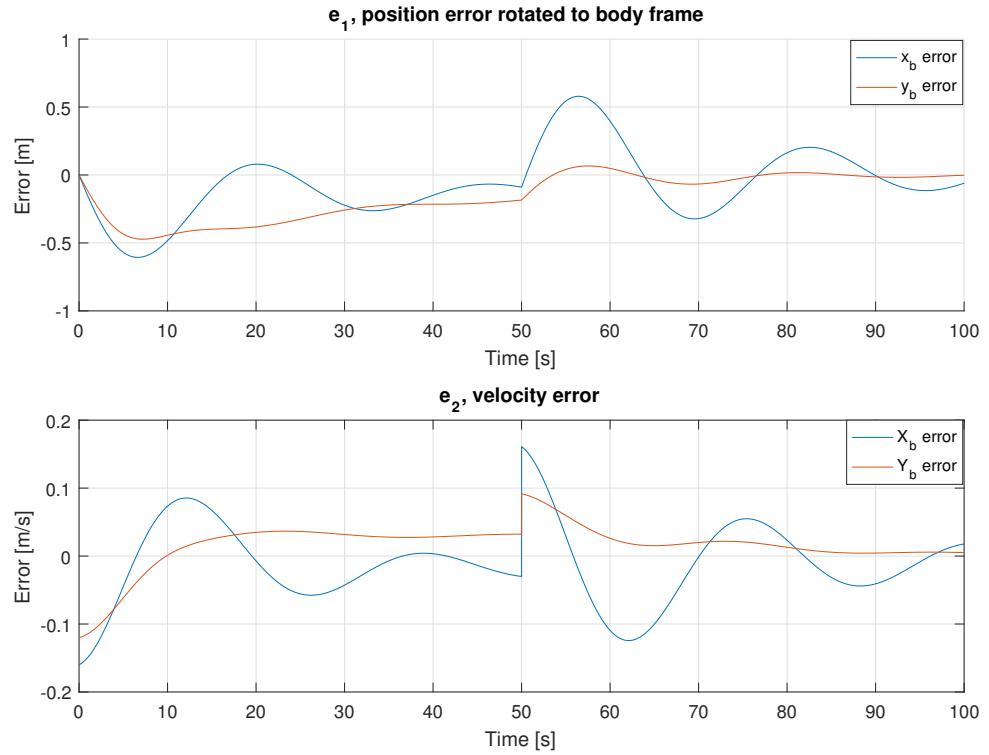


Figure 14: Error plot

**Results:** Since there is no lever arm,  $p_c = \eta[1 : 2]$ , and the ship converges to  $\eta[1 : 2] = p_d$ .

### 3.6.2 Lever arm (-0.5,0)

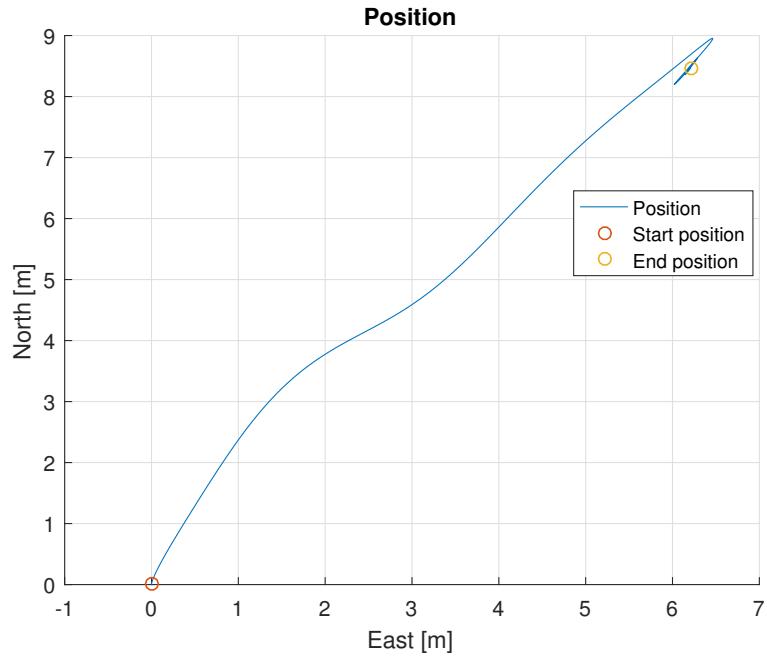


Figure 15: Position plot

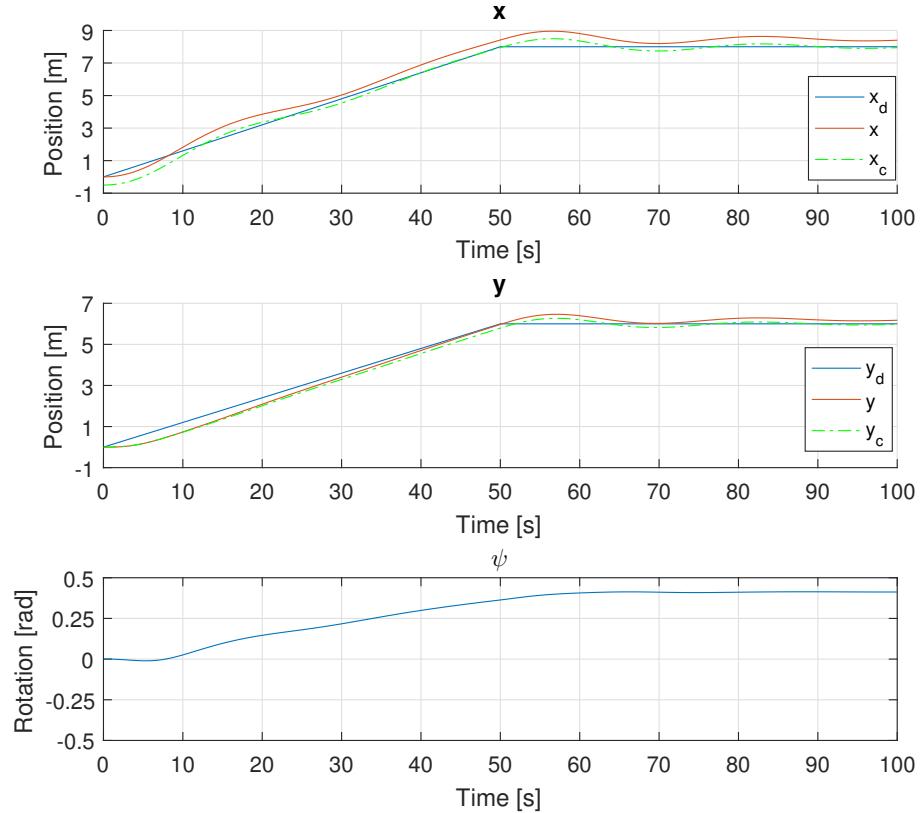


Figure 16: Eta plot

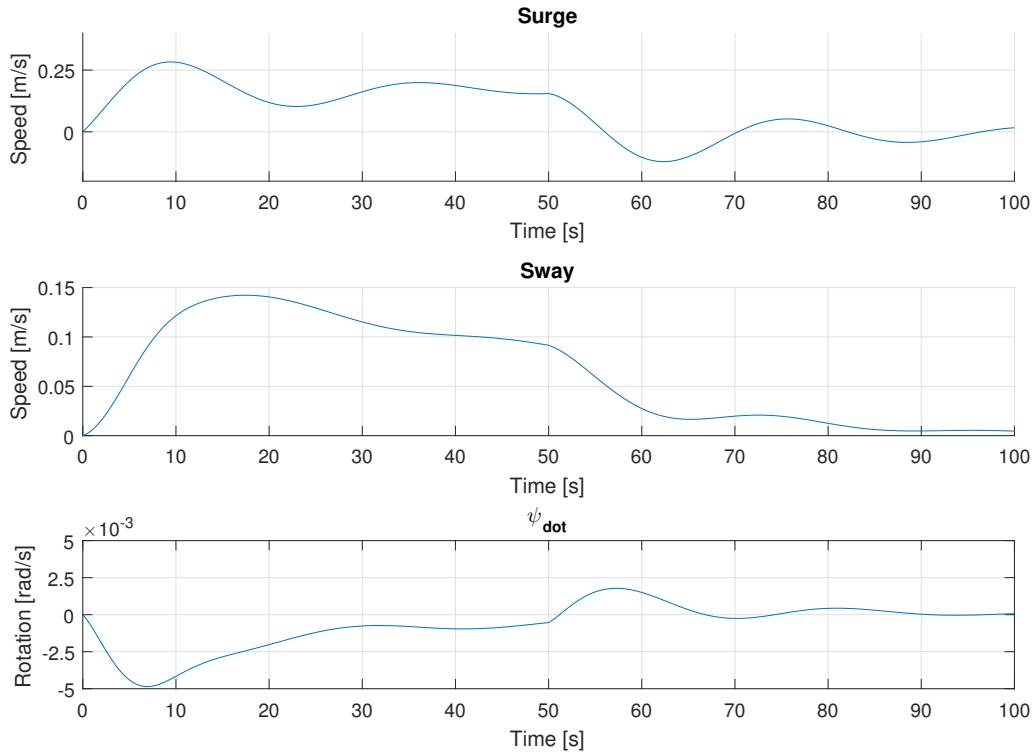


Figure 17: Nu plot

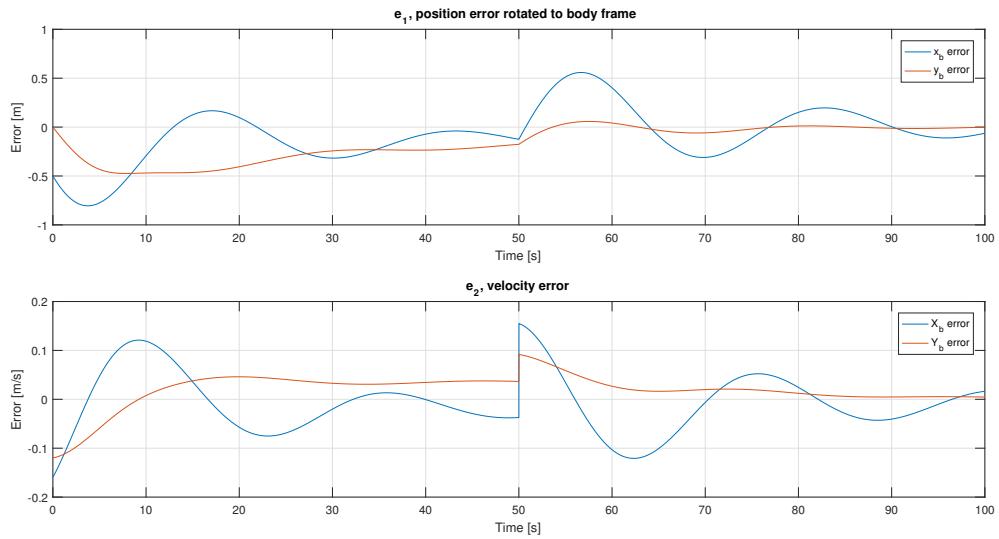


Figure 18: Error plot

**Results:** The ship converges to  $\eta[1 : 2] = [8.47, 6.21]$ . This makes sense, since the lever arm moves the point to control behind the middle of the ship.  $p_c = [8, 6]$  at the converged value of  $\psi$ , and the control objective is accomplished.

### 3.6.3 Lever arm (0.5,0)

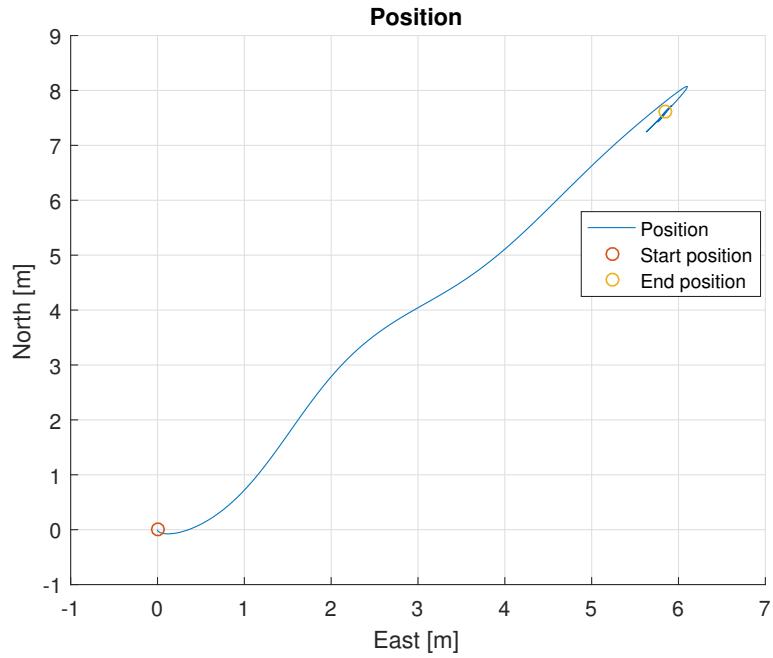


Figure 19: Position plot

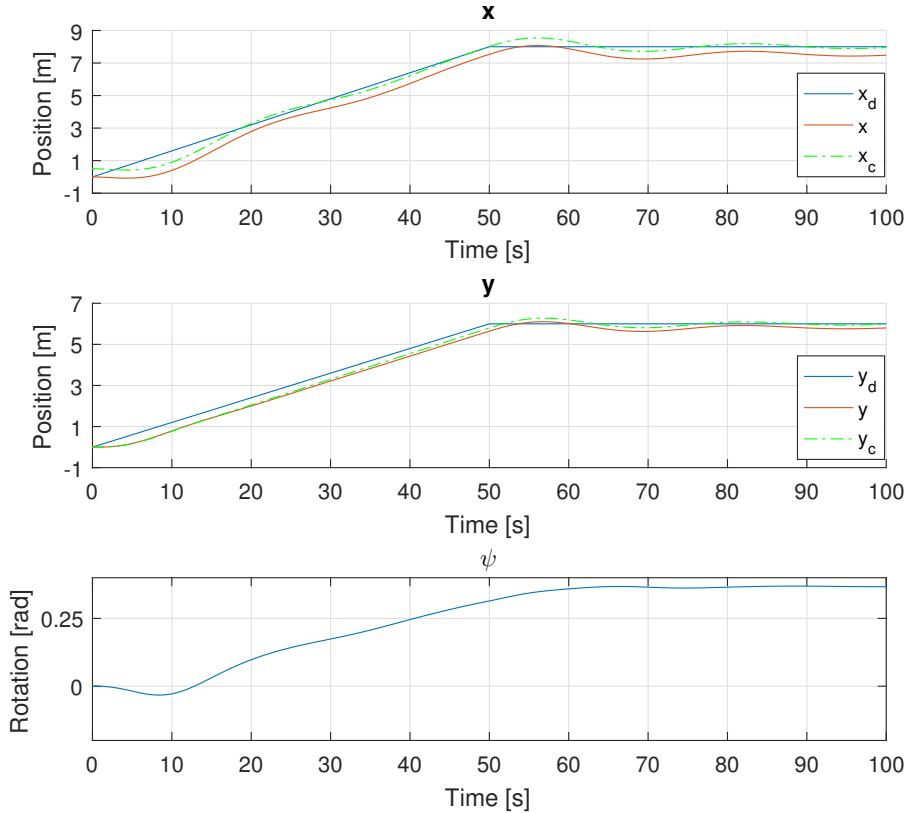


Figure 20: Eta plot

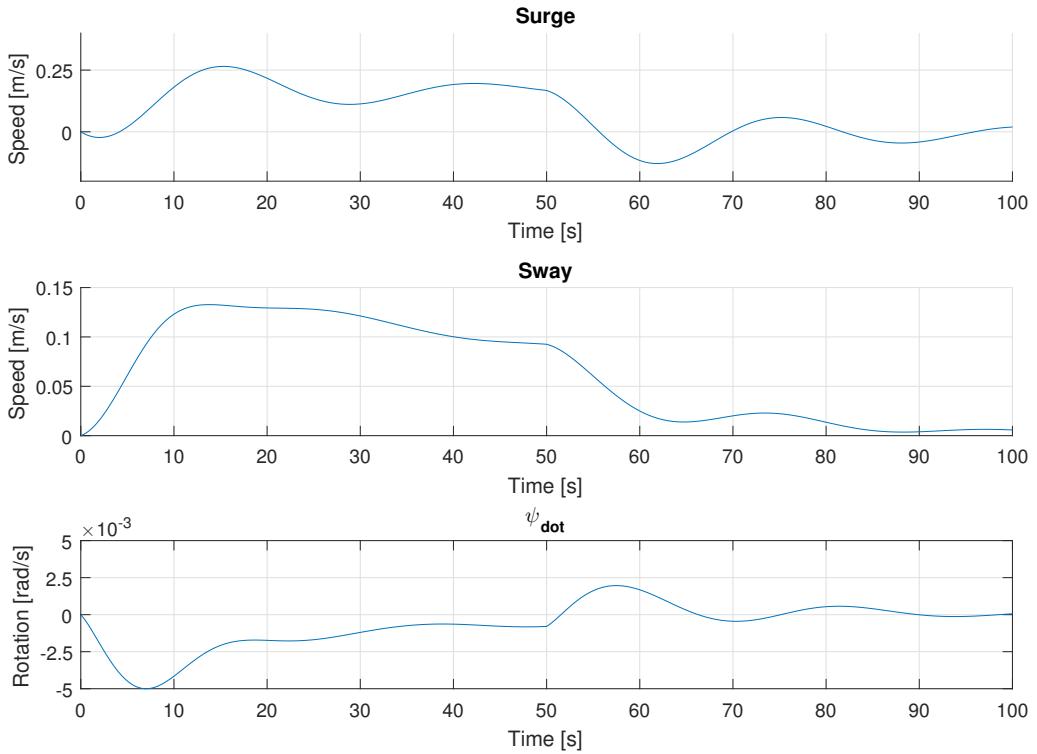


Figure 21: Nu plot

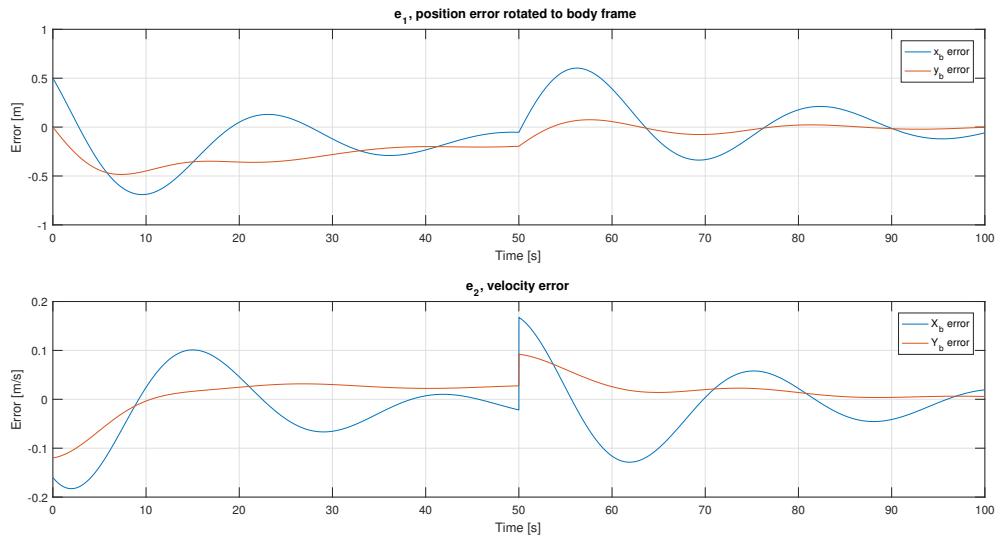


Figure 22: Error plot

**Results:** The ship converges to  $\eta[1 : 2] = [7.60, 5.85]$ . The lever arm moves the point to control in front of the middle of the ship.  $p_c = [8, 6]$  at the converged value of  $\psi$ , and the control objective is accomplished.

**Comment:** The results are very similar to the Simulink simulation, except that the yaw angle turns the other way, stabilizing at  $\psi = 23^\circ$ , driving parallel to the path. Since the only difference from the previous simulation model is the non-linear passive observer, we're led to

believe that the "black box" model we implemented has more sophisticated dynamics, than the Simulink model. Mainly including the effects from the resistance in the water.

## 4 Scale model test

In the scale model test we wanted to see if we can obtain good position control in a more realistic environment than in the Simulink and PIL simulations. In contrast to the PIL simulations, where the observer worked as a predictor, we now need to tune the observer so it can remove noise from the measurements and find good estimates of  $\eta$  and  $\nu$  for the controller.

### 4.1 Overview of the system

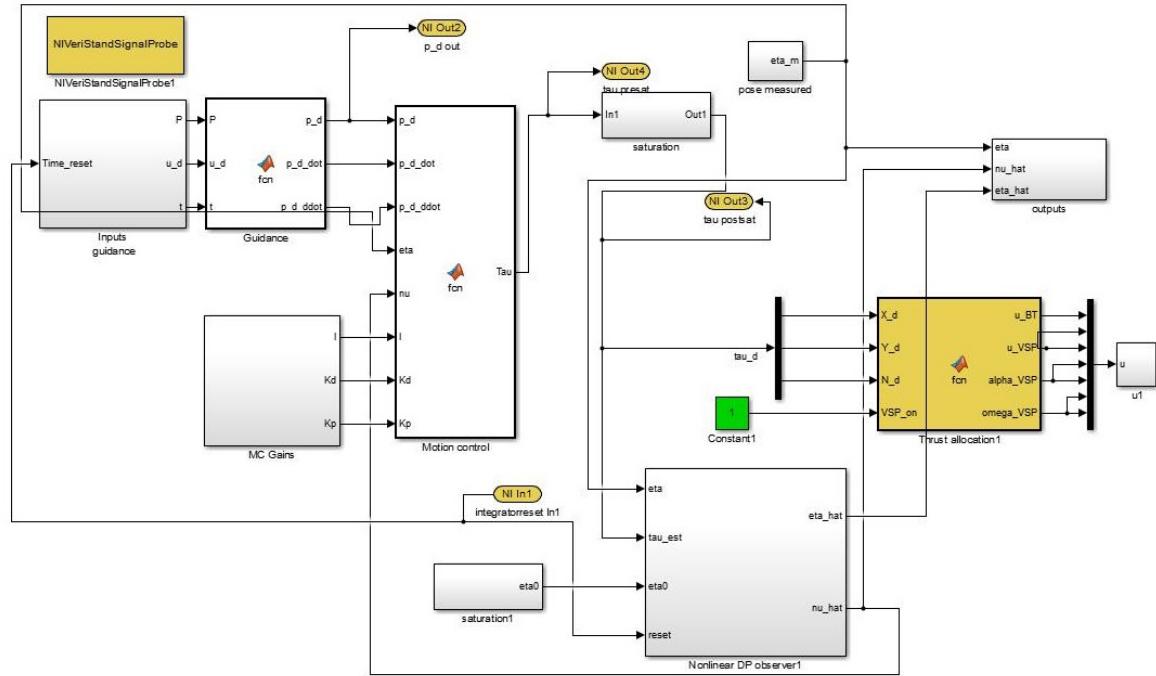


Figure 23: Overview of the system

## 4.2 Graphical user interface

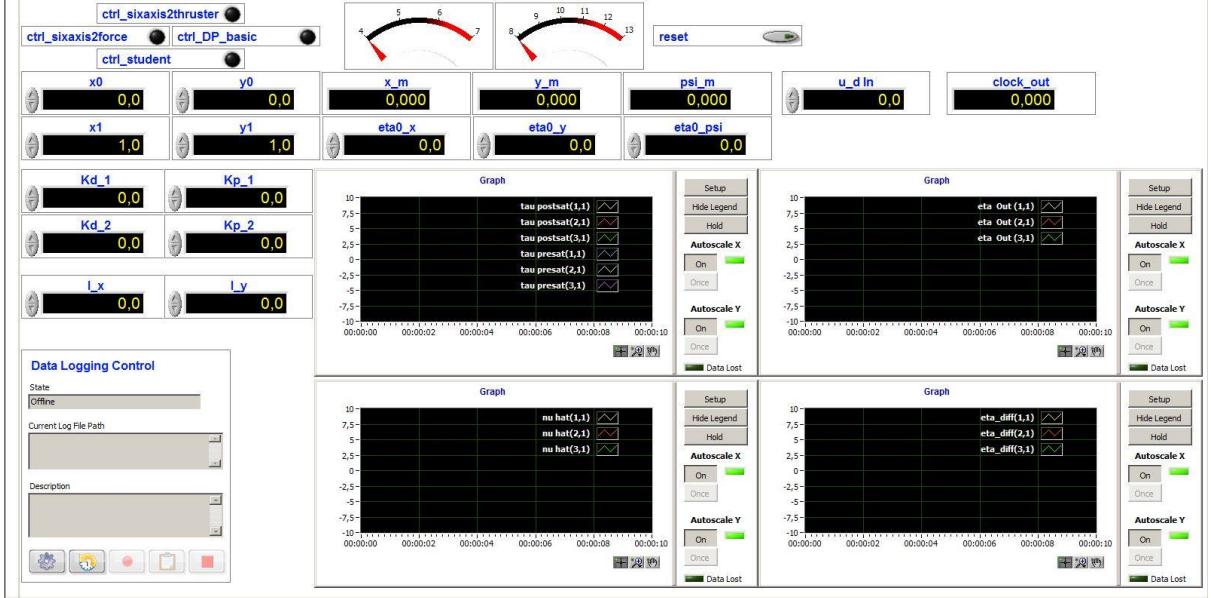


Figure 24: Veristand workspace for scale model test

## 4.3 Observer gains

$$K_2 = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad K_3 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} \quad K_4 = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad (14)$$

### Reason:

During the simulation in the basin we managed to break one of the motors which was temporarily fixed by using tape. This, in addition to the currents in the basin, gave bias which had to be compensated for. Therefore, we chose the matrices as in equation 14. See the appendix for further detail about the observer.

## 4.4 PD controller gains

The controller gains for the PD control law (equation 3) used in the scale model test :

$$K_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad K_d = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.7 \end{bmatrix} \quad (15)$$

## 4.5 Simulations

$$P_{d,0} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \quad P_{d,1} = \begin{bmatrix} 4 \\ 1 \end{bmatrix} \quad U_d = 0.2m/s$$

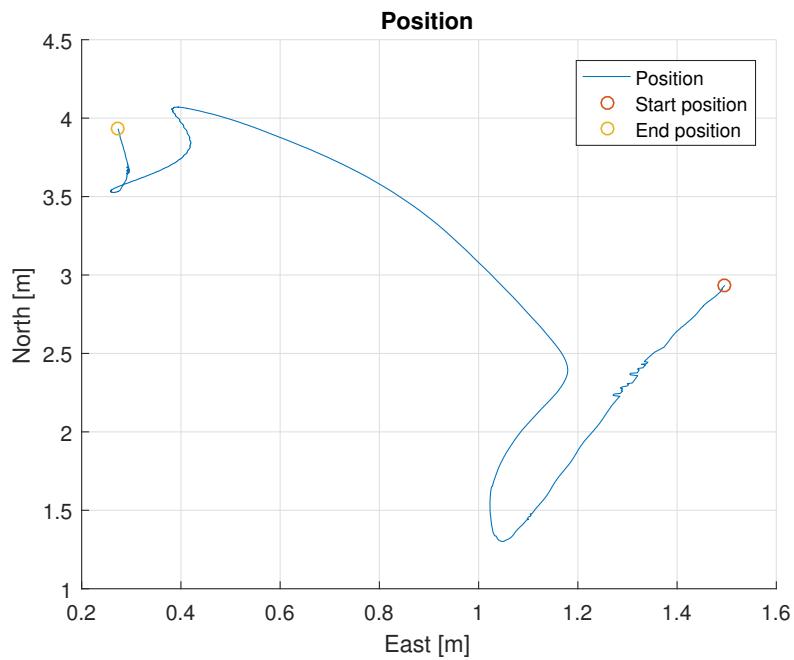


Figure 25: Position plot

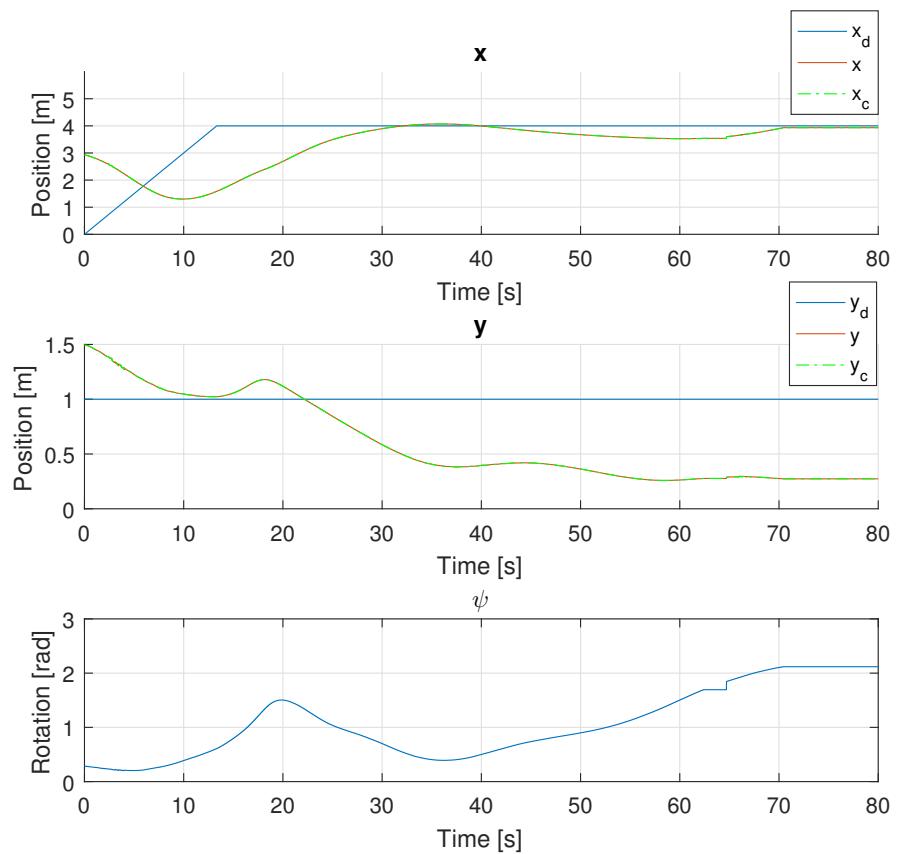


Figure 26: Eta plot

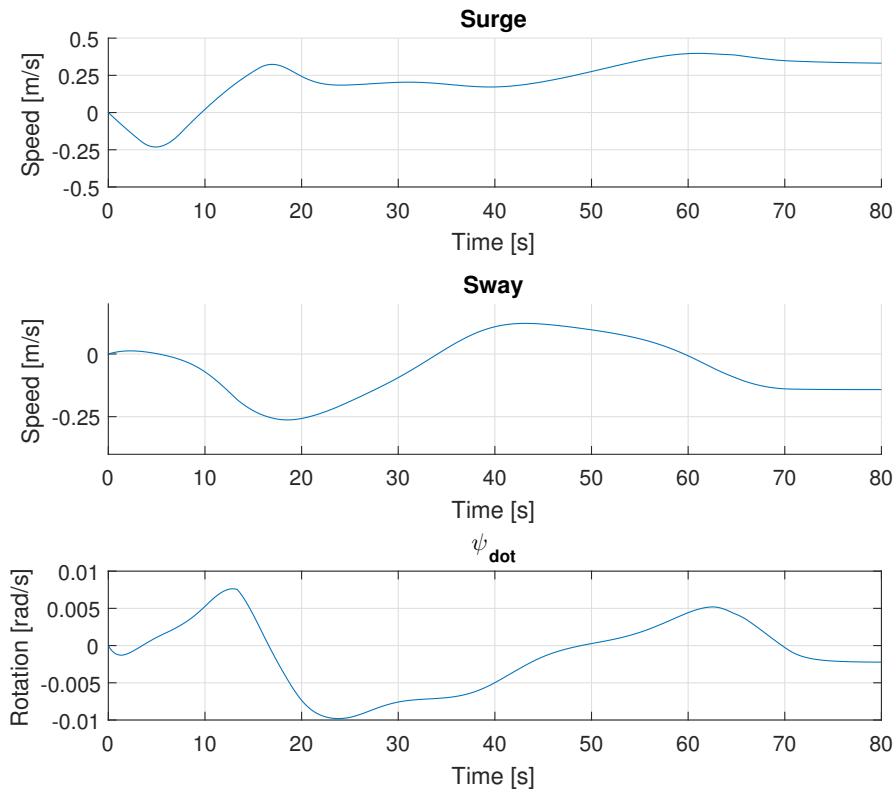


Figure 27: Nu plot

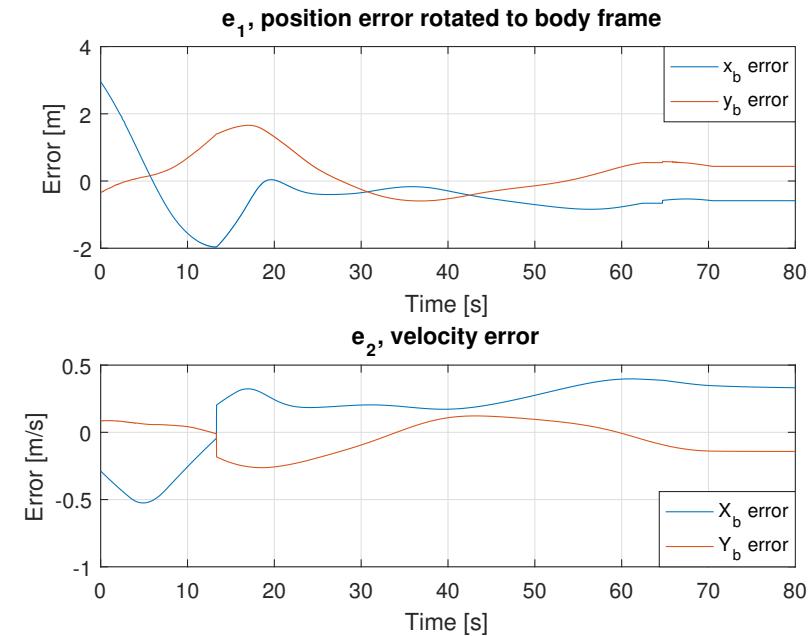


Figure 28: Error plot

**Comment:** The plots above are from one of the test runs. Unfortunately they don't make much sense, but are still included to show our results from the basin. As we can see from figure 26 the x-value seem to converge, but y drifts away with no sign of recovery. Why this seemed

to happen can be explained by several reasons. We had severe problems in the scale model lab, both with our model, and the lab equipment. When we investigated the model afterwards, one loose connection was found, making  $x_{d,0} = 0$ , regardless of the value set in the Veristand GUI. This means that the start of the desired path was outside of the pool bounds, and resulted in an unnaturally big step at the start of the simulation, since the boat was far from the start of the path. Another issue was that we kept losing the vessel position during the lab time. In addition a motor broke, but this just caused an additional bias to be compensated for.

## 5 Conclusion

We have in this lab investigated the pose dynamics of CSE1 when only the position is controlled, with three different points within the vessel hull used as controlled variable. In addition, one nonlinear observer is implemented. In the offline simulation we obtained a heading converging to  $\psi \approx -0.95$  after some time, and the position controller working properly, with only a small overshoot and little oscillation. The results from the PIL-testing on the other hand were not similar to this. Here, we obtained convergence in  $\psi \approx 0.4$ . The difference here is due to the use of the generated output-file instead of the control plant model. In the scale model testing, we obtained more varying results, mainly because we were not able to perform a successful test-run. The data collected are somewhat inconsistent, but interpreting it in the light of our results from the PIL-testing and offline simulation, we can still observe some interesting results. We seem to have some convergence in the positioning, but the pose dynamics are inconclusive at best, and lacks consistency.

Our main focus in the scale model testing was to ensure that the ship stayed within the valid areas for some time, to be able to obtain results on the pose dynamics of the CSE1. Unfortunately, we experienced some technical problems in the lab, and as a result of this, the focus on the conclusion is pointed towards the results obtained in the PIL-testing.

In the PIL-testing we experienced that both our guidance system, motion controller and nonlinear observer performed as expected. For all lever arm configurations tested, we obtained an accurate motion controller, and an uncontrolled yaw, slowly changing. We observe that the convergence in yaw does not really happen before  $\nu = [0, 0]^T$ , which is quite natural, as the thrust given is equal to zero when the ship acquires the desired position. Looking more closely at the plots of  $\psi$ , we see a small drop as the ship starts moving, which is followed by a slow positive change. If we imagine the CSE1 to be perfectly symmetric about the  $x_b$ -axis, it is fair to assume that the ships heading is slowly converging towards the ships bearing, as one would expect the same behaviour for an opposite case, where  $\psi$  would have a slow negative change. We also see that the convergence of  $\psi$  is faster for lever arm configurations with higher  $l_x$ , and vice versa. Again, this is natural, as the lever arm configuration changes the point in the vessel hull used as a control variable. As the control point is moved forward, the dynamics of the ships motion is slightly changed, and this would result in a faster convergence to the point of least resistance. If we imagine that the ships control point was located in the center of the bow, it becomes clear that the ship would have a yaw-angle directly opposite to the external forces, and that the further back the control point is placed, the slower the convergence would be.

Even though the results from the scale model testing are inconclusive, it is pretty clear that we don't get the same behaviour. In some cases one can observe tendencies to convergence, but with little or none consistency regarding what angle it converges to. In figure 26 we can see that the CSE1 never acquires a constant bearing, which makes it hard to make any conclusions regarding the yaw-behaviour, as the expected behaviour is constantly changing.

Overall, the work put down in this project has not given the desired results. With less problems and more time in the Marine Cybernetics laboratory, we would have been able to obtain more interesting results.

## Appendix A Nonlinear Passive Observer

### A.1 Observer Model

To design a Nonlinear Passive Observer for an DP system the following system can be used as presented by prof. Thor I. Fossen:

$$\dot{\hat{\eta}} = R(\psi)\hat{\nu} + K_2\tilde{y} \quad (16)$$

$$\dot{\hat{b}} = K_3\tilde{y} \quad (17)$$

$$M\dot{\hat{\nu}} = -D\hat{\nu} + R^T(\psi) + \tau + \tau_{wind} + K_4\tilde{y} \quad (18)$$

$$\hat{y} = \hat{\eta} \quad (19)$$

where  $\tilde{y} = y - \hat{y}$  and  $K_{2-4}$  are observer gain matrices.

In order to use the Nonlinear Passive observer the the observer gains have to be chosen such that stability is achieved. In order to do that, they can be chosen such that they satisfy the Kalman-Yakubovich-Popov (KYP) lemma.

Thus the matrices:

$$K_2 = \text{diag}\{K_{21}, K_{22}, K_{23}\} \quad (20)$$

$$K_3 = \text{diag}\{K_{31}, K_{32}, K_{33}\} \quad (21)$$

$$K_4 = \text{diag}\{K_{41}, K_{42}, K_{43}\} \quad (22)$$

had to satisfy:

$$K_{2i} = \omega_{ci} \quad (23)$$

$$\frac{1}{T_i} \ll \frac{K_{3i}}{K_{4i}} \leq \omega_{oi} \leq \omega_{ci} \quad (24)$$

where  $\omega_{ci}$  is the cut-off frequency of the wave,  $\omega_{oi}$  is the peak frequency of the wave and  $T_i$  is the time constant.

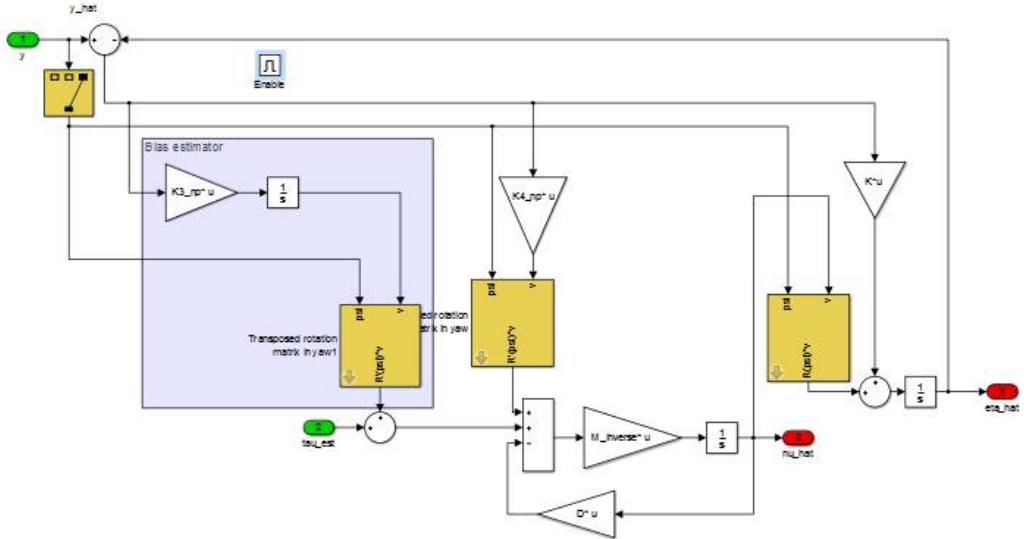


Figure 29: Nonlinear Passive Filter