TTK4190 Guidance and Control of Vehicles

Assignment 1

Written Fall 2019 By Name

Problem 1 - Attitude Control of Satellite

The equations of motion for a satellite is given by

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{T}_q(\mathbf{q})\boldsymbol{\omega} \\ \mathbf{I}_{CG}\dot{\boldsymbol{\omega}} - \mathbf{S}(\mathbf{I}_{CG}\boldsymbol{\omega})\boldsymbol{\omega} &= \boldsymbol{\tau} \end{aligned} \tag{1}$$

with inertia matrix $\mathbf{I}_{CG} = mr^2\mathbf{I_3}$, mass $m = 180$ kg and $r = 2.0$ m. The quaternions $\mathbf{q} = [\eta, \epsilon_1, \epsilon_2, \epsilon_3]^T$ and the angular velocities $\omega = [p, q, r]^T$ denotes the states of the system.

Problem 1.1 - Equilibrium and linearization

The EOM in eq. (1) can be expanded to the following form given by eq. (2) and eq. (3).

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} -\epsilon_1 & -\epsilon_2 & -\epsilon_3 \\ \eta & -\epsilon_3 & \epsilon_2 \\ \epsilon_3 & \eta & -\epsilon_1 \\ -\epsilon_2 & \epsilon_1 & \eta \end{bmatrix}\begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2}$$

$$\dot{\omega} = \begin{bmatrix} \frac{1}{mr^2} & 0 & 0 \\ 0 & \frac{1}{mr^2} & 0 \\ 0 & 0 & \frac{1}{mr^2} \end{bmatrix}\left(\begin{bmatrix} \tau_p \\ \tau_q \\ \tau_r \end{bmatrix} + mr^2\begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}\begin{bmatrix} p \\ q \\ r \end{bmatrix}\right) \tag{3}$$

where we have used the inversion rule for a the squared matrix $\mathbf{I}_{CG}$ with only diagonal entries and the skew symmetric operator $S(\cdot)$

Furthermore, we write this on the component form given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\big(x_4\sqrt{1-x_1^2-x_2^2-x_3^2} - x_5x_3 + x_6x_2\big) \\ \frac{1}{2}\big(x_4x_3 + x_5\sqrt{1-x_1^2-x_2^2-x_3^2} - x_6x_1\big) \\ \frac{1}{2}\big(-x_4x_2 + x_5x_1 + x_6\sqrt{1-x_1^2-x_2^2-x_3^2}\big) \\ \frac{1}{720}u_1 \\ \frac{1}{720}u_2 \\ \frac{1}{720}u_2 \end{bmatrix} \tag{4}$$

where we have used that $\mathbf{x} = \begin{bmatrix} \epsilon^T & \omega^T \end{bmatrix}$, $\mathbf{u} = \begin{bmatrix} \tau_1 & \tau_2 & \tau_3 \end{bmatrix}$, the fact that $\eta = \sqrt{1 - \epsilon^{\mathbf{T}}\epsilon}$ and the numerical values for m and r.

To find the equilibrium of the system we set $\dot{\mathbf{x}} = 0$ and find that the equilibrium $\mathbf{x}_0$ is given by

$$\mathbf{x}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{5}$$

which is clear from eq. (4)

Now we would like to linearize the system to get the system dynamics shown in eq. (4) on the form $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$. This is done by calculating the Jacobian of the system dynamics with respect to the states and the inputs and then evaluating the Jacobian at the equilibrium x_0. The calculation of $\mathbf{A}$ and $\mathbf{B}$ is shown bellow:

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\bigg|_{x_0} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$$\mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\bigg|_{x_0} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{720} & 0 & 0 \\ 0 & \frac{1}{720} & 0 \\ 0 & 0 & \frac{1}{720} \end{bmatrix} \tag{7}$$

which yields the complete linearized system about x_0 as shown in eq. (8)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{720} & 0 & 0 \\ 0 & \frac{1}{720} & 0 \\ 0 & 0 & \frac{1}{720} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \tag{8}$$

Problem 1.2

A regular PD controller can be implemented using the control law as stated in eq. (14).

$$\tau = -\mathbf{K}_d \boldsymbol{\omega} - k_p \boldsymbol{\epsilon} \tag{9}$$

If we apply this controller to the linearized system in eq. (8) we get the following closed loop system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{720} & 0 & 0 \\ 0 & \frac{1}{720} & 0 \\ 0 & 0 & \frac{1}{720} \end{bmatrix} \begin{bmatrix} -k_d x_4 - k_p x_1 \\ -k_d x_5 - k_p x_2 \\ -k_3 x_6 - k_p x_3 \end{bmatrix} \tag{10}$$

which simplifies to the following:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \\ \frac{-k_p}{720} & 0 & 0 & \frac{-k_d}{720} & 0 & 0 \\ 0 & \frac{-k_p}{720} & 0 & 0 & \frac{-k_d}{720} & 0 \\ 0 & 0 & \frac{-k_p}{720} & 0 & 0 & \frac{-k_d}{720} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \tag{11}$$

If we set $k_p = 2$ and $k_d = 40$ and denote the closed loop system matrix $\mathbf{A}$ we can compute the eigenvalues of the system by utilizing the Matlab command *eig(A)*. This yields the following closed loop eigenvalues:

$$\lambda = \begin{bmatrix} -0.0278 \pm 0.0248i & -0.0278 \pm 0.0248i & -0.0278 \pm 0.0248i \end{bmatrix} \tag{12}$$

The eigenvalues shown in eq. (12) shows that the system is stable as all the eigenvalues has a strict negative real part. There is also an imaginary part that can contribute to oscillatory behaviour. Thus we would expect the system response to show oscillatory behaviour before converging to the origin.

For this application, a critically damped system is desirable as we would like the system states to behave calmly without unnecessary oscillations to save fuel. Fuel is a very limited and expensive resource for the satellite. This means that the desired closed loop eigenvalues should be purely real and negative.

Problem 1.3

Simulating the closed loop system with the PD controller in eq. (11) with controller gains $k_p = 2$, $k_d = 40$ and initial conditions $\phi(0) = -5°$, $\theta(0) = 10°$, $\psi(0) = -20°$ and initial angular velocities of zero in Matlab yields the following response:

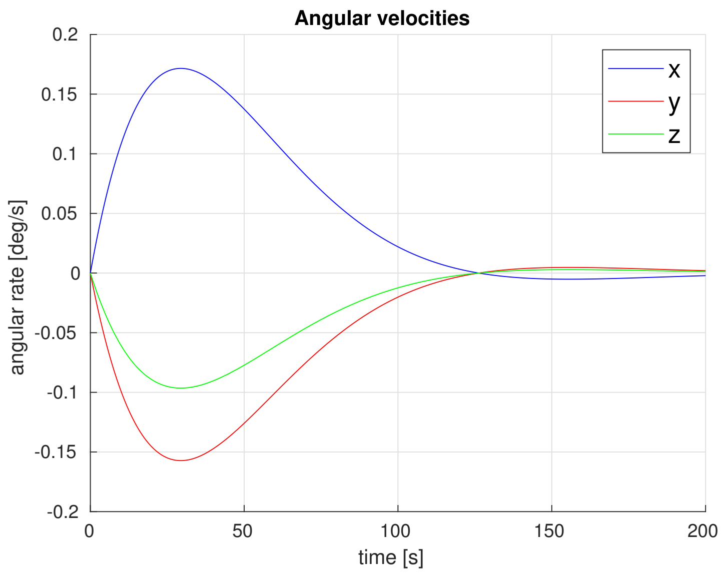

Figure 1: The angular velocities of the satellite.

From fig. 1, fig. 2 and fig. 3 we can see that the PD controller successfully controls the vehicle to the origin. The responses verifies the assumption of oscillatory behaviour in the start and eventually the system converges to the origin. One should also note that the initial conditions is somewhat far from the origin, so the eigenvalues found when linearizing about the origin might be different from those in the initial conditions.

To follow a non-zero *constant* reference trajectory one would like to add a desired trajectory to the control law, so that we can adjust where we would like to place the equilibrium of the closed loop system. One naive way to do this is the following control law:

$$\tau = -\mathbf{K}_d \boldsymbol{\omega} - k_p(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_d) \tag{13}$$

However, the last term of the control law needs further investigation as subtraction of quaternions is not as straight forward as thought. This shall we investigated in the following problems.

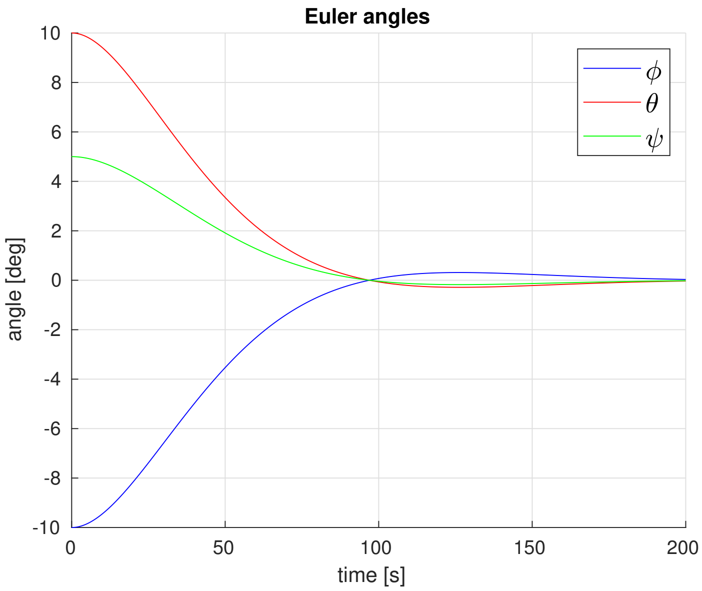

Figure 2: The euler angles of the satellite.

Problem 1.4

A modified version of the PD controller is shown bellow:

$$\tau = -\mathbf{K}_d\boldsymbol{\omega} - k_p\tilde{\boldsymbol{\epsilon}} \tag{14}$$

Here we add the imaginary part of the error quaternion to the control law $\tilde{\boldsymbol{\epsilon}}$ as proposed in the previous problem. The error quaternion $\tilde{\mathbf{q}}$ of the quaternions $\mathbf{q}_d$ and $\mathbf{q}$ is denoted as

$$\tilde{\mathbf{q}} = \bar{\mathbf{q}}_d \otimes \mathbf{q} \tag{15}$$

where $\bar{\mathbf{q}}$ denotes the conjugate of the quaternion $\mathbf{q}$. Thus the conjugate quaternion is given by

$$\bar{\mathbf{q}} = \begin{bmatrix}\boldsymbol{\eta} & -\boldsymbol{\epsilon}^T\end{bmatrix}^T \tag{16}$$

The quaternion product is defined as

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = \begin{bmatrix}\boldsymbol{\eta}_1\boldsymbol{\eta}_2 - \boldsymbol{\epsilon}_1^T\boldsymbol{\epsilon}_2 \\ \boldsymbol{\eta}_1\boldsymbol{\epsilon}_2 + \boldsymbol{\eta}_2\boldsymbol{\epsilon}_1 + \mathbf{S}(\boldsymbol{\epsilon}_1)\boldsymbol{\epsilon}_2\end{bmatrix} \tag{17}$$

Thus, the error quaternion $\tilde{\mathbf{q}}$ becomes

$$\tilde{\mathbf{q}} = \begin{bmatrix}\eta_d \\ -\boldsymbol{\epsilon}_d\end{bmatrix} \otimes \begin{bmatrix}\eta \\ \boldsymbol{\epsilon}\end{bmatrix} \tag{18}$$

using eq. (17) the component form of eq. (18) becomes

$$\begin{bmatrix}\tilde{\eta} \\ \tilde{\epsilon}_1 \\ \tilde{\epsilon}_2 \\ \tilde{\epsilon}_3\end{bmatrix} = \begin{bmatrix}\eta_d\eta + \epsilon_{1d}\epsilon_1 + \epsilon_{2d}\epsilon_2 + \epsilon_{3d}\epsilon_3 \\ \eta_d\epsilon_1 - \eta\epsilon_{1d} + \epsilon_{3d}\epsilon_2 - \epsilon_{2d}\epsilon_3 \\ \eta_d\epsilon_2 - \eta\epsilon_{2d} - \epsilon_{3d}\epsilon_1 + \epsilon_{1d}\epsilon_3 \\ \eta_d\epsilon_3 - \eta\epsilon_{3d} + \epsilon_{2d}\epsilon_1 - \epsilon_{1d}\epsilon_2\end{bmatrix} \tag{19}$$

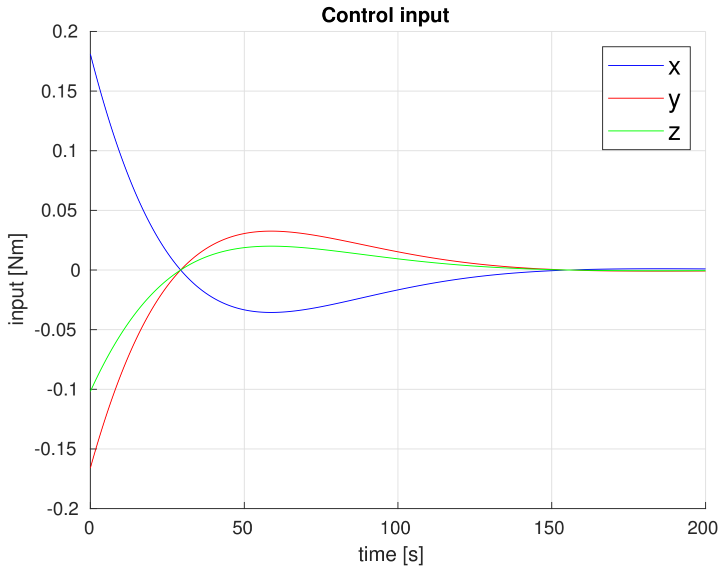

Figure 3: The control inputs of the satellite.

Now to see if the component form of eq. (20) is correctly computed, we insert the convergence criterion $\mathbf{q} = \mathbf{q}_d$

$$\begin{bmatrix} \tilde{\eta} \\ \tilde{\epsilon}_1 \\ \tilde{\epsilon}_2 \\ \tilde{\epsilon}_3 \end{bmatrix} = \begin{bmatrix} \eta\eta + \epsilon_1\epsilon_1 + \epsilon_2\epsilon_2 + \epsilon_3\epsilon_3 \\ \eta\epsilon_1 - \eta\epsilon_1 + \epsilon_3\epsilon_2 - \epsilon_2\epsilon_3 \\ \eta\epsilon_2 - \eta\epsilon_2 - \epsilon_3\epsilon_1 + \epsilon_1\epsilon_3 \\ \eta\epsilon_3 - \eta\epsilon_3 + \epsilon_2\epsilon_1 - \epsilon_1\epsilon_2 \end{bmatrix} \tag{20}$$

which becomes

$$\begin{bmatrix} \tilde{\eta} \\ \tilde{\epsilon}_1 \\ \tilde{\epsilon}_2 \\ \tilde{\epsilon}_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{21}$$

The error quaternion in eq. (21) is what we often call the identity quaternion. The identity quaternion indicates no rotation. In our case the error quaternion gives a measure of the difference of the actual and desired orientation of the satellite. This means that there are no difference in the desired and actual orientation of the satellite when we let $\mathbf{q}_d = \mathbf{q}$.

Problem 1.5

In this problem, the system is simulated with the control law given by eq. (14), with $k_p = 20$ and $K_d = 400$. The reference is time-varying. The response is shown in fig. 4, fig. 6 and fig. 7. The tracking error is shown in fig. 5. The tracking error plot shows how well each state manages to follow the reference. With the current controller, the satellite does not manage to keep up with the input reference dynamic. The higher frequency input reference θ_d gets damped more compared to ψ_d. This gives an indication that the system behaves like a low pass filter. The swinging behaviour is somehow expected as the equations of motion looks similar to a mass-damper system.

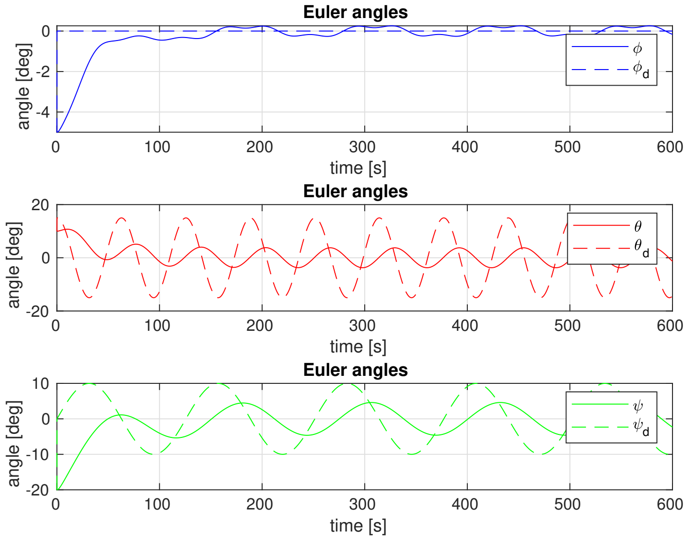

Figure 4: The Euler angles of the satellite with a PD controller and Euler angle error.

Problem 1.6

In this section, the PD controller incorporates the angular velocity as well ... The control law in this problem can be written as

$$\boldsymbol{\tau} = -\mathbf{K}_d\tilde{\boldsymbol{\omega}} - k_p\tilde{\boldsymbol{\epsilon}} \tag{22}$$

and the desired angular velocity as

$$\boldsymbol{\omega}_d = \mathbf{T}^{-1}_{\Theta_d}(\Theta_d)\dot{\Theta}_d \tag{23}$$

where

$$\mathbf{T}^{-1}_{\Theta_d} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{24}$$

and

$$\dot{\Theta}_d = \begin{bmatrix} 0 \\ -1.5\sin 0.1t \\ 0.5\cos 0.05t \end{bmatrix} \tag{25}$$

The response is shown in fig. 8, fig. 10 and fig. 11. The tracking error is shown in fig. 9. As it can be observed, the system now manages to follow the non-constant input references. The worst tracking error is in angle θ and it's peak-to-peak error is around 3 degrees, which is acceptable with regards to performance and from control point of view. This tracking performance is expected as now, the controller includes the angle rate error as well. This makes sense as in the previous controller, ω is included in the controller, which works against the error state dynamics, and makes the system states oscillate. Including the correct $\tilde{\omega} = \omega - \omega_d$ in the controller fixes this issue.

Problem 1.7

In this section the following Lyapunov function

$$V = \frac{1}{2}\tilde{\boldsymbol{\omega}}^\top \mathbf{I}_{CG}\tilde{\boldsymbol{\omega}} + 2k_p(1-\tilde{\eta}) \tag{26}$$

will be proven to satisfy the Lyapunov stability criteria (Barbalat's Lemma), which are:

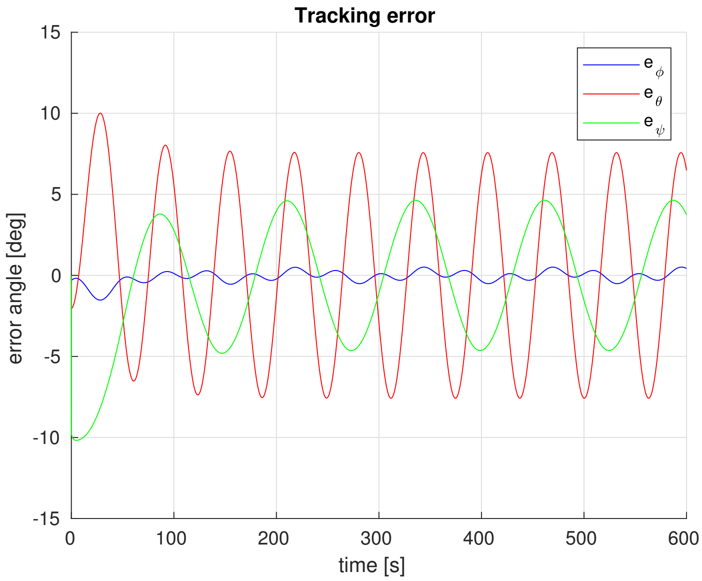

Figure 5: The tracking error of the PD controller with Euler angle error.

1. $V(x) > 0$ for $\tilde{\boldsymbol{\omega}} \neq 0$
2. $\dot{V}(x) < 0$ for $\tilde{\boldsymbol{\omega}} \neq 0$
3. $\dot{V}(x)$ is uniformly continuous

First V will be shown to be positive. We will first analyze the term $2k_p(1-\tilde{\eta})$ in eq. (26). Since $\tilde{\eta} \in (-1, 1)$, this $2k_p(1-\tilde{\eta}) \in (0, 4k_p), k_p > 0$. $\frac{1}{2}\tilde{\boldsymbol{\omega}}^\top \mathbf{I}_{CG}\tilde{\boldsymbol{\omega}} > 0$ for $\tilde{\boldsymbol{\omega}} \neq 0$ since $\mathbf{I}_{CG} > 0$. This leads to $V > 0$ for $\tilde{\boldsymbol{\omega}} \neq 0$.

V is radially unbounded as well since the term $\frac{1}{2}\tilde{\boldsymbol{\omega}}^\top \mathbf{I}_{CG}\tilde{\boldsymbol{\omega}}$ tends to infinity as ω goes to infinity.

Now we want to show that $\dot{V}(x) < 0$, using that $\dot{\tilde{\boldsymbol{\omega}}} = \mathbf{I}_{CG}^{-1}\tau$, and $\dot{\tilde{\eta}} = \frac{1}{2}\tilde{\epsilon}^T\omega$.

$$\dot{V} = \tilde{\boldsymbol{\omega}}^\top \mathbf{I}_{CG}\dot{\tilde{\boldsymbol{\omega}}} - 2k_p(\dot{\tilde{\eta}}) \tag{27}$$

Inserting for $\dot{\tilde{\omega}}$ and $\dot{\tilde{\eta}}$ and $\tau = -K_d\omega - k_p\tilde{\epsilon}$ gives

$$\dot{V} = -K_d\tilde{\omega}^T\omega \tag{28}$$

which is negative definite for $\omega \neq 0$.

V satisfies the Lyapunov criteria, and thereby the equilibrium of the closed-loop system is asymptotically stable. The system is only locally asymptotically stable since $V > 0, \dot{V} < 0$ and not $V \leq 0, \dot{V} \geq 0$ for $\omega \neq 0$. In order to be globally asymptotically stable, there should not exist more than one equilibrium point.

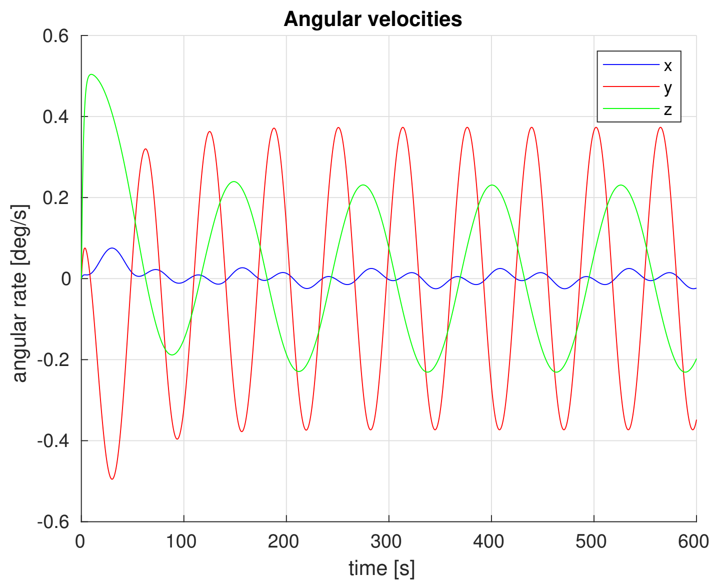

Figure 6: The Euler angles velocities of the satellite with a PD controller and Euler angle error.

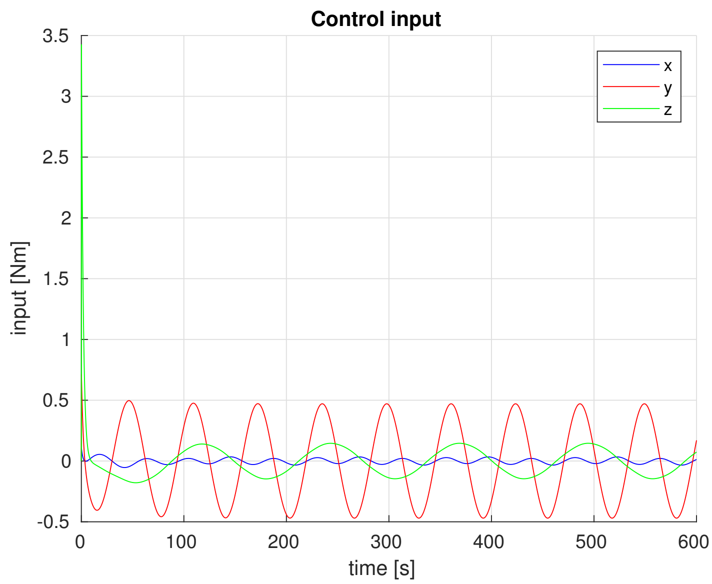

Figure 7: The control inputs of the satellite with a PD controller and Euler angle error.

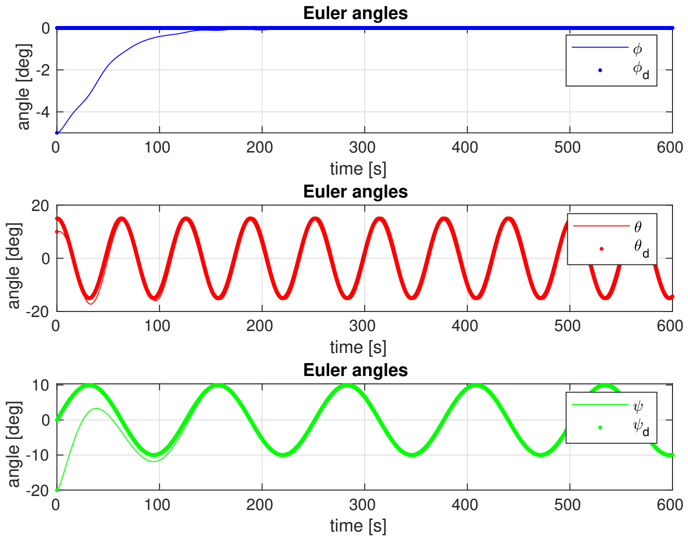

Figure 8: The Euler angles of the satellite with a PD controller, with Euler angle and angle rate error.

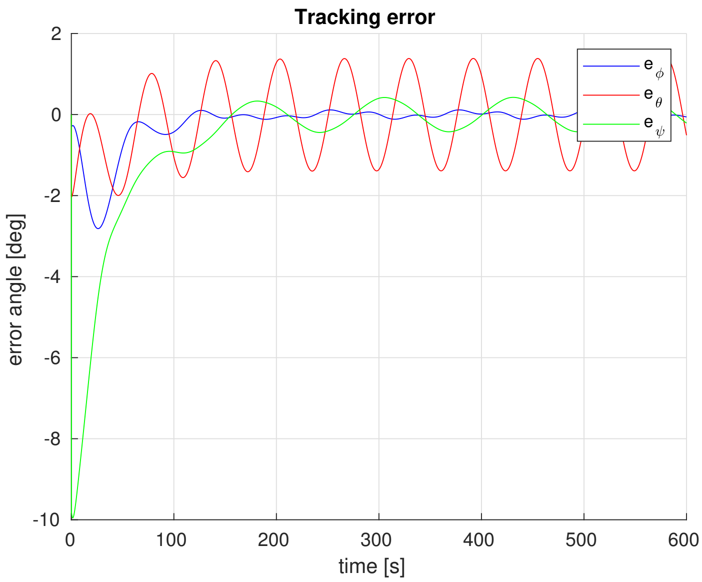

Figure 9: The tracking error of thePD controller, with Euler angle and angle rate error.

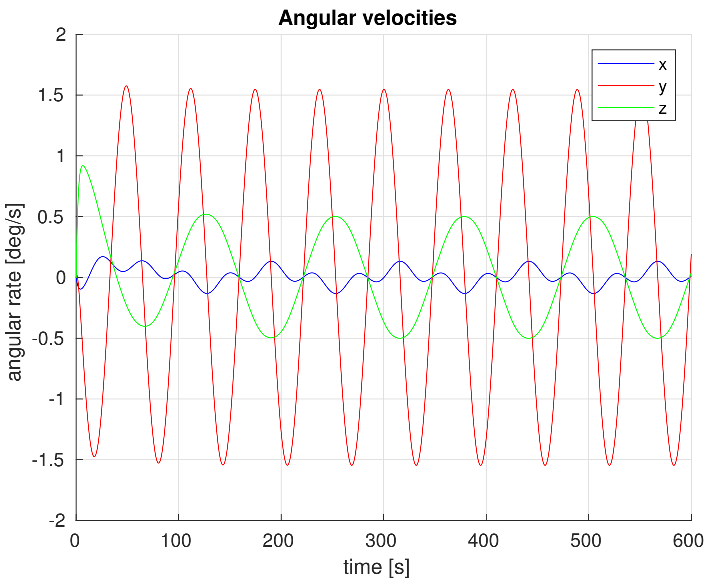

Figure 10: The Euler angles velocities of the satellite with a PD controller, with Euler angle and angle rate error.

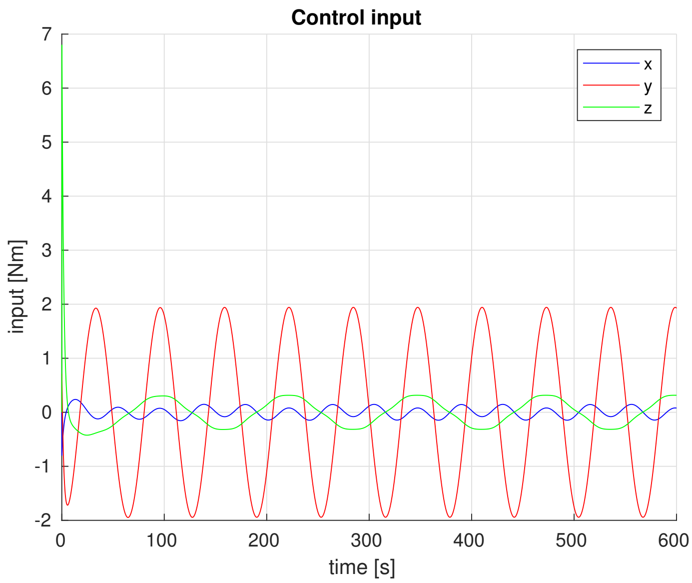

Figure 11: The control inputs of the satellite with a PD controller, with Euler angle and angle rate error.

Problem 2 - Straight-line path following in the horizontal plane

Answer Problem 2 in this file.

Problem 2.1

Answer problem 2.1 here. The Greek letters for sideslip, heading and course are β, ψ and χ, respectively. Equation (10) in the assignment is:

$$\begin{aligned} \dot{x} &= u\cos(\psi) - v\sin(\psi) \\ \dot{y} &= u\sin(\psi) + v\cos(\psi) \end{aligned} \tag{29}$$

You can refer to equations in the report by using the label and the "eqref" command. Example: equation (29) shows the north and east velocities.

Answer:

The body-frame velocities, denoted as u and v for the x- and y-axes respectively, can be transformed into the north and east velocities $\dot{x}$ and $\dot{y}$ through a rotation by the yaw-angle ψ. This is shown in equation (30).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} = \begin{bmatrix} u\cos(\psi) - v\sin(\psi) \\ u\sin(\psi) + v\cos(\psi) \\ r \end{bmatrix} \tag{30}$$

The 3 DOF simplification assumes that the neglected elements of the full 6-DOF (heave z, roll ϕ and pitch θ) have a negligible value during normal operation of the vessel.

The Flow-frame axes are such that the x_f axes is parallel to the freestream flow. Owing to this, the Flow-frame velocities can be described as shown in equation (31).

$$\boldsymbol{v}^{flow} = \begin{bmatrix} U \\ 0 \\ 0 \end{bmatrix}. \tag{31}$$

The transformation from Body- to Flow-frame is normally described by two principle rotations. The first is a rotation about the z-axis by the negative sideslip angle $-\beta$, followed by a rotation about the rotated y axis in the intermediate stability frame by the positive angle of attack α. As only 3 DOF motion is being considered, the angle of attack α is assumed to be close to 0, and the resulting rotation is reduced to identity.

$$\begin{aligned} \boldsymbol{R}_{stab}^{flow}(-\beta) &= \begin{bmatrix} \cos(-\beta) & -\sin(-\beta) & 0 \\ \sin(-\beta) & \cos(-\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 \\ -\sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ \boldsymbol{R}_{b}^{stab}(\alpha) &= \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix}, \\ \alpha \approx 0 \Rightarrow \boldsymbol{R}_{b}^{stab}(\alpha) = \boldsymbol{I} \Rightarrow \boldsymbol{R}_{b}^{flow} &= \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 \\ -\sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \tag{32}$$

The inverse transformation from Flow- to Body-frame are easily found by transposing the rotation matrix found in equation (32).

$$
\boldsymbol{R}^b_{flow} = (\boldsymbol{R}^{flow}_b)^\top = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{33}
$$

The transformation from Flow- to NED-frame can be derived by combining the rotation from equation (30) with the transformation in equation (33). The combined transformation matrix, as well as the relationship between the horizontal-place speed U is shown in equation (34).

$$
\begin{aligned}
\dot{\eta}^n &= \boldsymbol{R}^n_b \boldsymbol{R}^b_{flow} \boldsymbol{v}^{flow} \\
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} &= \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} U\cos(\beta)\cos(\psi) - U\sin(\beta)\sin(\psi) \\ U\cos(\beta)\sin(\psi) + U\sin(\beta)\cos(\psi) \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} U\cos(\beta+\psi) \\ U\sin(\beta+\psi) \\ 0 \end{bmatrix} = \begin{bmatrix} U\cos(\chi) \\ U\sin(\chi) \\ 0 \end{bmatrix}.
\end{aligned} \tag{34}
$$

Problem 2.2

Assuming that the crab angle β is small and negligible for the model yields the simplification shown in equation (35).

$$
\begin{aligned}
\beta \approx 0 \Rightarrow \dot{x} &= U\cos(\beta+\psi) \approx U\cos(\psi), \\
\dot{y} &= U\sin(\beta+\psi) \approx U\sin(\psi).
\end{aligned} \tag{35}
$$

A Taylor series approximation is used for the trigonometric functions in the equation (35). Here the $\mathcal{O}(\psi^2)$ terms are negligible compared to lower order terms for small values of ψ.

$$
\begin{aligned}
\dot{x} &\approx U\cos(\psi) = U(1 - \frac{\psi^2}{2!} + ...) \approx U, \\
\dot{y} &\approx U\sin(\psi) = U(\psi - \frac{\psi^3}{3!} + ...) \approx U\psi.
\end{aligned} \tag{36}
$$

The explicit formula for the cross-trach error $e(t)$, taken from equation (10.59) in [1], is presented in equation (37). Here $x(t), y(t)$ represent the position of the ship in the NED-frame, x_k, y_k represent the position of the closest point on the path being followed, and α_k is the angle between the NED-x-axis and the straight line from the previous to the current waypoint (the path being followed).

$$
e(t) = -[x(t) - x_k]\sin(\alpha_k) + [y(t) - y_k]\cos(\alpha_k). \tag{37}
$$

It is observed that the cross track error becomes equal to the error in y, ie. $e(t) = y(t) - y_k$, if α_k is 0. This corresponds to the path being parallel to the NED-x-axis, ie. that the path is going directly north. As such, using the deviation in the y-coordinate in the NED-frame is only valid when the path is going north.

Problem 2.3

Transfer functions can be written as

$$H(s) = \frac{a_n s^n + ... + a_1 s + a_0}{b_m s^m + ... + b_1 s + b_0} \tag{38}$$

Answer:

The Nomoto model can be written as

$$\begin{aligned} T\dot{r} + r &= K\delta + b \\ \dot{\psi} &= r \end{aligned} \tag{39}$$

Transforming equation (39) using the Laplace-domain yields the expressions in equation (40).

$$\begin{aligned} Tr(s)s + r(s) &= K\delta(s) + b(s) \\ \psi(s)s &= r(s) \end{aligned} \tag{40}$$

The simplified expression for $\dot{y}$ shown in equation (36) is transformed using the Laplace-domain and combined with equation (40) to yield equation (41)

$$\begin{aligned} y(s)s = U\psi(s) \Rightarrow \psi(s) &= \frac{1}{U}y(s)s \Rightarrow r(s) = \frac{1}{U}y(s)s^2, \\ \Rightarrow Tr(s)s + r(s) &= \frac{T}{U}y(s)s^3 + \frac{1}{U}y(s)s^2 \\ &= \frac{1}{U}(Ts+1)s^2 y(s) = K\delta(s) + b(s) \\ \Rightarrow y(s) &= \frac{UK}{(Ts+1)s^2}\delta(s) + \frac{U}{(Ts+1)s^2}b(s). \end{aligned} \tag{41}$$

When selecting a controller for the cross-track error $e(t) = y(t)$, it is natural to select a PID controller. The integral term is necessary due to the bias (constant disturbance), represented by b, which must be removed with integral action. A derivative term is necessary since the three s-terms in the denominator of the process transfer function contribute a negative phase shift of 270 degrees. The derivative term contributes a positive shift of 90 degrees, which can stabilize the system. A more in depth analysis of the controllers contributions to the system stability can be done using Bode-analysis.

Problem 2.4

The PID-controller gains are shown in the table below.

Gain	**Proportional**	**Integral**	**Derivative**
Value	$2.5 \cdot 10^{-3}$	$1.1 \cdot 10^{-6}$	$2.0 \cdot 10^{-1}$

The resulting controller is used to control a vessel which is simulated using the model introduced in equation (39). The simulation results with the PID-controller applied to the rudder to drive the cross-track error $e(t) = y(t)$ to zero are shown in figures 12-16.

To test the effects of the integral-term in the PID-controller, the integral gain is set to 0 and the simulations are run once more. The simulation results are shown in figures 17-17. As shown, there is a steady-state error in the cross-track error, due to the constant bias term.

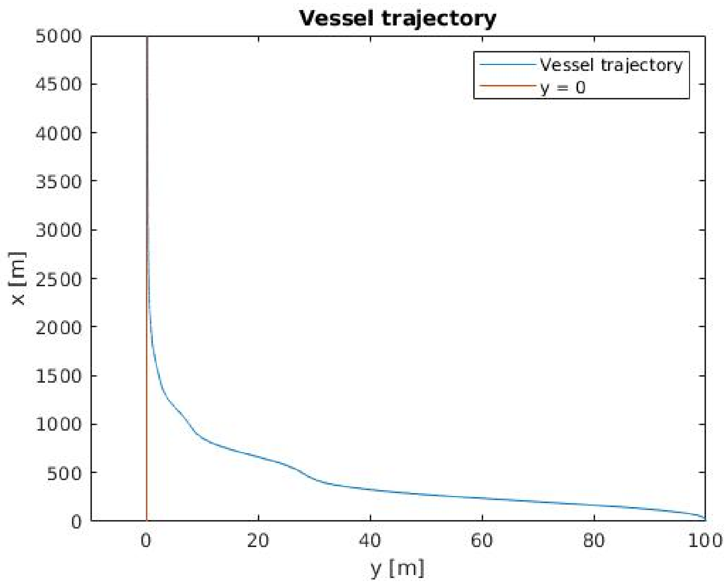

Figure 12: North-east trajectory of vessel.

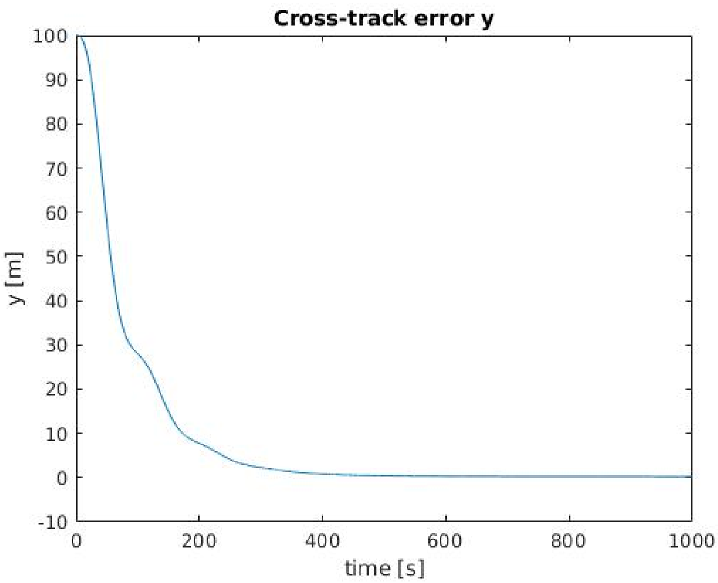

Figure 13: Cross-track error of vessel.

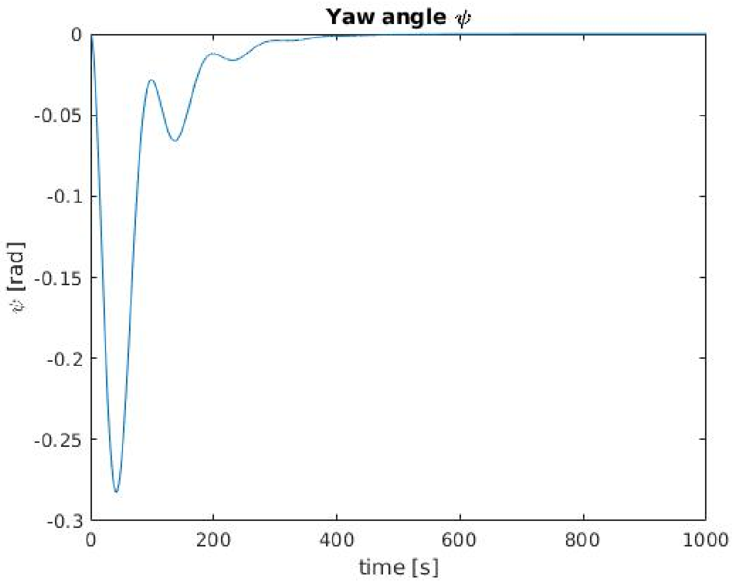

Figure 14: Yaw angle of vessel.

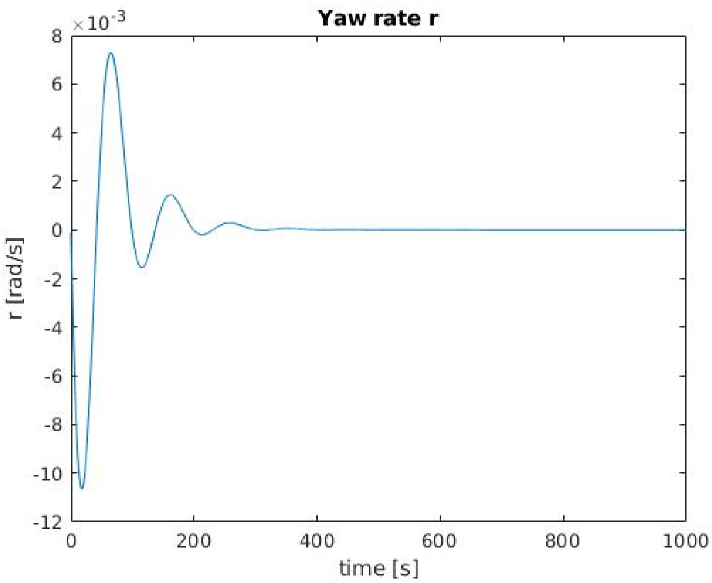

Figure 15: Yaw rate of vessel.

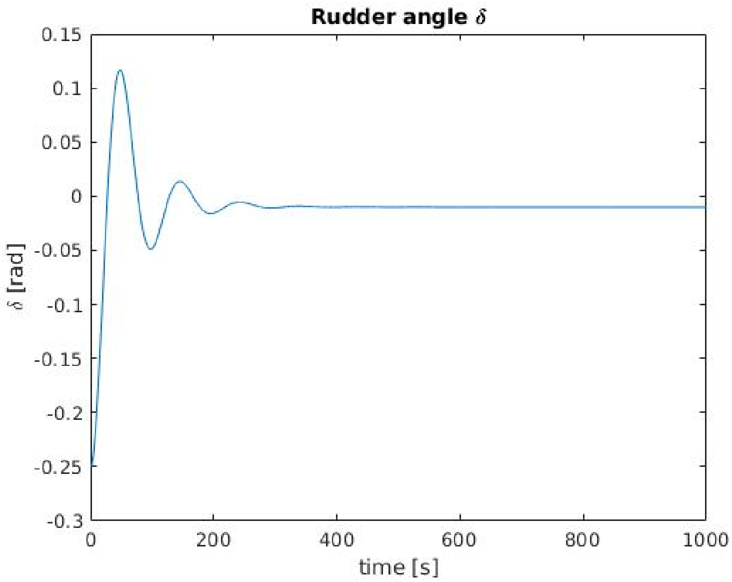

Figure 16: Rudder input of vessel.

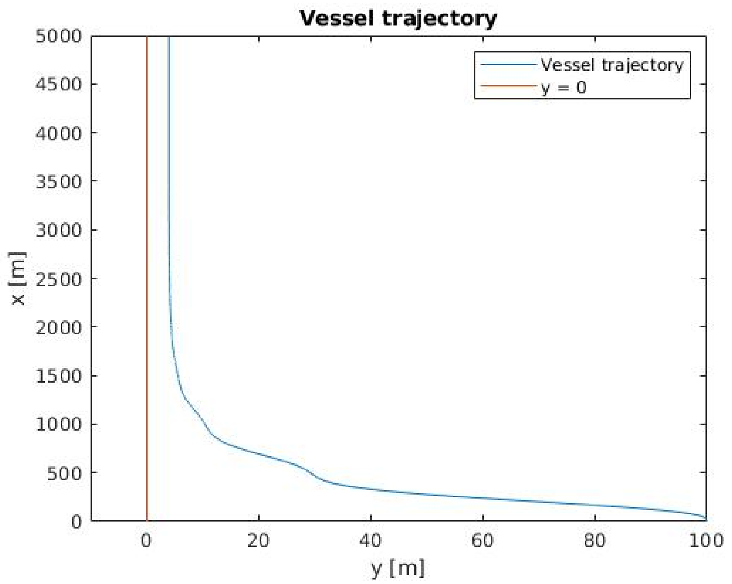

Figure 17: North-east trajectory of vessel, without integral effect.

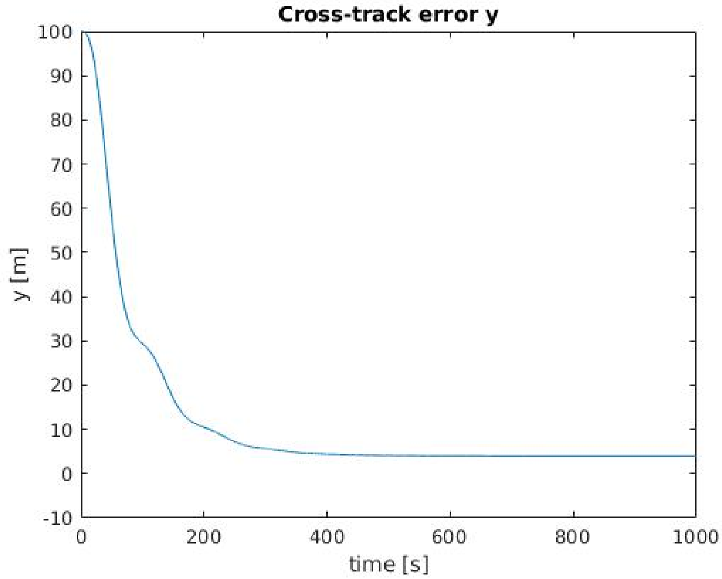

Figure 18: Cross-track error of vessel, without integral effect.

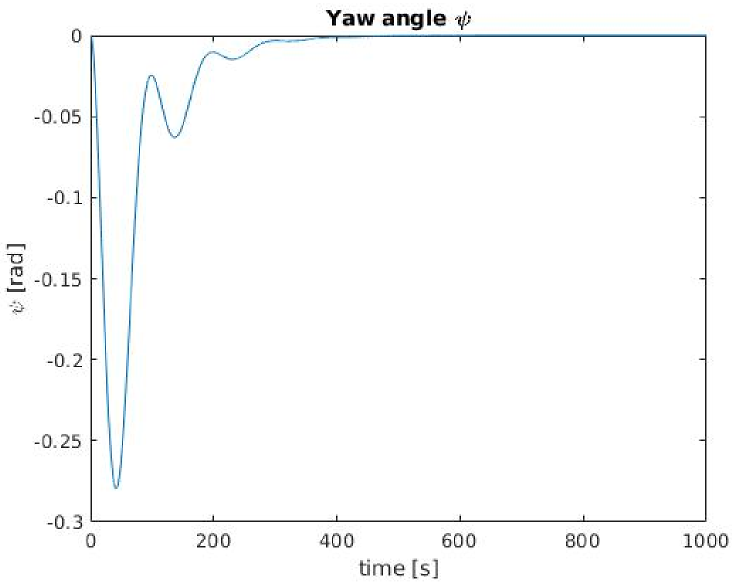

Figure 19: Yaw angle of vessel, without integral effect.

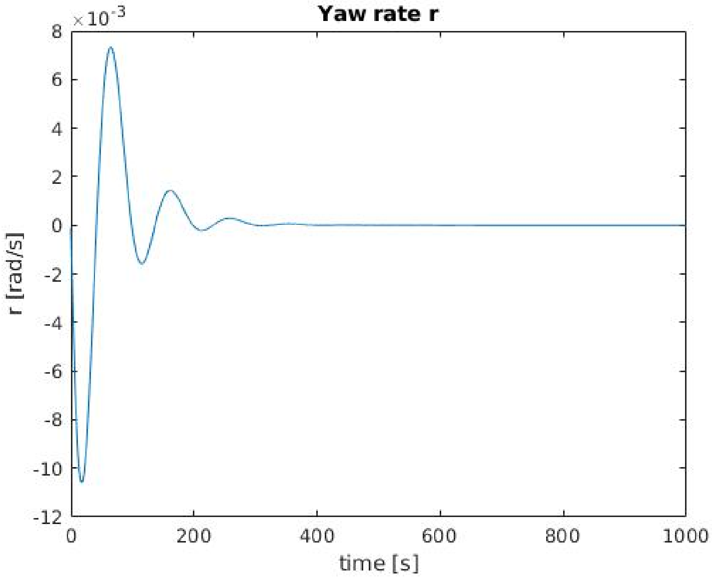

Figure 20: Yaw rate of vessel, without integral effect.

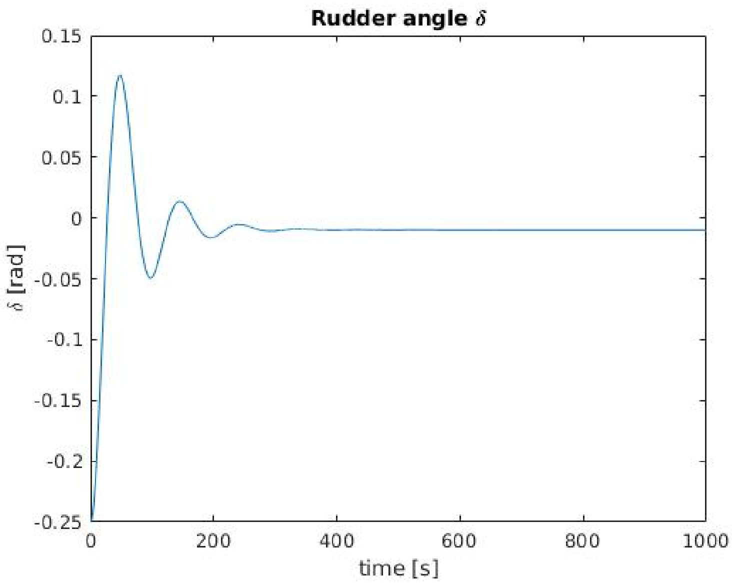

Figure 21: Rudder input of vessel, without integral effect.

References

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