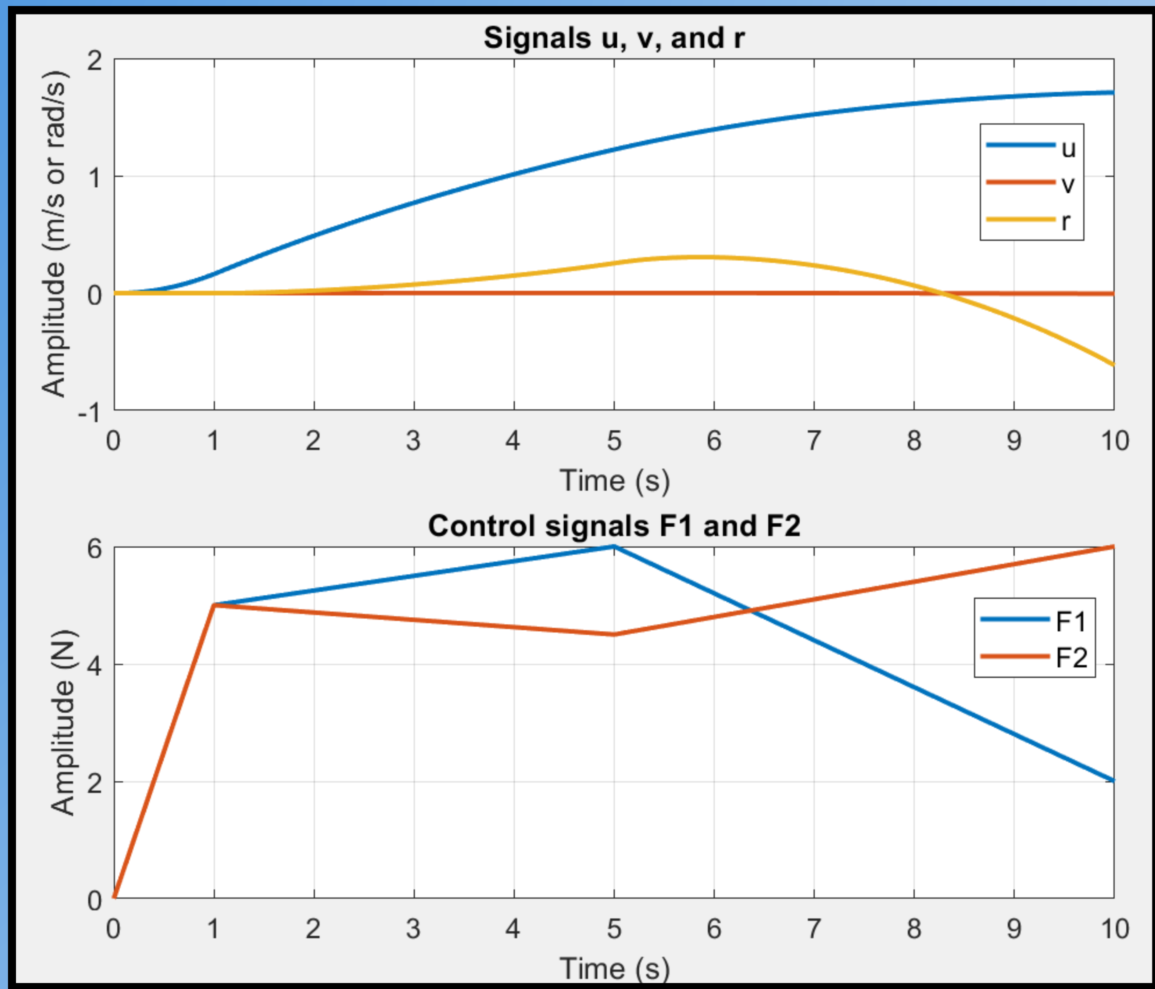
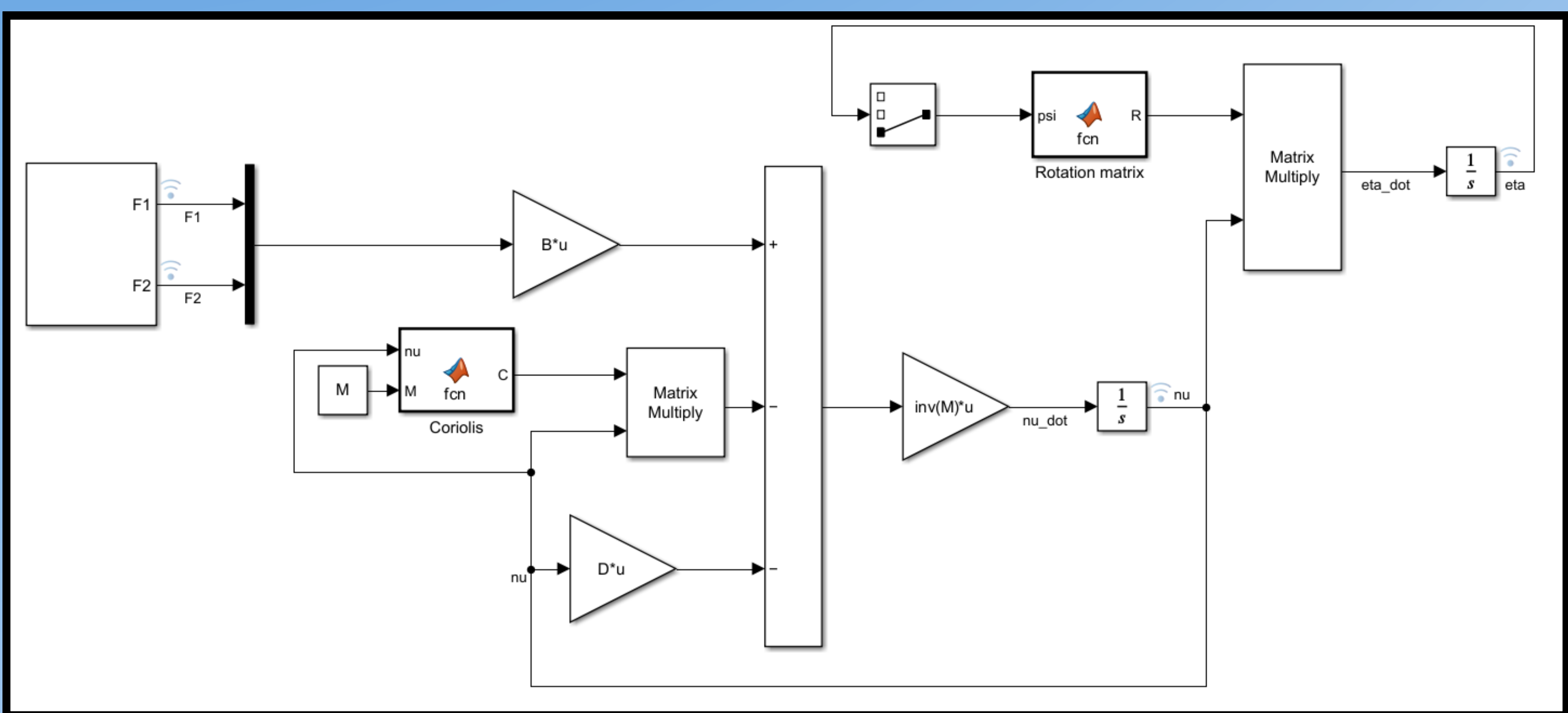
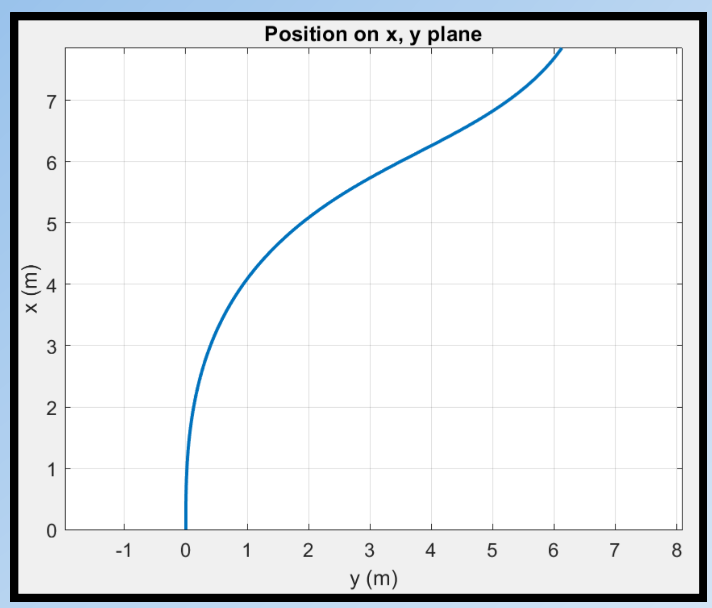
Autonomous Marine Surface Vessel Model





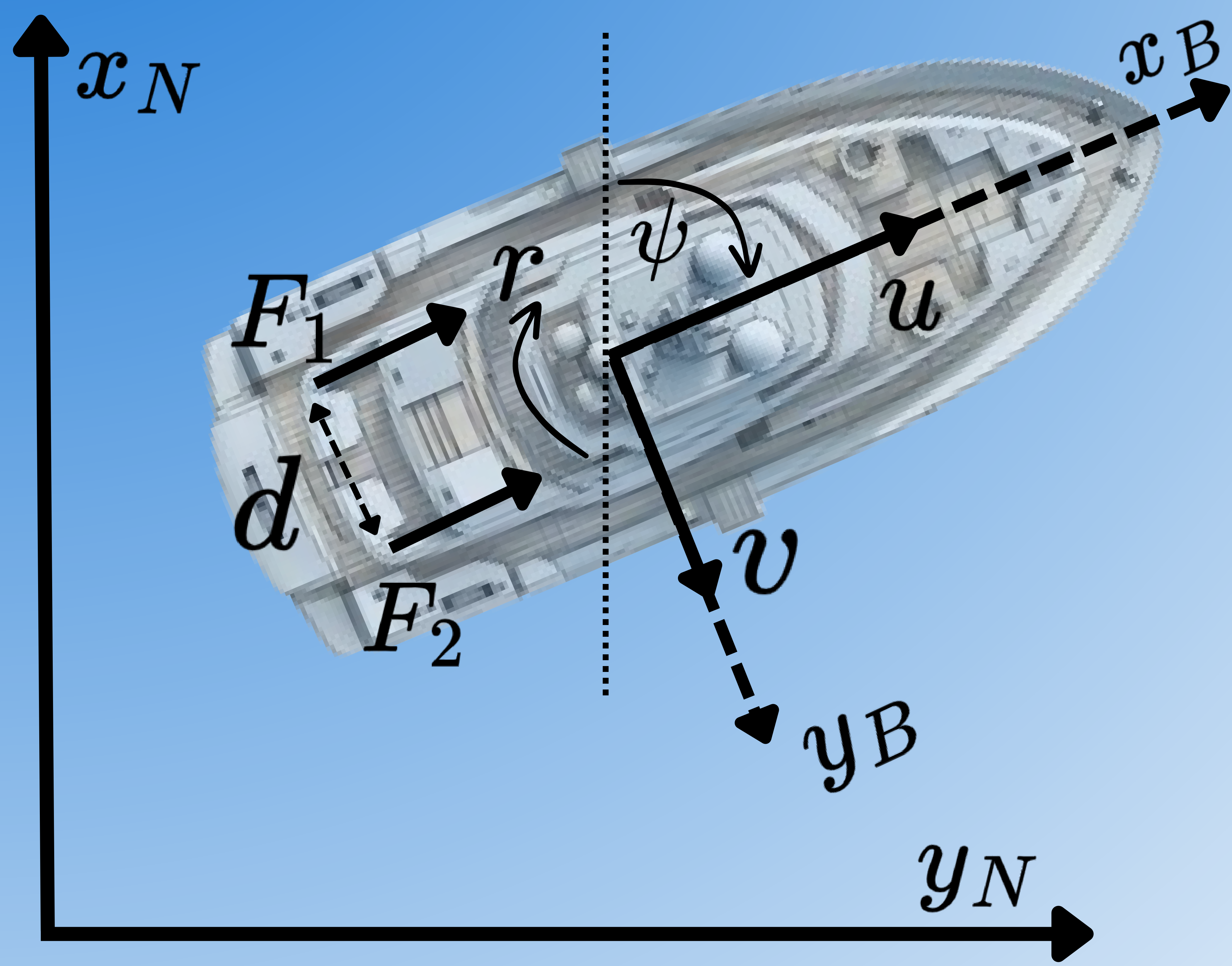


Model

https://github.com/simorxb/autonomous-marine-surface-vessel-model

SIMONE BERTONI
CONTROL LAB

Model – Diagram



Model – Equations – 1

The motion of a ship in the horizontal plane can be described using an inertial frame (x_N, y_N) associated with the sea map and the body-fixed reference frame (x_B, y_B) associated with the ship.

We can use two vectors: $\boldsymbol{\eta} = [x, y, \psi]^T$ and $\boldsymbol{\nu} = [u, v, r]^T$, where (x, y) are the coordinates of the ship's position, ψ is the ship's heading, (u, v) are the linear velocity components x_B and y_B directions, and r is the yaw rate.

The velocity vector determined in the inertial frame (X_N, Y_N) is related to the one in the body-fixed reference frame (X_B, Y_B) by the following kinematic relationship:

$$\dot{\eta} = R(\psi)\boldsymbol{\nu}$$

where $R(\psi)$ is the rotation matrix by angle ψ:

$$R(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and finally the mathematical model of the dynamics:

$$M\dot{\boldsymbol{\nu}} + C(\boldsymbol{\nu})\boldsymbol{\nu} + D(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}$$

where M is the inertia matrix, C is the matrix of Coriolis and centripetal terms, D is the damping matrix, and $\boldsymbol{\tau} = [\tau_x, \tau_y, \tau_n]^T$ is the input control vector.

Model – Equations – 2

The inertia matrix M, which includes the hydrodynamic added inertia, can be written as:

$$M = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & m x_G - Y_{\dot{r}} \\ 0 & m x_G - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix}$$

where m is the vessel mass, I_z is the moment of inertia about the fixed z-axis of the vessel, and $X_{\dot{u}}, Y_{\dot{v}}, Y_{\dot{r}}, N_{\dot{v}}$, and $N_{\dot{r}}$ are hydrodynamic derivatives.

The matrix of Coriolis and centripetal terms has the form:

$$C = \begin{bmatrix} 0 & 0 & -m_{22}v - m_{23}r \\ 0 & m_{11}u & 0 \\ m_{22}v + m_{23}r & -m_{11}u & 0 \end{bmatrix}$$

For a straight-line stable vessel, D is a positive damping matrix due to linear wave drift and laminar skin friction:

$$D = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix}$$

Note that part of this modelling approach has been taken from the article "Control of Dynamic Positioning System with Disturbance
Observer for Autonomous Marine Surface Vessels" by Mirosław Tomera and Kamil Podgórski.

Model – Equations – 3

The parameters for the model are from CyberShip I which is a model of a shipdeveloped in the Department
of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU):

$$M = \begin{bmatrix} 26.4272 & 0 & 0 \\ 0 & 51.3671 & -0.7372 \\ 0 & -0.7372 & 1.2645 \end{bmatrix}$$

$$D = \begin{bmatrix} 4.3411 & 0 & 0 \\ 0 & 6.2983 & 0 \\ 0 & 0 & 1.2577 \end{bmatrix}$$

The model used here has a simplified thruster system respect to CyberShip I, shown in the diagram above. Based on the diagram, we have $\boldsymbol{\tau} = B[F_1, F2]^T$, where:

$$B = \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ \frac{d}{2} & -\frac{d}{2} \end{bmatrix}$$

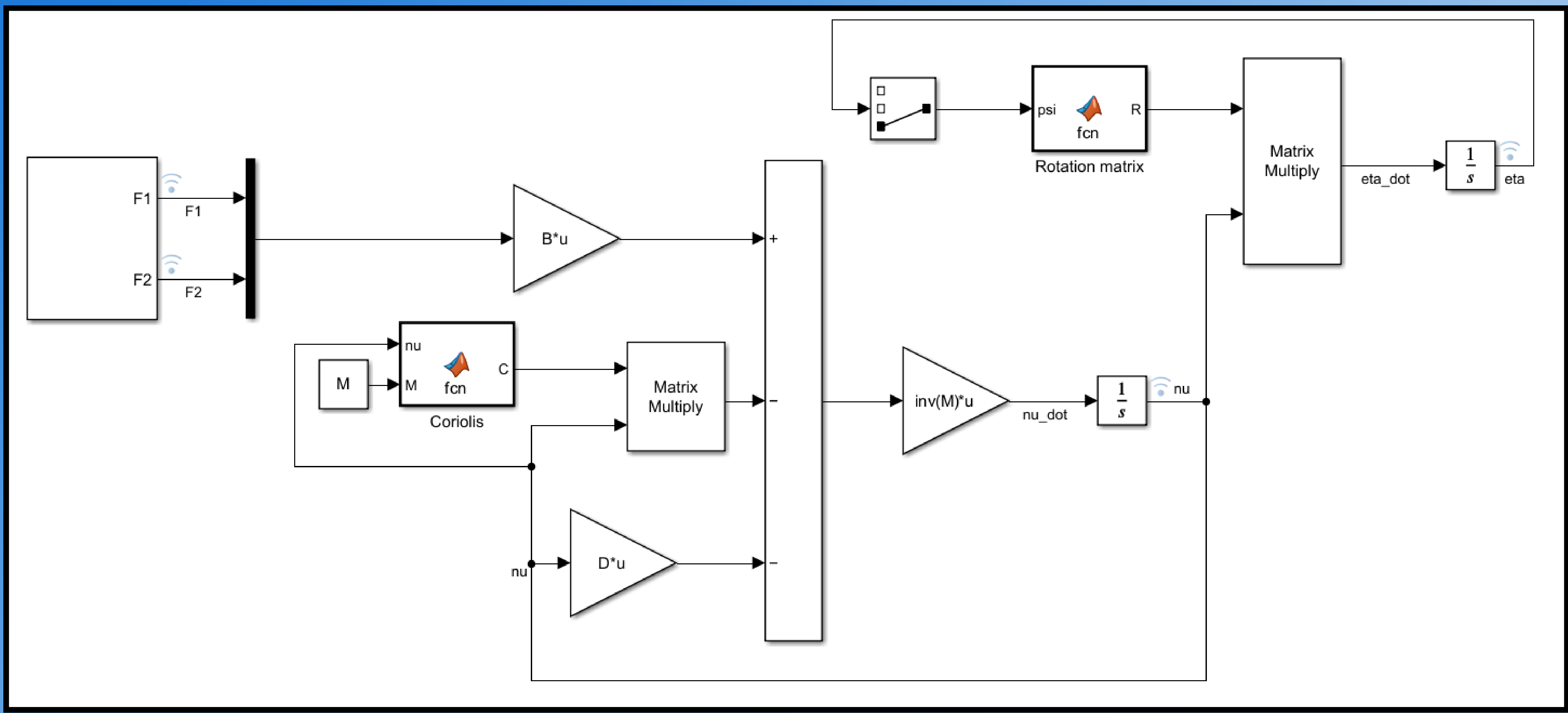
with $d = 0.4$m. To implement the dynamic model in Simulink we need to use:

$$\dot{\boldsymbol{\nu}} = M^{-1}(\boldsymbol{\tau} - C(\boldsymbol{\nu})\boldsymbol{\nu} - D(\boldsymbol{\nu})\boldsymbol{\nu})$$

and

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

Simulink Model



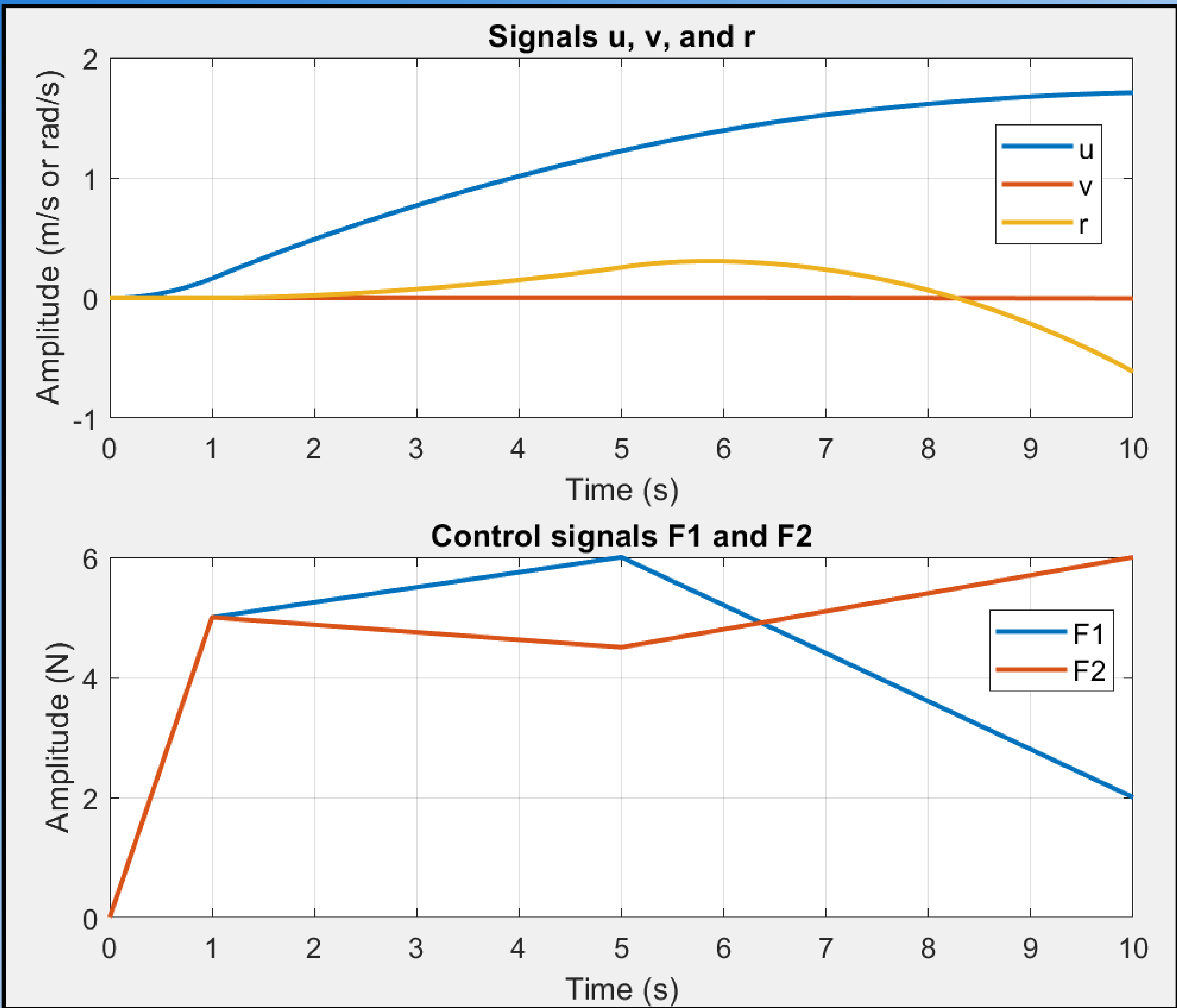
SurfaceVesselModel ▸ Coriolis

```
function C = fcn(nu, M)

% Get u, v, r from nu
u = nu(1);
v = nu(2);
r = nu(3);

% Calculate C
C = [0, 0, -M(2,2)*v - M(2,3)*r;
     0, M(1,1)*u, 0;
     M(2,2)*v + M(2,3)*r, -M(1,1)*u, 0];
```

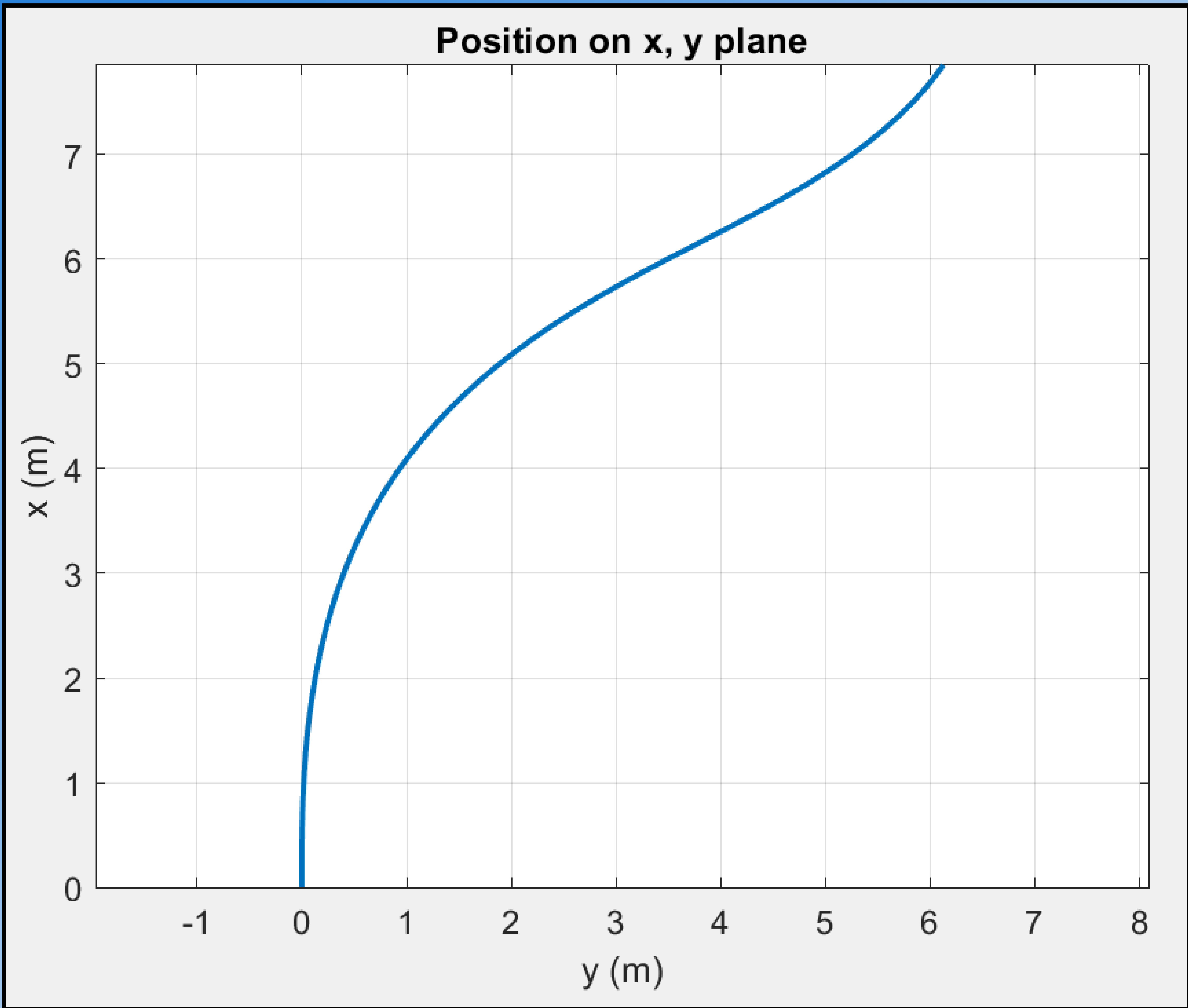
SurfaceVesselModel ▸ Rotation matrix

```
function R = fcn(psi)

% Calculate R
R = [cos(psi), -sin(psi), 0; sin(psi), cos(psi), 0; 0, 0, 1];
```

Result – Plot 1



Result – Plot 2



PID Controller Course

https://simonebertonilab.com

★★★★★

Very helpful and practical

Yoav Golan

I enjoyed this course very much. I learned a lot of practical knowledge in a short time. Simone is very clear and teaches well, thank you! In the future, I would be very interested if Simone added a course with more subjects, such as cascading controllers, rate limiting, and how the controllers look in actual code. Thanks again!

★★★★★

Intuitive and Practical

Ranya Badawi

Simone's explanation of PID control was very intuitive. This is a great starter course to gain a fundamental understanding and some practical knowledge of PID controllers. I highly recommend it. For future topics, I'd be interested in frequency response, transfer functions, Bode plots (including phase/gain margin), Nyquist plots, and stability.

★★★★★

Very good sharing of experience

Romy Domingo Bompart Ballache

I have background in control system for power electronics, I see every lesson very useful.