

Chapter 15 – Motion Control Systems

15.1 Open-Loop Stability and Maneuverability

15.2 Autopilot Design Using Successive Loop Closure

15.3 PID Pole-Placement Algorithms

Chapter 15 covers state-of-the-art successive loop closure and PID control methods for:

- Setpoint regulation
- Trajectory-tracking control
- Path-following control

of marine craft. This includes autopilot design, stationkeeping, position mooring systems, cross-tracking control systems and LOS control systems. In addition to this, control allocation methods are discussed.

Advanced methods such as linear-quadratic optimal control, state feedback linearization, integrator backstepping and sliding mode control are discussed in Chapter 16.



Chapter Goals

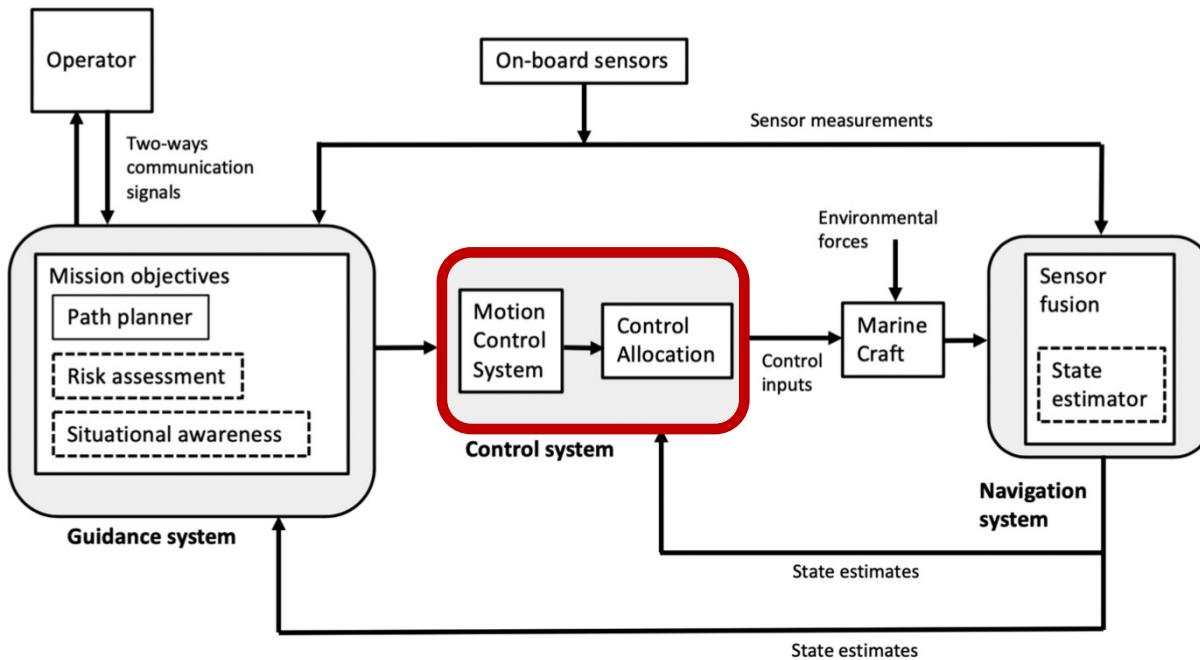
Open-loop stability and maneuverability

- **Stability vs. maneuverability.** Be able to explain the interplay between stability and maneuverability.
- Be able to explain the concepts of **straight-line**, **directional** and **positional motion stability** and how these classical approaches relate to eigenvalues and time constants in a feedback control system.
- Be familiar with **the classical ship maneuvers** and their use, including the:
 - Turning circle
 - Kempf's zigzag maneuver
 - Pull-out maneuver
 - Dieudonné's spiral maneuver
 - Bech's reverse spiral maneuver
 - Stopping trials

State-of-the-art autopilot design

- Be able to design **classical heading** and **course autopilots** using:
 - Successive-loop closure
 - PID control based on pole placement
- Be able to design more advanced control systems such as
 - **Depth and diving control systems**
 - **Path-following control systems**
 - **3-DOF dynamic positioning systems**

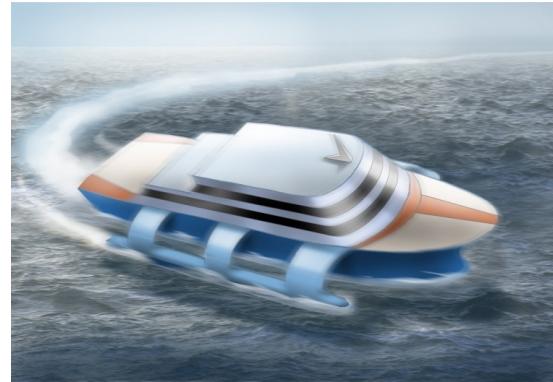
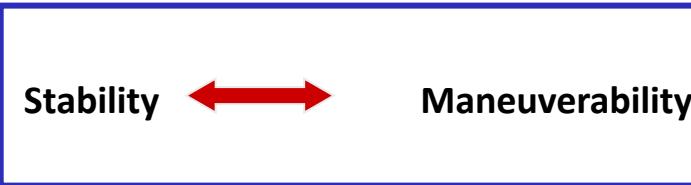
Chapter 15 – Motion Control Systems



Control, or more specifically *motion control* and control allocation, is the action of determining the necessary control forces and moments to be provided by the craft in order to satisfy a certain control objective.

Control allocation: Distribution of generalized control forces to the actuators (see Section 11.2)

15.1 Open-Loop Stability and Maneuverability



Copyright © Bjarne Stenberg

Stability of the uncontrolled ship can be defined as the ability to return to an equilibrium point after a disturbance, without any corrective action of the rudder.

Maneuverability, on the other hand, is defined as the capability of the ship to carry out specific maneuvers.

Excessive stability implies that the control effort will be excessive in a maneuvering situation whereas a marginally stable ship is easy to maneuver. Thus, a **compromise between stability and maneuverability** must be made.

15.1.1 Straight-Line, Directional and Positional Motion Stability

For ships it is common to distinguish between three types of stability:

- Straight-line stability
- Directional stability
- Positional motion stability

Consider the following test system

$$\begin{aligned}\dot{x}^n &= U \cos(\psi) \\ \dot{y}^n &= U \sin(\psi) \\ \dot{\psi} &= r \\ T\dot{r} + r &= K\delta + w\end{aligned}$$

Nomoto model



PD controller

$$\delta = -K_p (\psi - \psi_d) - K_d r$$

Closed-loop test system

$$\underbrace{T\ddot{\psi}}_m + \underbrace{(1 + KK_d)\dot{\psi}}_d + \underbrace{KK_p\psi}_k = \underbrace{KK_p\psi_d + w}_{f(t)}$$

$$m\ddot{\psi} + d\dot{\psi} + k\psi = f(t)$$

15.1.1 Straight-Line, Directional and Positional Motion Stability

Test system

$$m\ddot{\psi} + d\dot{\psi} + k\psi = f(t)$$

$$f(t) = k\psi_d + w$$

Disturbance
impulse at $t = 100$ s

$$\lambda_{1,2} = \frac{-d \pm \sqrt{d^2 - 4km}}{2m}, \quad \omega_n = \sqrt{\frac{k}{m}}, \quad \zeta = \frac{d}{2} \frac{1}{\sqrt{km}}$$

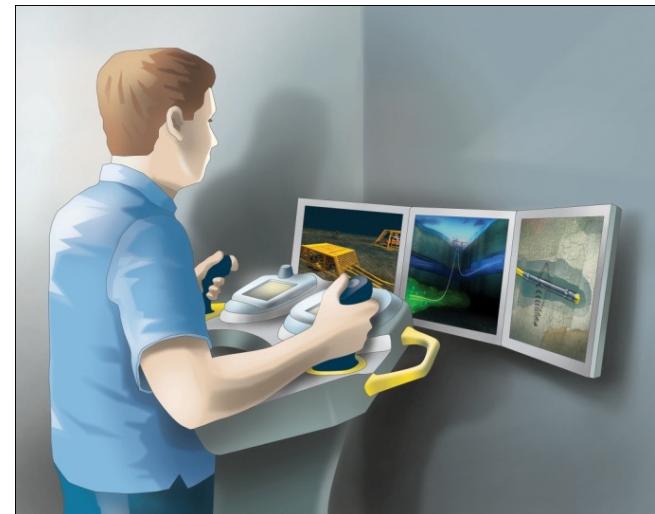
Matlab:

The test system (15.8) is simulated in Matlab for varying model parameters using the MSS toolbox script

StabDemo

The simulation results and the stability analysis are presented on the next pages. This includes the following cases:

- Instability
- Straight-line stability
- Directional stability
- Positional motion stability



Copyright © Bjarne Stenberg

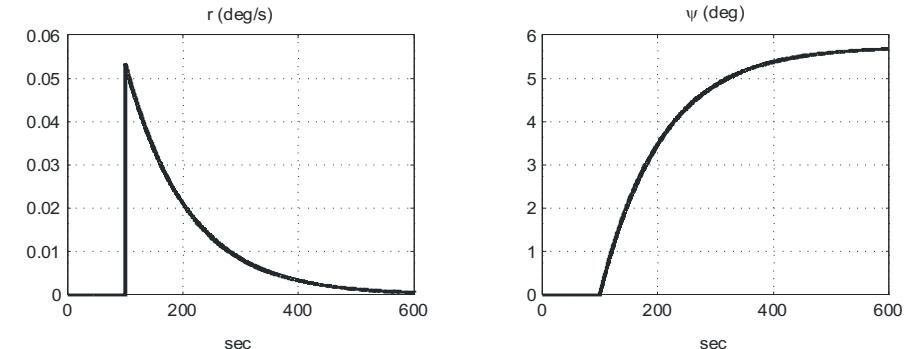
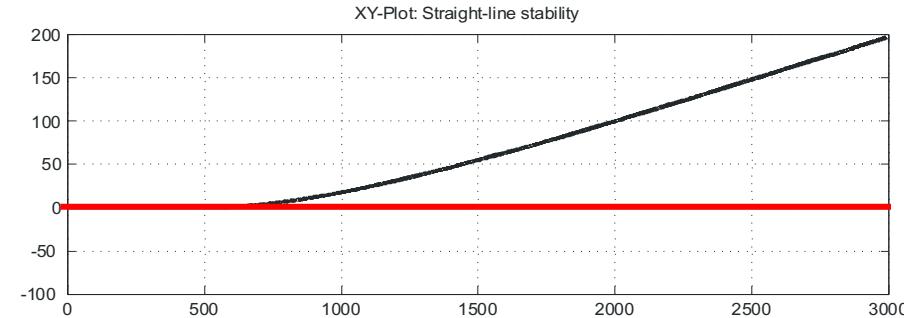
15.1.1 Straight-Line, Directional and Positional Motion Stability

Straight-Line Stability: Consider an uncontrolled ship ($K_p = K_d = 0$) moving on a straight-line path. If the new path is straight after a disturbance w in yaw the ship is said to have straight-line stability.

The direction of the new path will usually differ from the initial path because no restoring forces are present ($k = 0$). This corresponds to:

$$\lambda_1 = -\frac{d}{m} = -\frac{1}{T} < 0, \quad \lambda_2 = 0$$

The requirement $T > 0$ implies straight-line stability for the uncontrolled ship ($\delta = 0$)



15.1.1 Straight-Line, Directional and Positional Motion Stability

Directional Stability (Stability on Course): Directional stability requires the final path to be parallel to the initial path which is obtained for $K_p > 0$. Additional damping is added through $K_d > 0$, that is, PD-control.

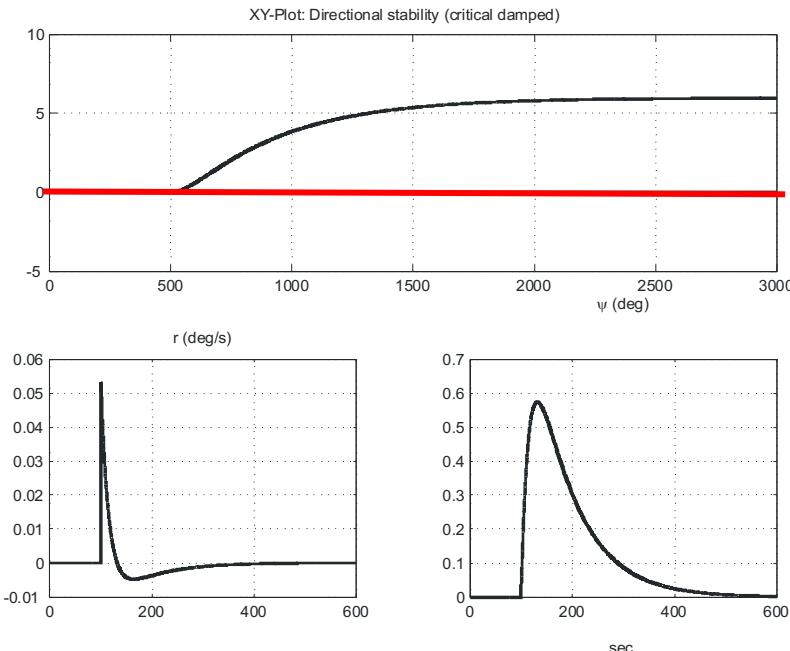
The ship is said to be directionally stable if both eigenvalues have negative real parts, that is

$$\operatorname{Re}\{\lambda_{1,2}\} < 0$$

Two types of directional stability are observed:

1. No Oscillations: ($d^2 - 4km \geq 0$): This implies that both eigenvalues are negative and real-i.e. $\zeta \geq 1$ such that:

$$\lambda_{1,2} = \frac{-d \pm \sqrt{d^2 - 4km}}{2m} = \left(-\zeta \pm \sqrt{\zeta^2 - 1} \right) \omega_n < 0$$



15.1.1 Straight-Line, Directional and Positional Motion Stability

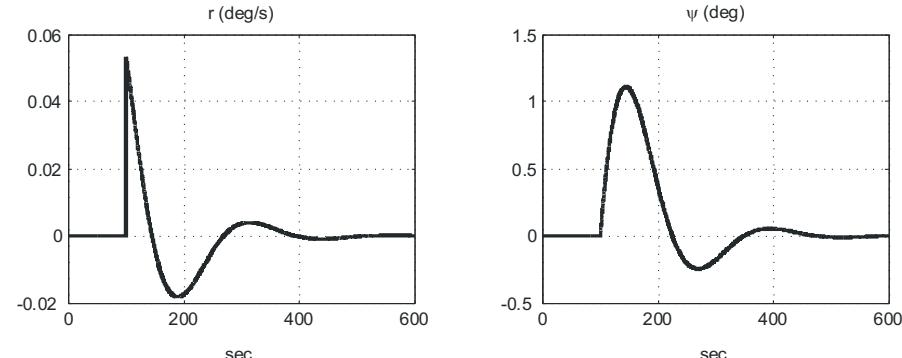
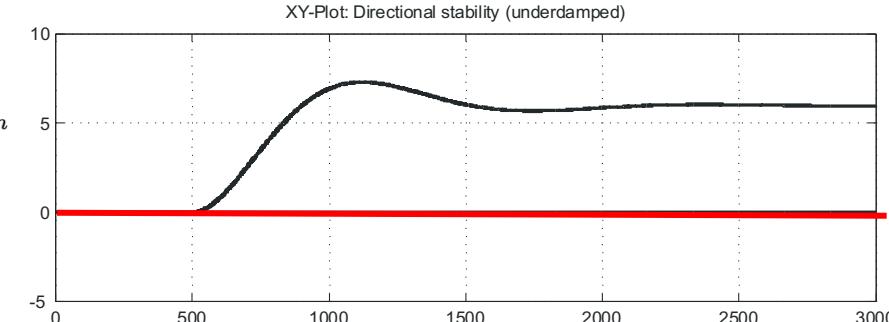
2. Damped Oscillator ($d^2 - 4km < 0$):

This corresponds to two imaginary eigenvalues with negative real parts ($\zeta < 1$), that is:

$$\lambda_{1,2} = \frac{-d \pm j\sqrt{4km - d^2}}{2m} = \left(-\zeta \pm j\sqrt{1 - \zeta^2} \right) \omega_n$$

Note the **oscillations** in both **positions** and **yaw angle**. Directional stability requires feedback control since there are no restoring forces in yaw.

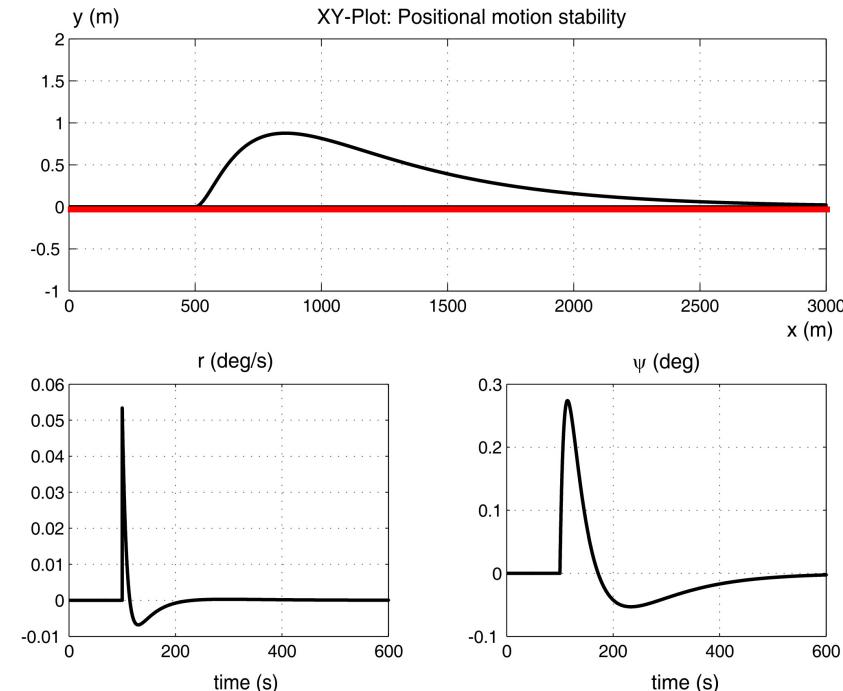
However, in **heave**, **roll** and **pitch** where **metacentric restoring forces** are present ($k > 0$), no feedback is required to damp out the oscillations.



15.1.1 Straight-Line, Directional and Positional Motion Stability

Positional Motion Stability: Positional motion stability implies that the **ship** should return to its original path after a **disturbance**. This can be achieved by including **integral action** in the controller.

A PID controller can be designed to compensate for the unknown disturbance term **w** while a PD controller will generally result in a steady-state offset.



15.1.2 Maneuverability

Ship maneuvers can be used to evaluate the robustness, performance and limitations of a ship.

This is usually done by defining a criterion in terms of a **maneuvering index** or by using a **maneuvering characteristic**.

The Norrbin Measure of Maneuverability

$$P := \frac{\psi'(t' = 1)}{\delta'(t' = 1)}$$

Heading change per unit rudder angle in one ship length traveled at $U=1$ m/s
 $t' = t(U/L)$ is the non-dimensional time

$$T' \ddot{\psi}' + \dot{\psi}' = K' \delta'$$

$$\psi'(t') = K' \left[t' - T' + T' e^{-\frac{t'}{T'}} \right] \delta'(t')$$

$$e^{-\frac{t'}{T'}} = 1 - \frac{t'}{T'} + \frac{(t')^2}{2(T')^2} + O(3)$$

$$\frac{\psi'(t')}{\delta'(t')} \approx K' \left[t' - T' + T' \left(1 - \frac{t'}{T'} + \frac{(t')^2}{2(T')^2} \right) \right] = K \frac{(t')^2}{2T'}$$

$$\frac{\psi'(t' = 1)}{\delta'(t' = 1)} \approx K' \left[\frac{(t')^2}{2T'} \right]_{t'=1} = \frac{K'}{2T'}$$

$$\rightarrow P \approx \frac{1}{2} \frac{K'}{T'}$$

Norrbin concludes that $P > 0.3$ guarantees a reasonable standard of course-change quality for most ships, while $P > 0.2$ seems to be sufficient for large oil-tankers.

15.1.2 Maneuverability

Maneuvering Characteristics

A **maneuvering characteristic** can be obtained by changing or keeping a predefined course and speed of the ship in a systematic manner by means of active controls. The following ship maneuvers have been proposed by ITTC:

- **Turning Circle:** This trial is mainly used to calculate the ship's steady turning radius and to check how well the steering machine performs under course-changing maneuvers.
- **Kempf's Zigzag Maneuver:** The zigzag test is a standard maneuver used to compare the maneuvering properties and control characteristic of a ship with those of other ships. Experimental results of the test can be used to calculate the **K** and **T** values of Nomoto's 1st-order model.
- **Pull-Out Maneuver:** The pull-out maneuver can be used to check whether the ship is straight-line stable or not.
- **Dieudonné's Spiral Maneuver:** The spiral maneuver is used to check straight-line stability. The maneuver gives an indication of the range of validity of the linear theory.
- **Bech's Reverse Spiral Maneuver:** The reverse spiral maneuver can be used for unstable ships to produce a nonlinear maneuvering characteristic. The results from the test indicate which rudder corrections that are required to stabilize an unstable ship.
- **Stopping Trials:** Crash-stops and low-speed stopping trials can be used to determine the ship's head reach and maneuverability during emergency situations.

ISO 13643-1:2017

Ships and marine technology

Manoeuvring of ships — Part 1: General concepts, quantities and test conditions



ITTC – Recommended Procedures

Full Scale Measurements
Manoeuvrability
Full Scale
Manoeuvring Trials Procedure

7.5 - 04

02 - 01

Page 1 of 18

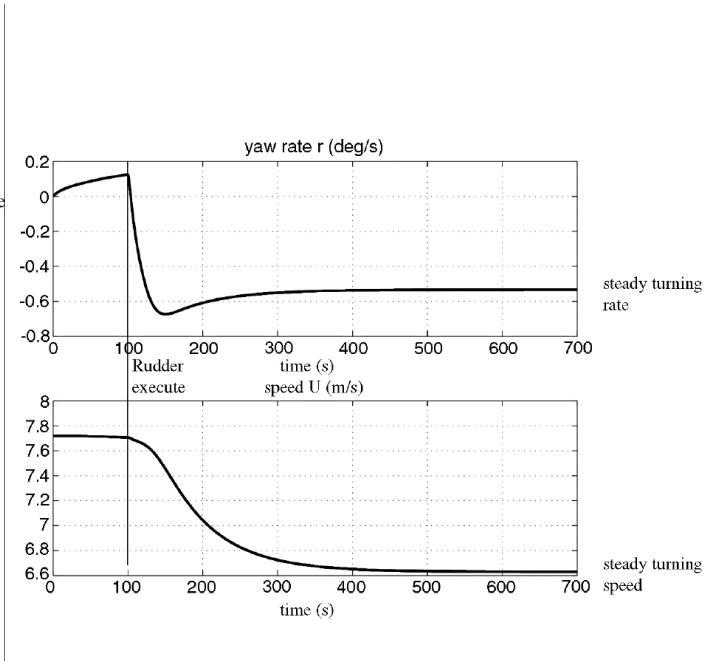
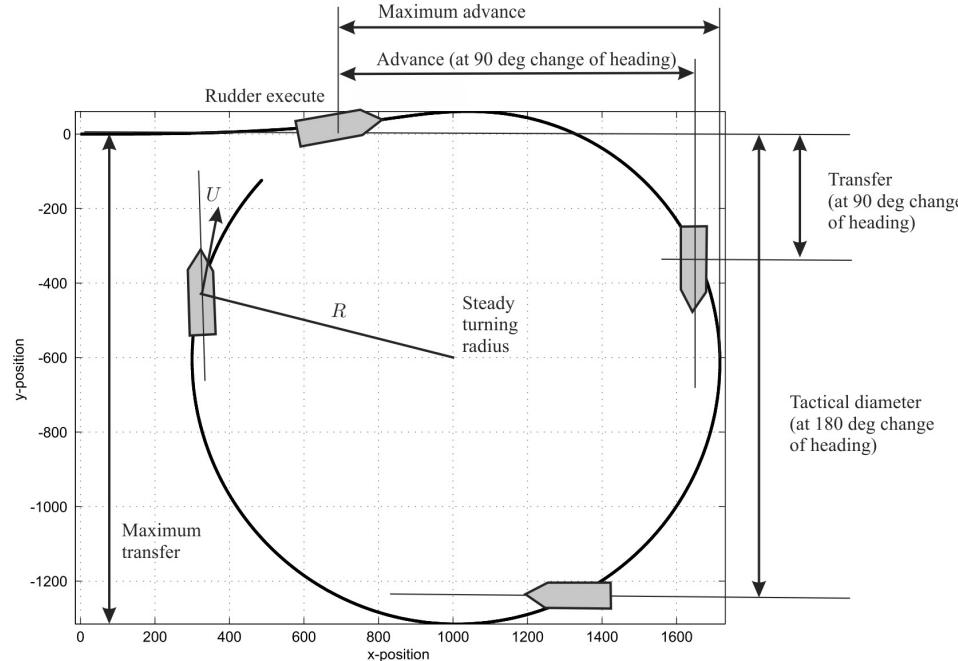
Effective Date
2002

Revision
01

15.1.2 Maneuverability

Turning Circle

The turning circle is probably the oldest maneuvering test. The test can be used as an indication on how well the steering machine and rudder control performs during course-changing maneuvers. It is also used to calculate standard measures of maneuverability such as **tactical diameter**, **advance** and **transfer**.



15.1.2 Maneuverability

Matlab:

The turning circle for the Mariner class vessel is computed using the MSS toolbox script `ExTurnCircle.m`, where

```
t_final = 700;           % final simulation time (s)
t_rudderexecute = 100;   % time rudder is executed (s)
h = 0.1;                 % sampling time (s)

% Mariner class cargo ship, cruise speed U0 = 7.7 m/s
x = zeros(7,1);          % x=[u v r x y psi delta]' (initial values)
ui = -15*pi/180;         % delta_c=-delta_R at time t=t_rudderexecute

[t,u,v,r,x,y,psi,U] = ...
    turncircle('mariner', x, ui, t_final, t_rudderexecute, h)
```

The results are plotted in Figure 15.6. Similar results are obtained by replacing `mariner.m` with the container ship `container.m`; see `ExTurnCircle.m`. The maneuvering characteristics for the Mariner class vessel were computed to be:

Rudder execute (x_b coordinate)	769 m
Steady turning radius	711 m
Maximum transfer	1315 m
Maximum advance	947 m
Transfer at 90 degrees heading	534 m
Advance at 90 degrees heading	943 m
Tactical diameter at 180 degrees heading	1311 m

15.1.2 Maneuverability

The steady turning radius R is perhaps the most interesting quantity obtained from the turning trials. Steady-state analysis gives

$$M\dot{\nu} + N\nu = b\delta$$

$$N\nu = b\delta \implies \nu = N^{-1}b\delta$$

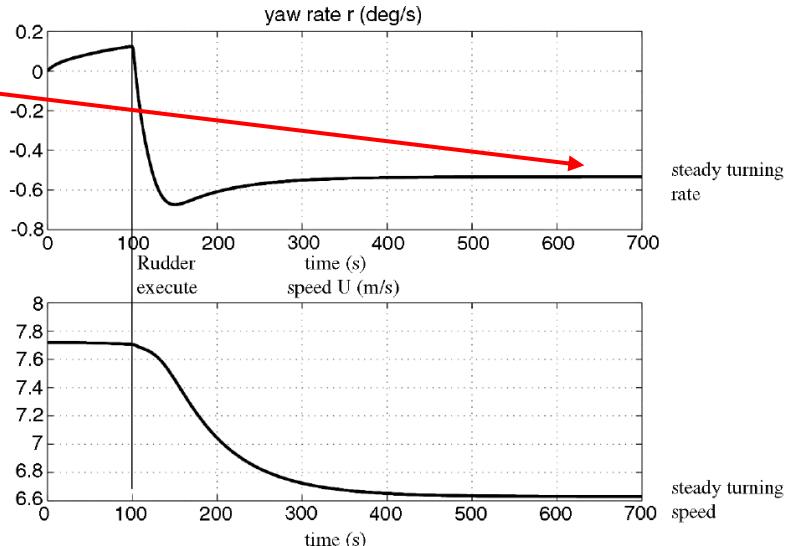
$$r = \frac{(Y_v N_\delta - N_v Y_\delta)}{Y_v(N_r - mx_g U) - N_v(Y_r - mU)} \delta$$

The ship's turning radius R is defined as

$$R := \frac{U}{r} \quad \text{where} \quad U = \sqrt{u^2 + v^2}$$

This gives the non-dimensional formula for the turning radius

$$\left(\frac{R}{L}\right) = \left(\frac{U}{L}\right) \frac{C}{(Y_v N_\delta - N_v Y_\delta)} \frac{1}{\delta}, \quad \delta \neq 0$$



15.1.2 Maneuverability

Example: Determination of the Nomoto Gain and Time Constants

The Nomoto gain and time constants can be computed from a turning test by using nonlinear least-squares curve fitting. For a step input $\delta = \delta_0$, we get

$$T\dot{r} + r = K\delta \quad \longrightarrow \quad r(t) = e^{-\frac{t}{T}} r(0) + (1 - e^{-\frac{t}{T}}) K\delta_0$$

where K and T are unknowns. The Matlab MSS toolbox script [ExKT.m](#) fits this model to a simulated step response of the model [mariner.m](#) which is a nonlinear model of the Mariner class vessel.

The results for a step $\delta_0 = 5 \text{ deg}$ and $U = 7.7 \text{ m/s} = 15 \text{ knots}$, are

$$K = 0.09 \text{ s}^{-1}, \quad T = 22.6 \text{ s}$$

The Norrbin measure of maneuverability becomes

$$P = \frac{1}{2} (K'/T') = 0.87$$

which guarantees good maneuverability since $P > 0.3$.

12.1.2 Maneuverability

Matlab:

```
% ExKT Script for computation of Nomoto gain and time constants
% using nonlinear least squares.
N = 2000;           % number of samples
h = 0.1;            % sample time

xout = zeros(N,2);
x = zeros(7,1);
delta_R = 5 * (pi/180);      % rudder angle step input

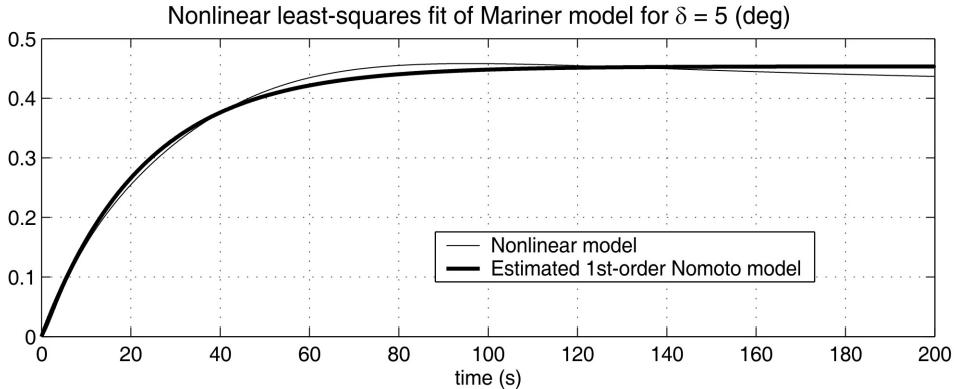
for i=1:N
    xout(i,:) = [(i-1)*h, x(3)];
    xdot = mariner(x,delta_R); % nonlinear Mariner model
    x = euler2(xdot,x,h);    % Euler integration
end

% time-series
tdata = xout(:,1);
rdata = xout(:,2)*180/pi;
```

```
% nonlinear least-squares parametrization: x(1)=1/T and x(2)=K
x0 = [0.01 0.1]';
F = inline('exp(-tdata*x(1))*0 + ...
            x(2)*(1-exp(-tdata*x(1)))*5','x','tdata');
x = lsqcurvefit(F,x0, tdata, rdata);

% plots
plot(tdata,rdata, 'g', tdata, exp(-tdata*x(1))*0 + ...
      x(2)*(1-exp(-tdata*x(1)))*5, 'r');
title('NLS fit of Mariner model for \delta = 5 (deg)')
xlabel('time (s)')
legend('Nonlinear model', 'Estimated 1st-order Nomoto model')
```

$$r(t) = e^{-\frac{t}{T}} r(0) + (1 - e^{-\frac{t}{T}}) K \delta_0$$



15.1.2 Maneuverability

Kempf's Zigzag Maneuver

The **zigzag test** was first proposed by Kempf (1932).

20°-20° Maneuver

(1st angle = rudder command, 2nd angle = change of heading before the rudder is reversed)

The zigzag time response is obtained by moving the rudder to 20° starboard from an initially straight course. The rudder setting is kept constant until the heading is changed 20°, then the rudder is reversed 20° to port. Again, this rudder setting is maintained until the ship's heading has reached 20° in the opposite direction. This process continues until a total of 5 rudder step responses have been completed.

Standardized by the International Towing Tank Conference (ITTC) in 1963

For larger ships, ITTC has recommended the use of a **10°-10°** or a **20°-10° maneuver** to reduce the time and water space required.

Matlab:

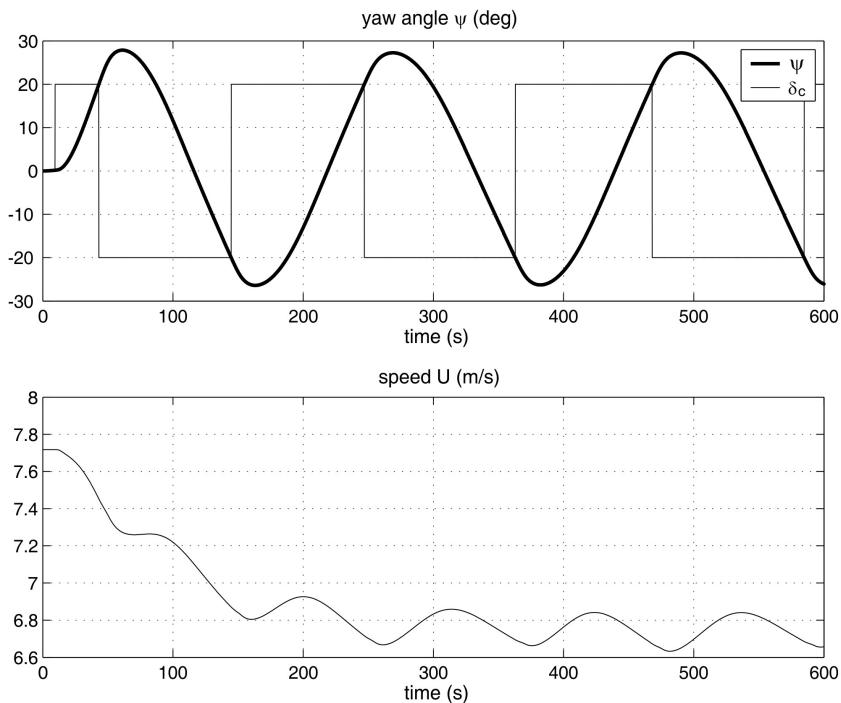
```
t_final = 600; % final simulation time (s)
t_rudderexecute = 10; % time rudder is executed (s)
h = 0.1; % sampling time (s)

% 20-20 zigzag maneuver for the Mariner class cargo ship
% cruise speed U0 = 7.7 m/s (see mariner.m)
x = zeros(7,1); % x = [u v r x y psi delta]' (initial values)
ui = 0; % delta_c = 0 for time t < t_rudderexecute
[t,u,v,r,x,y,psi,U] = ...
    zigzag('mariner',x,ui,t_final,t_rudderexecute,h,[20,20]);

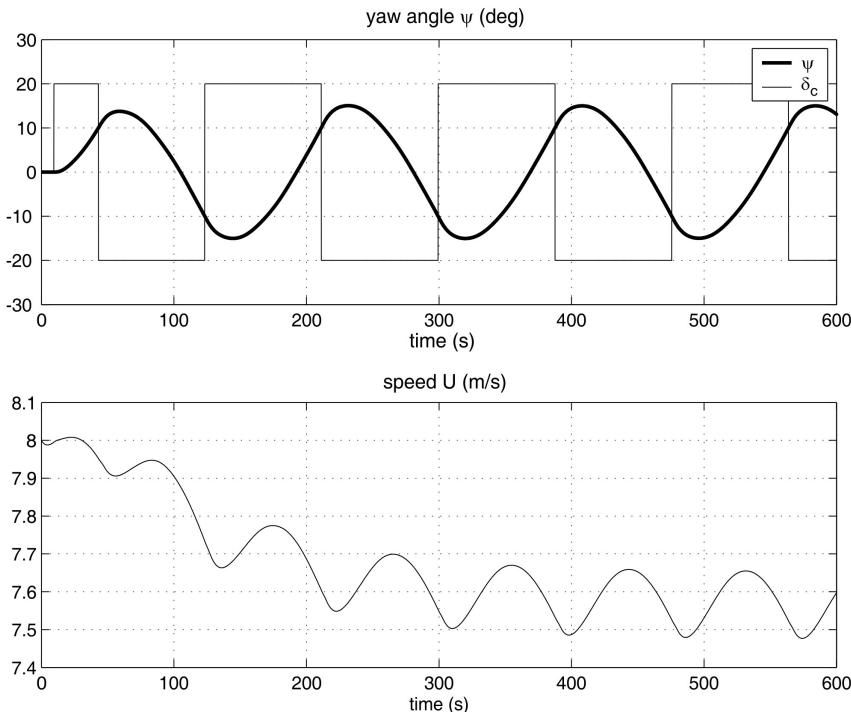
% 20-10 zigzag maneuver for a container ship
% cruise speed 8.0 m/s (see container.m)
x = [8.0 0 0 0 0 0 0 0 0 70]'; % x = [u v r x y psi delta n]'
delta_c = 0; % delta_c = 0 for time t < t_rudderexecute
n_c = 80; % n_c = propeller revolution in rpm
ui = [delta_c, n_c];
[t,u,v,r,x,y,psi,U] = ...
    zigzag('container',x,ui,t_final,t_rudderexecute,h,[20,10]);
```

15.1.2 Maneuverability

20°-20° Maneuver



20°-10° Maneuver



15.1.2 Maneuverability

Pull-Out Maneuver

- In 1969 Roy Burcher proposed a simple test procedure to determine whether a ship is straight line stable or not. This test is referred to as the **pull-out maneuver** (12th ITTC).
- The pull-out maneuver involves a pair of maneuvers in which a **rudder angle of approximately 20°** is applied and returned to zero after steady turning has been attained.
- Both a port and starboard turn must be performed.

Matlab:

```
delta_c = 20*pi/180;      % rudder angle for maneuver (rad)
h = 0.1;                  % sampling time (s)

% Mariner class cargo ship, speed U0 = 7.7 m/s (see mariner.m)
x = zeros(7,1); % x = [ u v r x y psi delta ]' (initial values)
ui = delta_c; % ui = delta_c
[t,r1,r2] = pullout('mariner',x,ui,h);

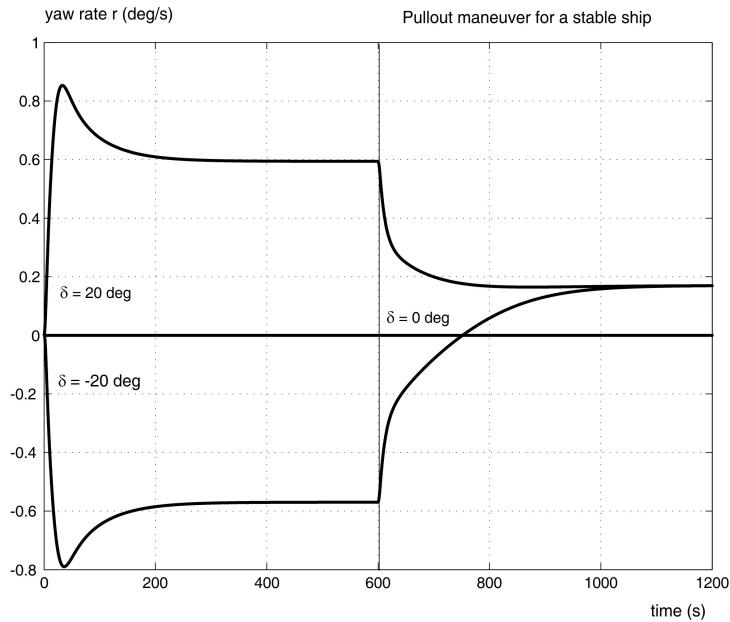
% The Esso Osaka tanker (see tanker.m)
n = 80;
U = 8.23;
x = [ U 0 0 0 0 0 n ]'; % x = [ u v r x y psi delta n ]'
n_c = 80;                % n_c = propeller revolution in rpm
depth = 200;              % water depth
ui = [delta_c, n_c, depth];
[t,r1,r2] = pullout('tanker',x,ui,h);
```

During the test the ship's rate of turn must be measured or at least calculated by numerical derivation of the measured compass heading.

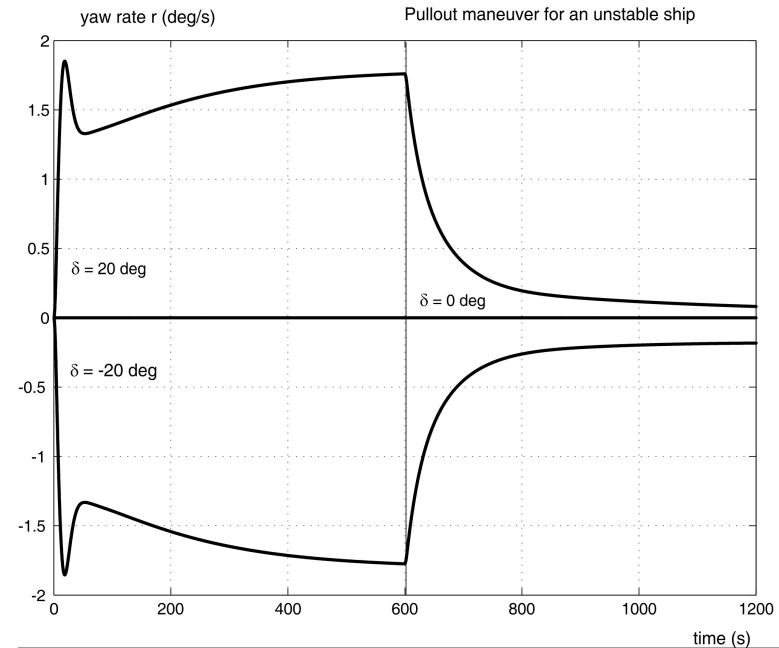
Straight-line stable: the rate of turn will decay to the same value for both the starboard and port turn.

Unstable: the steady rate of turn from the port and starboard turn differ.

15.1.2 Maneuverability



Pullout maneuver for the Mariner class vessels. Note that the **positive and negative curves meet** for the **stable** ship.



Pullout maneuver for a tanker. Note that **the positive and negative curves do not meet** for the **unstable** ship.

15.1.2 Maneuverability

Dieudonné's Spiral Maneuver

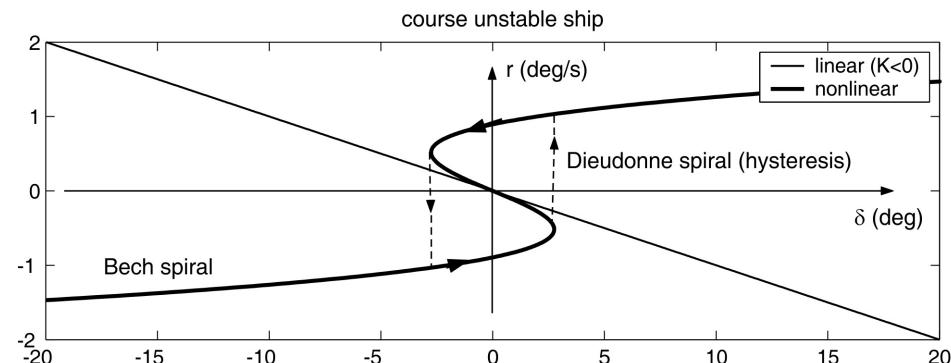
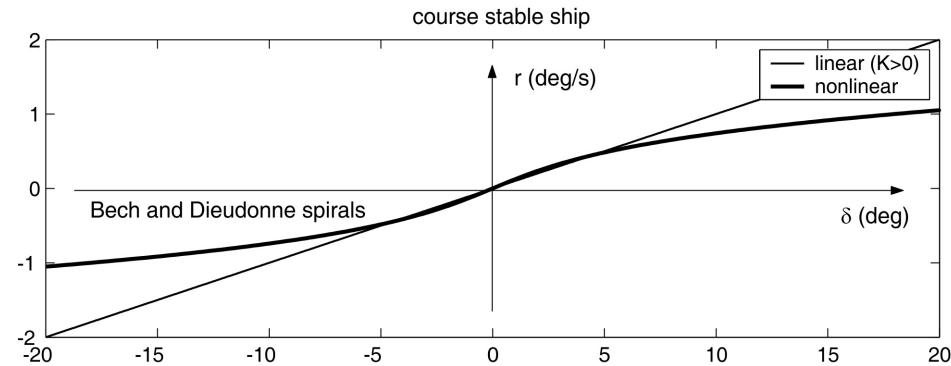
- The direct spiral test was published first in 1949-1950 by the French scientist [Jean Dieudonné](#). An English translation was available in 1953.
- The direct spiral maneuver is used to [check straight-line stability](#). The maneuver also gives an indication of the [degree of stability](#) and [validity of linear theory](#).
- To perform the test the ship should initially be held on a straight course. The rudder angle is then put to 25° starboard and held until steady yawing rate is obtained. After this the rudder angle is decreased in steps of 5° and again held until constant yawing rates are obtained for all the rudder angles. The procedure is performed for all rudder angles between 25° starboard and 25° port. In the range around zero rudder angle the step of 5° rudder should be reduced to obtain more precise values.

The results are plotted in an [r- \$\delta\$ diagram](#).

For [straight-line unstable ships](#) it is recommended to use [Bech's reverse spiral maneuver](#) because of the hysteresis at small turning rates.

15.1.2 Maneuverability

r- δ diagram showing the **Dieudonne** and **Bech spirals** for both a **stable** and **unstable** ship. Note the hysteresis loop in the Dieudonne spiral for the unstable ship



15.1.2 Maneuverability

Bech's Reverse Spiral Maneuver

For [unstable ships](#) within the limits indicated by the pull-out maneuver, [Bech's reverse spiral](#) should be applied. The reverse spiral test was published in 1966.

The ship steering characteristic is nonlinear outside a limited area. The *mean* value of the required rudder deflection δ_{ss} to steer the ship at a constant rate of turn r_{ss} is a nonlinear function

$$\delta_{ss} = H_B(r_{ss})$$

where $H_B(r_{ss})$ is a nonlinear function describing the maneuvering characteristic.

This can be understood by considering the nonlinear model

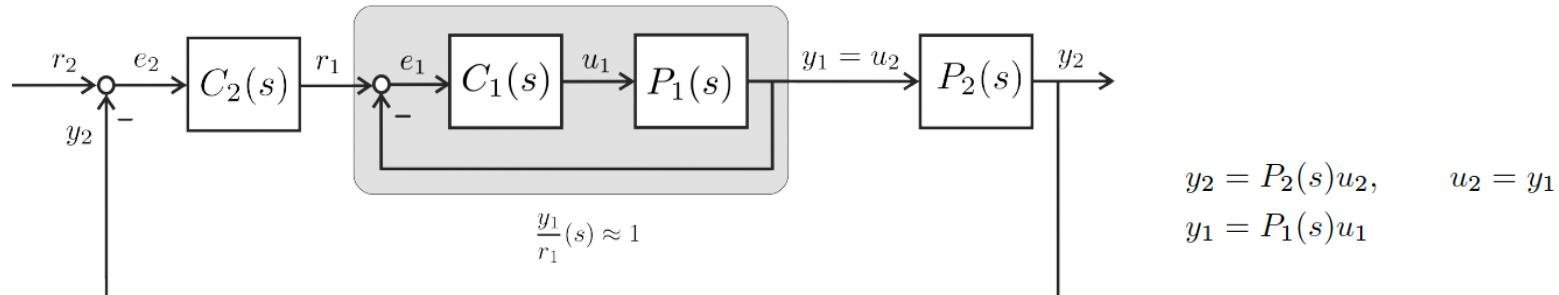
$$T_1 T_2 \ddot{r} + (T_1 + T_2) \dot{r} + K H_B(r) = K(\delta + T_3 \dot{\delta})$$

If $r = r_{ss}$ is constant in steady-state, that is, $\ddot{r} = \dot{r} = \dot{\delta} = 0$ directly gives $\delta_{ss} = H_B(r_{ss})$

The full-scale test is performed by measuring the necessary rudder action required to bring the ship into a desired rate of turn. For an unstable ship this implies that the rudder angle will oscillate about a mean rudder angle. The amplitude of the rudder oscillations should be kept to a minimum. After some time a “[balance condition](#)” is reached and both the mean rudder angle and rate of turn can be calculated.

15.2 Autopilot Design Using Successive Loop Closure

The basic idea behind successive loop closure is to close several simple feedback loops in succession around the open-loop plant dynamics rather than designing a single (presumably more complicated) control system.



Inner loop control system

$$u_1 = C_1(s)e_1, \quad e_1 = r_1 - y_1$$

$$\frac{y_1}{r_1}(s) = \frac{P_1(s)C_1(s)}{1 + P_1(s)C_1(s)} \underset{C_1(s) \gg 1}{\approx} 1$$

Outer loop control system

Bandwidth factor W between 5 and 10

$$\omega_2 = \frac{1}{W} \omega_1$$

$$\frac{y_2}{r_2}(s) \approx \frac{P_2(s)C_2(s)}{1 + P_2(s)C_2(s)}$$

15.2 Autopilot Design Using Successive Loop Closure

Pole-placement algorithm (Beard and McLain 2012)

Consider the second-order open-loop system

$$\frac{y}{u}(s) = \frac{b_0}{s^2 + a_1 s + a_0}$$

$$u = -K_p e_y - K_d \dot{y}, \quad e_y = y - y_d$$

Let the maximum amplitude of the control input be saturated for $\dot{y} = 0$ such that

$$K_p = \frac{u^{\max}}{e_y^{\max}}$$

The maximum control effort u^{\max} and step error e_y^{\max} from a step input of nominal size are user inputs.

The user specify the three parameters: u^{\max} , e_y^{\max} and ζ  K_p and K_d

The desired closed-loop poles are

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$$

and the closed-loop system is

$$\frac{y}{y_d}(s) = \frac{K_p b_0}{s^2 + (a_1 + b_0 K_d)s + (a_0 + b_0 K_p)}$$

Equating the coefficients give

$$\omega_n = \sqrt{a_0 + b_0 \frac{u^{\max}}{e_y^{\max}}}$$

$$K_d = \frac{2\zeta\omega_n - a_1}{b_0}$$

15.2.2 Case Study: Heading Autopilot for Marine Craft

Nomoto model

$$r = \frac{K}{Ts + 1} \delta_R + d_r \quad \psi = \frac{K}{s(Ts + 1)} \delta_R + \frac{1}{s} d_r$$

PD control law

$$\delta_R = -K_{p_\psi} e_\psi - K_{d_\psi} \dot{\psi}, \quad e_\psi = \psi - \psi_d$$

Pole placement algorithm

$$K_{p_\psi} = \frac{\delta_R^{\max}}{e_\psi^{\max}}, \quad K_{d_\psi} = \frac{2\zeta_\psi \omega_\psi T - 1}{K}, \quad \omega_\psi = \sqrt{\frac{K}{T} \frac{\delta_R^{\max}}{e_\psi^{\max}}}$$

PID autopilot with **ssa** modification

$$\delta_R = -K_{p_\psi} \text{ssa}(e_\psi) - K_{d_\psi} \dot{\psi} - K_{i_\psi} \int_0^t \text{ssa}(e_\psi(\tau)) d\tau$$

$$K_{i_\psi} = \frac{\omega_\psi}{10} K_{p_\psi}$$

Integrator gain based on rule of thumb



15.2.3 Case Study: Path-Following Control System for Marine Craft

Cross-track error

$$\dot{y}_e^p = U \sin(\chi_p) \approx U \chi_p \quad \chi_p = \chi - \pi_p$$

Course angle $\chi = \psi + \beta_c$ based on the Nomoto model

$$\chi = \frac{K}{s(Ts + 1)} \delta_R + d_\chi \quad d_\chi = \frac{1}{s} d_r + \beta_c$$

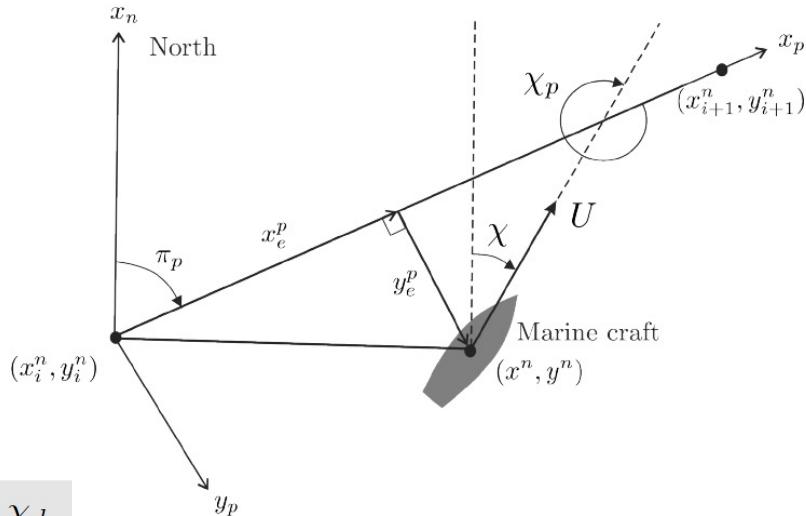
Inner loop PD control law with **ssa** modification

$$\delta_R = -K_{p_\chi} \text{ssa}(e_\chi) - K_{d_\chi} \dot{\chi}, \quad e_\chi = \chi - \chi_d$$

$$K_{p_\chi} = \frac{\delta_R^{\max}}{e_\chi^{\max}}, \quad K_{d_\chi} = \frac{2\zeta_\chi \omega_\chi T - 1}{K}, \quad \omega_\chi = \sqrt{\frac{K}{T} \frac{\delta_R^{\max}}{e_\chi^{\max}}}$$

Closed-loop transfer function:

$$\chi = \underbrace{\frac{\frac{KK_{p_\chi}}{T}}{s^2 + \left(\frac{1}{T} + \frac{KK_{d_\chi}}{T}\right)s + \frac{KK_{p_\chi}}{T}}}_{\approx 1} \chi_d + \underbrace{\frac{1}{s^2 + \left(\frac{1}{T} + \frac{KK_{d_\chi}}{T}\right)s + \frac{KK_{p_\chi}}{T}} d_\chi}_{\approx \frac{T}{KK_{p_\chi}}}$$



15.2.3 Case Study: Path-Following Control System for Marine Craft

Outer loop transfer function

$$y_e^p = \frac{U}{s} \chi_p \approx \frac{U}{s} (\chi_d - \pi_p)$$

Closed-loop loop transfer function

$$\frac{y_e^p}{r_y}(s) = \frac{K_{p_y} U s + K_{i_y} U}{s^2 + K_{p_y} U s + K_{i_y} U}$$

Bandwidth separation

$$\omega_y = \frac{1}{W_y} \omega_\chi$$

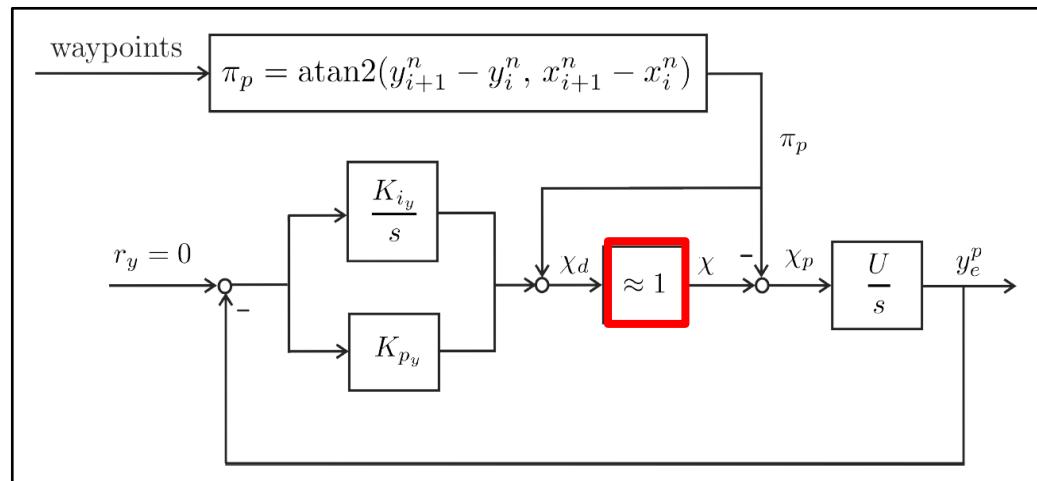
with W_y between 5 and 10

Outer loop PI control law

$$\chi_d = \pi_p - K_{p_y} e_y - K_{i_y} \int_0^t e_y(\tau) d\tau, \quad e_y = y_e^p - 0$$

$$K_{p_y} = \frac{2\zeta_y \omega_y}{U}, \quad K_{i_y} = \frac{\omega_y^2}{U}$$

Depends on speed U



15.2.4 Case Study: Diving Autopilot for Underwater Vehicles

Pitch dynamics controlled by stern dive planes

$$\dot{z}^n = -U\theta + d_z$$

$$(I_y - M_{\dot{q}})\ddot{\theta} - M_q\dot{\theta} + BG_z W\theta = M_{\delta_S}\delta_S$$

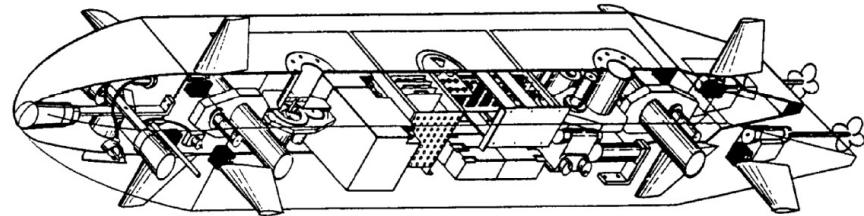
Transfer function

$$\frac{\theta}{\delta_S}(s) = \frac{\frac{M_{\delta_S}}{I_y - M_{\dot{q}}}}{s^2 + \frac{-M_q}{I_y - M_{\dot{q}}}s + \frac{BG_z W}{I_y - M_{\dot{q}}}}$$

Closed-loop transfer function with DC gain

$$\frac{\theta}{\theta_d}(s) = \frac{\frac{M_{\delta_S} K_{p\theta}}{I_y - M_{\dot{q}}}}{s^2 + \left(\frac{-M_q}{I_y - M_{\dot{q}}} + \frac{M_{\delta_S} K_{d\theta}}{I_y - M_{\dot{q}}}\right)s + \left(\frac{BG_z W}{I_y - M_{\dot{q}}} + \frac{M_{\delta_S} K_{p\theta}}{I_y - M_{\dot{q}}}\right)} \stackrel{s=0}{=} \frac{M_{\delta_S} K_{p\theta}}{\underbrace{BG_z W + M_{\delta_S} K_{p\theta}}_{K_{DC}}}$$

where K_{DC} is the nonunity DC gain in pitch.



Inner loop PD control law with **ssa** modification

$$\delta_S = -K_{p\theta} \text{ssa}(e_\theta) - K_{d\theta} \dot{\theta}, \quad e_\theta = \theta - \theta_d$$

$$K_{p\theta} = \frac{\delta_S^{\max}}{e_\theta^{\max}}, \quad K_{d\theta} = \frac{2\zeta_\theta \omega_\theta (I_y - M_{\dot{q}}) + M_{\dot{q}}}{M_{\delta_S}}$$

$$\omega_\theta = \sqrt{\frac{BG_z W}{I_y - M_{\dot{q}}} + \frac{M_{\delta_S}}{I_y - M_{\dot{q}}} \frac{\delta_S^{\max}}{e_\theta^{\max}}}$$

15.2.4 Case Study: Diving Autopilot for Underwater Vehicles

Outer loop transfer function for depth

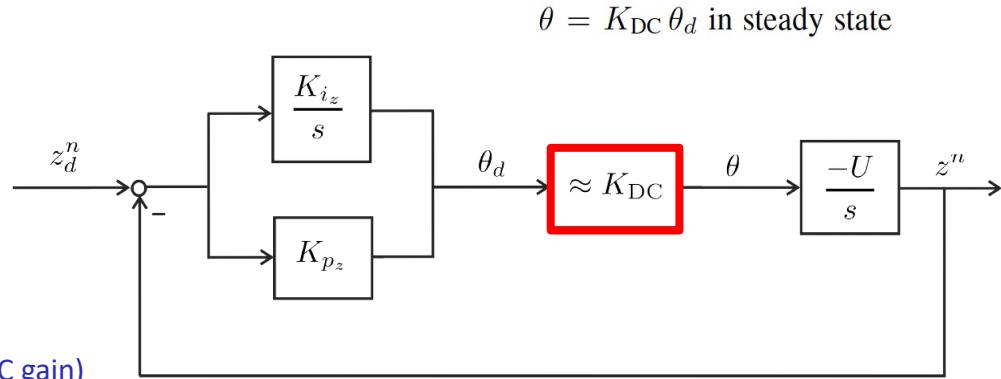
$$z = -\frac{U}{s}\theta \approx -\frac{K_{DC}U}{s}\theta_d$$

Outer loop PI control law (compensates the DC gain)

$$\theta_d = -K_{p_z}e_z - K_{i_z} \int_0^t e_z(\tau) d\tau, \quad e_z = z^n - z_d^n$$

Closed-loop transfer function

$$\frac{z^n}{z_d^n}(s) = \frac{K_{DC}K_{p_z}Us + K_{DC}K_{i_z}U}{s^2 + K_{DC}K_{p_z}Us + K_{DC}K_{i_z}U}$$



$$\theta = K_{DC} \theta_d \text{ in steady state}$$

$$K_{p_z} = \frac{2\zeta_z \omega_z}{K_{DC} U}, \quad K_{i_z} = \frac{\omega_z^2}{K_{DC} U}$$

$$\omega_z = \frac{1}{W_z} \omega_\theta \quad \text{with } W_z \text{ between 5 and 10.}$$

15.3 PID Pole-Placement Algorithms

This section discusses nonlinear PID control for SISO and MIMO systems using pole-placement algorithms. Lyapunov stability analyses for optimal shaping of the kinetic and potential energies of a marine craft are the main tool to derive the pole-placement algorithms. The PID controllers can be applied to a large number of industrial motion control systems including dynamic positioning systems, autopilots for steering and diving as well as path-following control systems.

Mass-damper-spring system

$$m\ddot{x} + d\dot{x} + kx = 0$$

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2x = 0$$

$$2\zeta\omega_n = \frac{d}{m}, \quad \omega_n^2 = \frac{k}{m}$$

$$\lambda_{1,2} = -\underbrace{\zeta\omega_n}_a \pm j\omega$$

$$\omega_n = \sqrt{\frac{k}{m}} \quad \text{natural frequency (undamped oscillator when } d = 0)$$

$$\zeta = \frac{d}{2m\omega_n} \quad \text{relative damping ratio}$$

Damped Oscillator

For a damped system $d > 0$, the frequency of the oscillation will be smaller than the natural frequency of the undamped system

$$a^2 + (r\omega_n)^2 = \omega_n^2$$

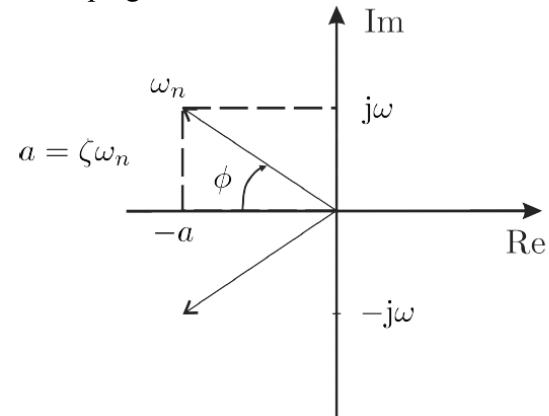
↓

$$a = \underbrace{\sqrt{1 - r^2}}_{\zeta} \omega_n$$

$$\omega = r\omega_n$$

For marine craft a reduction of 0.5% in the natural frequency is common

$$r = 1 - \frac{0.5}{100} = 0.995$$



Undamped oscillator: $a = 0$

$$a^2 + \omega^2 = \omega_n^2, \quad \zeta = \frac{a}{\omega_n} = \cos(\phi)$$

15.3.1 Linear Mass-Damper-Spring Systems

The upper plot shows a mass-damper-spring system for different relative damping ratios.

The lower plot shows the undamped oscillator together with a damped oscillator.

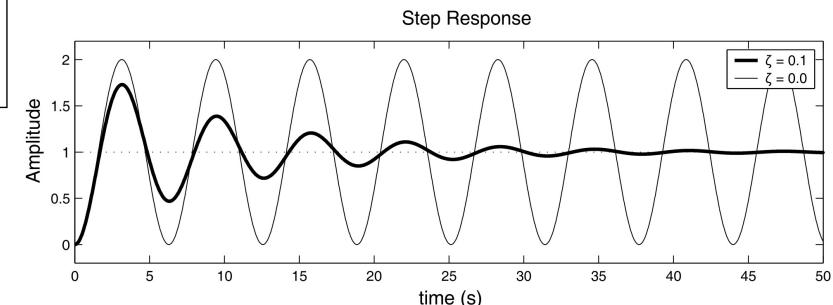
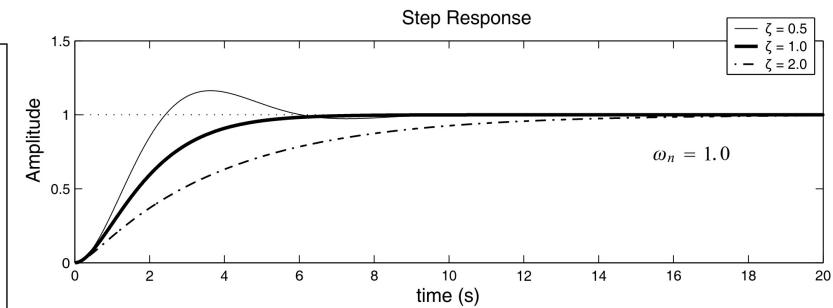
Matlab MSS toolbox: `ExMDS.m`

```

subplot(211)
t = 0:0.01:20;
z = 0.5; sys = tf([wn*wn], [1 2*z*wn wn*wn]); step(sys,t)
hold on
z = 1.0; sys = tf([wn*wn], [1 2*z*wn wn*wn]); step(sys,t)
z = 2.0; sys = tf([wn*wn], [1 2*z*wn wn*wn]); step(sys,t)
hold off

subplot(212)
t = 0:0.01:50;
z = 0.1; sys = tf([wn*wn], [1 2*z*wn wn*wn]); step(sys,t)
hold on
sys = tf([wn*wn], [1 0 wn*wn]); step(sys,t)
hold off

```



15.3.1 Linear Mass-Damper-Spring Systems

Heave, roll and pitch damping: For the *mass-damper-spring* system (15.97) we obtain the following formula for linear damping

$$d = 2\zeta\sqrt{km}, \quad \zeta = \sqrt{1 - r^2}$$



where r is the design parameter. This formula is quite useful to determine the linear damping in *heave, roll* and *pitch*. The mass parameter m and spring coefficient k are easily obtained by other methods; see Chapters 3–5.

For marine craft a reduction of 0.5% in the natural frequency is common (Faltinsen 1990)

$$r = 1 - \frac{0.5}{100} = 0.995$$

Surge, sway and yaw damping: Damping in *surge*, *sway* and *yaw*, however, cannot be determined by formula (15.104) since $k = 0$ in a pure *mass-damper* system. Linear damping for such a system can be found by specifying the time constant $T > 0$ corresponding to

$$m\ddot{x} + d\dot{x} = \tau$$

which for the design parameter $T = m/d$ is equivalent to

$$T\ddot{x} + \dot{x} = \frac{1}{d}\tau$$

This yields the following design formula for linear damping

$$d = \frac{m}{T}$$



For marine craft we can specify the time constant. A step response can be used to estimate the numerical value.

15.3.2 SISO Linear PID Control

PD control law applied to a mass damper spring system

$$\tau = \underbrace{kx_d}_{\substack{\text{reference} \\ \text{feedforward}}} - \underbrace{\left(K_p \tilde{x} + K_d \dot{\tilde{x}} + K_i \int_0^t \tilde{x}(\tau) d\tau \right)}_{\text{PID controller}}$$

Closed-loop system

$$m\ddot{x} + (d + K_d)\dot{x} + (k + K_p)\tilde{x} = 0$$

PID pole-placement formulas

$$K_p = m\omega_n^2 - k$$

$$K_d = 2\zeta\omega_n m - d$$

$$\zeta = \frac{d + K_d}{2m\omega_n}, \quad \omega_n = \sqrt{\frac{k + K_p}{m}}$$

$$K_i = \frac{\omega_n}{10} K_p = \frac{\omega_n}{10} (m\omega_n^2 - k)$$

Rule of thumb for the integral time – 10 times slower than the natural frequency

Pole placement, is a method employed in feedback control system theory to place the closed-loop poles of a plant in pre-determined locations in the s-plane.

15.3.2 SISO Linear PID Control

Relationship between natural frequency and control bandwidth

Definition 15.1 (Control Bandwidth)

The control bandwidth of the system $y = h(s)u$ in Figure 15.18 with negative unity feedback is defined as the frequency ω_b at which the loop transfer function $l(s) = h(s) \cdot 1$ satisfies

$$|l(j\omega)|_{\omega=\omega_b} = \frac{\sqrt{2}}{2}$$

or equivalently

$$20 \log |l(j\omega)|_{\omega=\omega_b} = -3 \text{ dB}$$

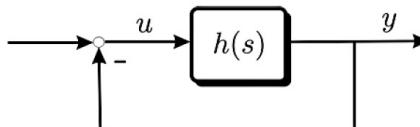
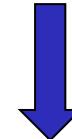


Figure 15.18: Closed-loop feedback system.

$$h(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$



$$\omega_b = \omega_n \sqrt{1 - 2\zeta^2 + \sqrt{4\zeta^4 - 4\zeta^2 + 2}}$$

For a critically damped system, $\zeta = 1.0$, this expression reduces to

$$\omega_b = \omega_n \sqrt{\sqrt{2} - 1} \approx 0.64 \omega_n$$

What does control bandwidth mean? In a linear system, the control law can track a reference signal $y_d = A \cos(\omega t)$ if $\omega < \omega_b$

15.3.2 SISO Linear PID Control

Main Result (PID Controller Tuning Rule)

$$\tau = \underbrace{kx_d}_{\substack{\text{reference} \\ \text{feedforward}}} - \underbrace{\left(K_p \tilde{x} + K_d \dot{\tilde{x}} + K_i \int_0^t \tilde{x}(\tau) d\tau \right)}_{\text{PID controller}}$$

Algorithm 15.1 (SISO PID Pole-Placement Algorithm)

1. Specify the bandwidth: $\omega_b > 0$
2. Specify the relative damping ratio: $\zeta > 0$
3. Compute the natural frequency: $\omega_n = \frac{1}{\sqrt{1-2\zeta^2+\sqrt{4\zeta^4-4\zeta^2+2}}} \omega_b$
4. Compute the P gain: $K_p = m\omega_n^2 - k$
5. Compute the D gain: $K_d = 2\zeta\omega_n m - d$
6. Compute the I gain: $K_i = \frac{\omega_n}{10} K_p$

15.3.2 SISO Linear PID Control

Example 15.7 (Ship Autopilot Design)

Consider the Nomoto model (Nomoto et al. 1957)

$$T\ddot{\psi} + \dot{\psi} = K\delta$$

where ψ is the yaw angle and δ is the rudder angle.

$$m = \frac{T}{K}, \quad d = \frac{1}{K}, \quad k = 0$$

Application of Algorithm 15.1 gives

$$w_n = \frac{1}{\sqrt{1 - 2\zeta^2 + \sqrt{4\zeta^4 - 4\zeta^2 + 2}}} \omega_b$$

and the PID controller gains

$$K_p = \omega_n^2 \frac{T}{K}, \quad K_d = \frac{2\zeta\omega_n T - 1}{K}, \quad K_i = \omega_n^3 \frac{T}{10K}$$

where ω_b and ζ are the design parameters.



15.3.3 MIMO Nonlinear PID Control

Nonlinear equations of motion

$$\begin{aligned}\dot{\boldsymbol{\eta}} &= \mathbf{J}_{\Theta}(\boldsymbol{\eta})\boldsymbol{\nu} \\ \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) &= \boldsymbol{\tau}\end{aligned}$$

PID control law

$$\boldsymbol{\tau} = \mathbf{g}(\boldsymbol{\eta}) + \mathbf{J}_{\Theta}^{\top}(\boldsymbol{\eta})\boldsymbol{\tau}_{\text{PID}}$$

$$\boldsymbol{\tau}_{\text{PID}} = -\mathbf{K}_p\tilde{\boldsymbol{\eta}} - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}} - \mathbf{K}_i \int_0^t \tilde{\boldsymbol{\eta}}(\tau) \mathrm{d}\tau \quad \tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \boldsymbol{\eta}_d$$

Error dynamics for $\mathbf{K}_i = \mathbf{0}$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + [\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})]\boldsymbol{\nu} + \mathbf{J}_{\Theta}^{\top}(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}} = \mathbf{0}$$

$$\mathbf{K}_d^*(\boldsymbol{\eta}) = \mathbf{J}_{\Theta}^{\top}(\boldsymbol{\eta})\mathbf{K}_d\mathbf{J}_{\Theta}(\boldsymbol{\eta})$$



15.3.3 MIMO Nonlinear PID Control

Lyapunov stability analysis

In the Lyapunov stability analysis it is assumed that $\dot{\eta}_d = 0$, that is $\eta_d = \text{constant}$

$$V = \underbrace{\frac{1}{2} \boldsymbol{\nu}^\top \mathbf{M} \boldsymbol{\nu}}_{\text{kinetic energy}} + \underbrace{\frac{1}{2} \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}}}_{\text{potential energy}}$$

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

$$\begin{aligned} \dot{V} &= \boldsymbol{\nu}^\top \mathbf{M} \dot{\boldsymbol{\nu}} + \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}} \\ &= \boldsymbol{\nu}^\top \left[\mathbf{M} \dot{\boldsymbol{\nu}} + \mathbf{J}_\Theta^\top(\boldsymbol{\eta}) \mathbf{K}_p \tilde{\boldsymbol{\eta}} \right] \end{aligned}$$

where $\dot{\tilde{\boldsymbol{\eta}}} = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d = \dot{\boldsymbol{\eta}}$ and $\dot{\boldsymbol{\eta}}^\top = \boldsymbol{\nu}^\top \mathbf{J}_\Theta^\top(\boldsymbol{\eta})$.

$$\begin{aligned} \dot{V} &= -\boldsymbol{\nu}^\top [\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})] \boldsymbol{\nu} \\ &= -\boldsymbol{\nu}^\top [\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})] \boldsymbol{\nu} \end{aligned}$$

Krasovskii–LaSalle’s Theorem (Appendix A.1) can be used to prove that the nonlinear system with nonlinear PD control ($\mathbf{K}_i = \mathbf{0}$) is globally asymptotically stable (GAS) if $\mathbf{J}_\Theta(\boldsymbol{\eta})$ is defined for all $\boldsymbol{\eta}$ (no representation singularity)

15.3.3 MIMO Nonlinear PID Control

MIMO pole-placement algorithm

The MIMO pole-placement algorithm is based on the closed-loop system

$$M\dot{\nu} + [C(\nu) + D(\nu) + K_d^*(\eta)]\nu + J_\Theta^\top(\eta)K_p\tilde{\eta} = 0$$

Assume that $\dot{J}_\Theta(\eta) \approx \mathbf{0}$ and premultiply (15.137) with $J_\Theta^{-\top}$ such that

$$J_\Theta^{-\top}(\eta)M J_\Theta^{-1}(\eta)\ddot{\eta} + J_\Theta^{-\top}(\eta)[C(\nu) + D(\nu)]J_\Theta^{-1}(\eta)\dot{\eta} + K_d\dot{\eta} + K_p\tilde{\eta} = 0$$

The next step is to compare this expression to

$$M^*(\eta)[\ddot{\eta} + 2Z\Omega_n\dot{\eta} + \Omega_n^2\tilde{\eta}] = 0$$

where Z and Ω_n are two design matrices to be specified later, and

$$M^*(\eta) := J_\Theta^{-\top}(\eta)M J_\Theta^{-1}(\eta)$$

Equating the above equations gives the gain requirements

$$\begin{aligned} 2M^*(\eta)Z\Omega_n &:= J_\Theta^{-\top}(\eta)[C(\nu) + D(\nu)]J_\Theta^{-1}(\eta) + K_d \\ M^*(\eta)\Omega_n^2 &:= K_p \end{aligned}$$

which can be solved for K_p and K_d .



15.3.3 MIMO Nonlinear PID Control

Main Result (PID Controller Tuning Rule)

$$\tau = g(\eta) + J_{\Theta}^{\top}(\eta) \tau_{\text{PID}}$$

$$\tilde{\eta} = \eta - \eta_d$$

$$\tau_{\text{PID}} = -K_p \tilde{\eta} - K_d \dot{\eta} - K_i \int_0^t \tilde{\eta}(\tau) d\tau$$

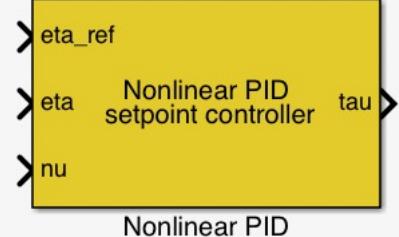
MSS m-file (3 DOF DP control law)

```
function tau = PIDnonlinearMIMO(eta,nu,eta_ref,M,wn,zeta,T_f,h)
% tau = PIDnonlinearMIMO(eta,nu,eta_ref,M,wn,zeta,T_f,h)
% Nonlinear MIMO PID regulator for dynamic positioning (DP). For 3-DOF
% models (surge, sway, and yaw) the output is: tau = [tau1, tau2, tau6]'.
% In the general case, tau = [tau1, tau2, 0, 0, 0, tau6]', where
% .
% z_int = eta - eta_d,      where eta_d = 1 / (T_f * s + 1) * eta_ref
% .
% tau = -R(psi)' * ( Kp * (eta - eta_d) + Kd * nu + Ki * z_int )
% .
% Kp = M_diag * wn * wn,          M_diag = diag(diag(M))
% Kd = M_diag * 2 * zeta * wn
% Ki = 1/10 * Kp * wn
```

Algorithm 15.2 (MIMO nonlinear PID Pole-Placement Algorithm)

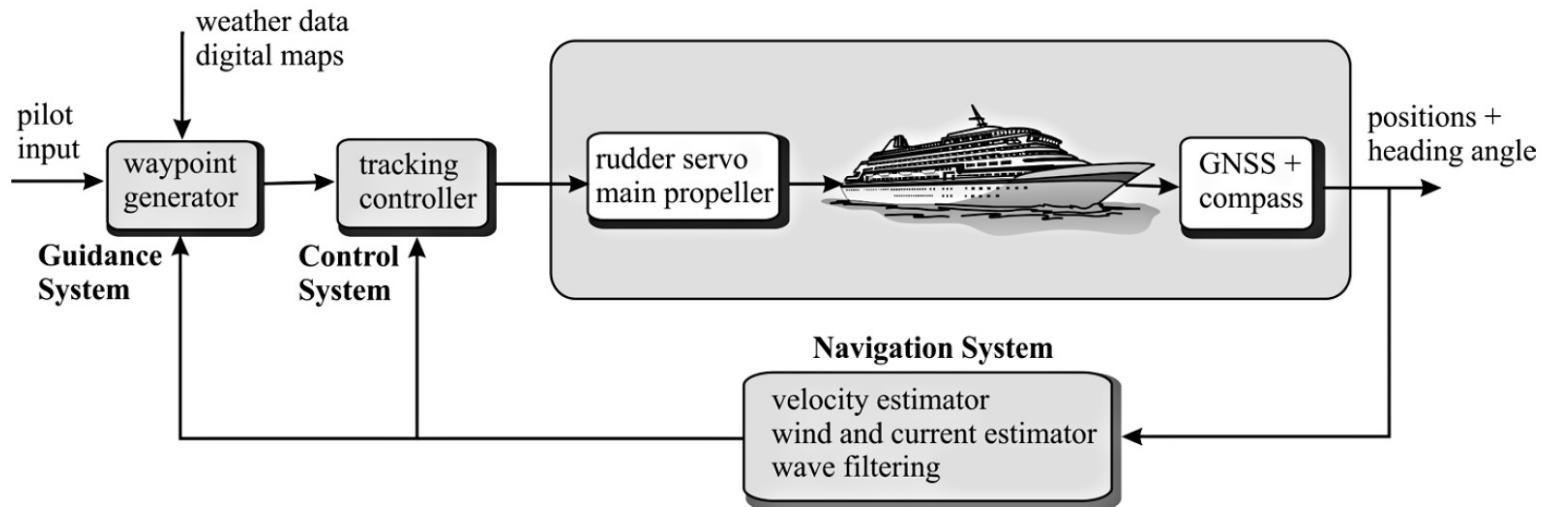
1. Specify a matrix of bandwidths: $\Omega_b = \text{diag}\{\omega_{b_1}, \omega_{b_2}, \dots, \omega_{b_6}\} > 0$
2. Specify a matrix of relative damping ratios: $Z = \text{diag}\{\zeta_1, \zeta_2, \dots, \zeta_6\} > 0$
3. Compute the natural frequencies: $\omega_{n_i} = \frac{1}{\sqrt{1-2\zeta_i^2+\sqrt{4\zeta_i^4-4\zeta_i^2+2}}} \omega_{b_i}$
 $\Omega_n = \text{diag}\{\omega_{n_1}, \omega_{n_2}, \dots, \omega_{n_6}\} > 0$
4. Compute the P gain matrix: $K_p = M^*(\eta) \Omega_n^2$
5. Compute the D gain matrix: $K_d = 2M^*(\eta) Z \Omega_n$
 (compensation of $C(\nu) + D(\nu)$ is optionally) $-J_{\Theta}^{-\top}(\eta) [C(\nu) + D(\nu)] J_{\Theta}^{-1}(\eta)$
6. Compute the I gain matrix: $K_i = \frac{1}{10} K_p \Omega_n$

mssSimulink



15.3.4 Case study: Heading Autopilot for Marine Craft

Block diagram showing a modern autopilot system



15.3.4 Case study: Heading Autopilot for Marine Craft

The principal blocks of a heading angle autopilot system are:

- **PID Feedback Control:** The control system provides the necessary feedback signal to track the desired yaw angle ψ_d . The output is the yaw moment τ_N .
- **Compass/Yaw Gyro:** The compass measures the yaw angle ψ which is needed for feedback. In some cases, a yaw gyro is available for yaw rate feedback—that is, feedback from r .
- **Observer/Wave Filter:** In its simplest form the 1st-order wave-induced motion components ψ_w and r_w are filtered out from the measurements $y_1 = \psi + \psi_w$ and $y_2 = r + r_w$, and consequently prevented from entering the feedback loop. This is known as wave filtering where the output of the filter is the LF motion components ψ and r . This is necessary to avoid excessive rudder action. In cases where y_2 is not measured the wave filter must be constructed as a state observer so that r can be estimated from the yaw angle measurement y_1 .
- **Wind Feedforward:** In cases where a wind sensor is available for wind speed and direction, a wind model can be used for wind feedforward. This is often advantageous since the integral action term in the PID controller does not have to integrate up the wind disturbance term. However, an accurate model of the wind force and moment as a function of ship speed and wind direction is needed to implement wind feedforward.
- **Reference Feedforward** using a reference model for course-changing maneuvers. Course keeping is obtained by using a constant reference signal.
- **Control Allocation:** This module distributes the output from the feedback control system, usually the yaw moment τ_N , to the actuators (rudders and in some cases propellers and thrusters) in an optimal manner.

15.3.4 Case study: Heading Autopilot for Marine Craft

Course keeping

$$\psi_d = \text{constant}$$

The main motivation for using a rate limiting element in the reference model is that the course-changing maneuver will be described by **3 phases** (positive turn):

- I:** Start of turn, acceleration ($r_d > 0$ and $0 < \dot{r}_d \leq a^{\max}$)
- II:** Steady turning ($r_d = r^{\max}$ and $\dot{r}_d = 0$)
- III:** End of turn, deceleration ($r_d > 0$ and $-a^{\max} \leq \dot{r}_d < 0$)

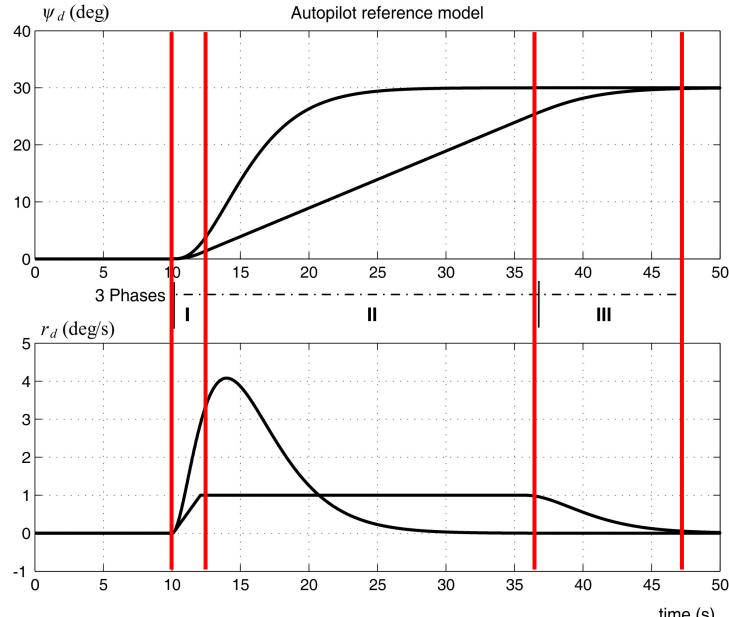
The same 3 phases applies to negative turns but with opposite signs on r_d and a_d

Reference model for course changing (turning)

$$\dot{\psi}_d = \text{sat}(r_d)$$

$$\dot{r}_d = \text{sat}(a_d)$$

$$\dot{a}_d = -(2\zeta + 1)\omega_n \text{sat}(a_d) - (2\zeta + 1)\omega_n^2 \text{sat}(r_d) + \omega_n^3(\psi_r - \psi_d)$$

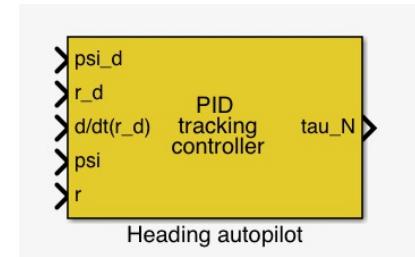


15.3.4 Case study: Heading Autopilot for Marine Craft

Yaw Dynamics (Nomoto Model)

$$(I_z - N_r)\dot{r} - N_r r = \tau_{\text{wind}} + \tau_N$$

$$\dot{r} + \frac{1}{T}r = \frac{1}{m}(\tau_{\text{wind}} + \tau_N) \quad T = \frac{m}{d} = \frac{I_z - N_r}{-N_r} \quad \tau_N = N_\delta \delta$$



mssSimulink

Control System (PID Controller with Reference and Wind Feedforward)

$$\tau_N(s) = -\hat{\tau}_{\text{wind}} + \tau_{\text{FF}}(s) - \underbrace{K_p \left(1 + T_d s + \frac{1}{T_i s} \right) \tilde{\psi}(s)}_{\tau_{\text{PID}}}$$

where $\tilde{r} := r - r_d$ and $\tilde{\psi} := \psi - \psi_d$.

Time domain

$$\tau_N = -\hat{\tau}_{\text{wind}} + \tau_{\text{FF}} - K_p \text{ssa}(\tilde{\psi}) - \underbrace{K_p T_d}_{K_d} \tilde{r} - \underbrace{\frac{K_p}{T_i}}_{K_i} \int_0^t \text{ssa}(\tilde{\psi}(\tau)) d\tau$$

$$\tau_{\text{FF}} = m \left(\dot{r}_d + \frac{1}{T} r_d \right)$$

$$\omega_n = \frac{1}{\sqrt{1 - 2\zeta^2 + \sqrt{4\zeta^4 - 4\zeta^2 + 2}}} \omega_b$$

$$\begin{aligned} K_p &= m\omega_n^2 \\ K_d &= m \left(2\zeta\omega_n - \frac{1}{T} \right) \xrightarrow{T \gg 0} 2\zeta\omega_n m \\ K_i &= \frac{\omega_n}{10} K_p \end{aligned}$$

15.3.4 Case study: Heading Autopilot for Marine Craft

Wind Feedforward

$$\hat{\tau}_{\text{wind}} = \frac{1}{2} \rho_a V_{rw}^2 C_N(\gamma_{rw}) A_{Lw} L_{oa}$$

$$V_{rw} = \sqrt{u_{rw}^2 + v_{rw}^2}$$

$$\gamma_{rw} = -\text{atan2}(v_{rw}, u_{rw})$$

$$u_{rw} = u - u_w = u - V_w \cos(\beta_{V_w} - \psi)$$

$$v_{rw} = v - v_w = v - V_w \sin(\beta_{V_w} - \psi)$$

Wind feedforward is frequently used since PID integral action does not have to integrate up the wind disturbance term.

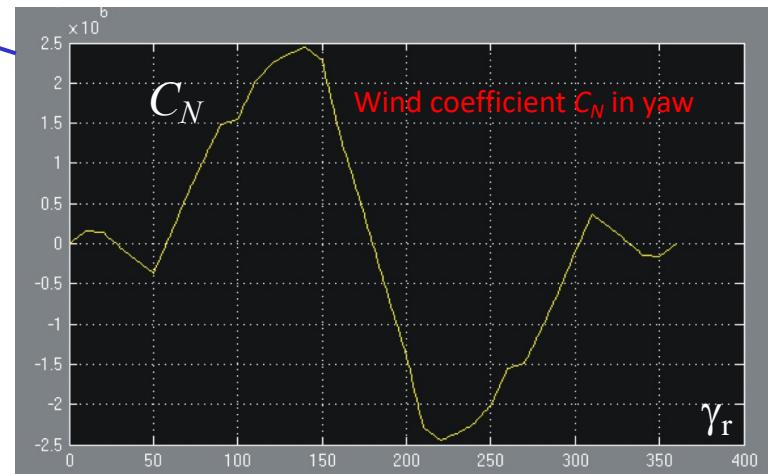
However, an accurate model of the wind moment as a function of measured wind speed and direction is needed to successfully implement wind feedforward.

V_{rw} is the relative wind speed

γ_{rw} is the relative wind angle of attack

u_{rw} and v_{rw} are the relative wind velocities

Anemometer measurements: Wind speed and direction V_w and β_{V_w}



```
function [tau_w,CX,CY,CK,CN] = blendermann94(gamma_r,V_r,AFw,ALw,sH,sL,Loa,vessel_no)
% [tau_w,CX,CY,CK,CN] = blendermann94(gamma_r,V_r,AFw,ALw,sH,sL,Loa,vessel_no) returns
% force/moment vector w_wind = [tauX,tauY,tauN] and the optionally wind coeffisients
% cx,cy and cn for merchant ships using the formulas of Isherwood (1972).
```

15.3.4 Case study: Heading Autopilot for Marine Craft

Control Allocation

The yaw moment can be generated by a **single rudder**

$$\tau_N = N_\delta \delta$$

or **several actuators** u_i ($i=1 \dots r$) satisfying

$$\tau_N = \mathbf{b}^\top \mathbf{u}, \quad \mathbf{u} = [u_1, \dots, u_r]^\top$$

For a rudder-controlled craft, the input command is computed

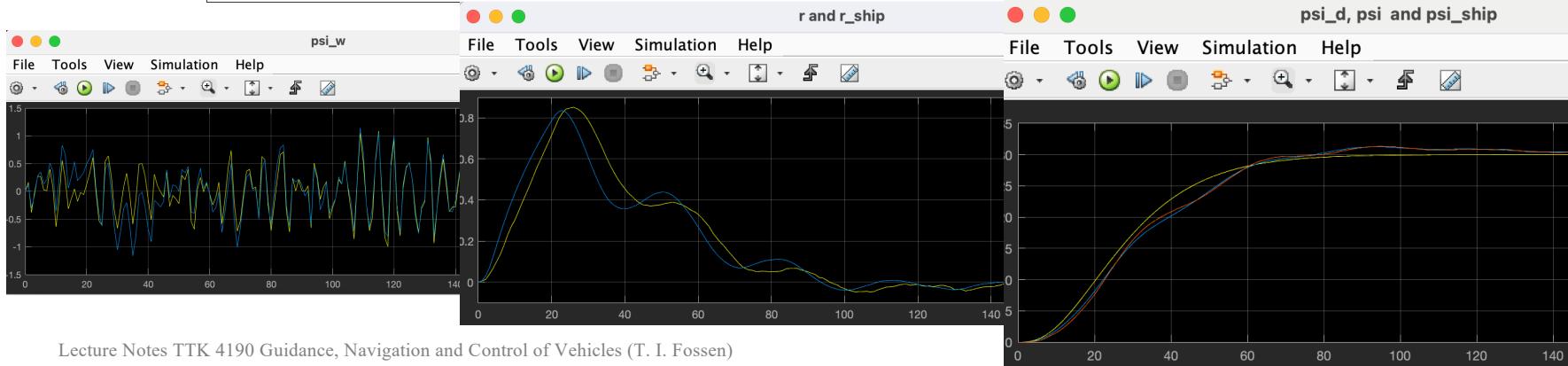
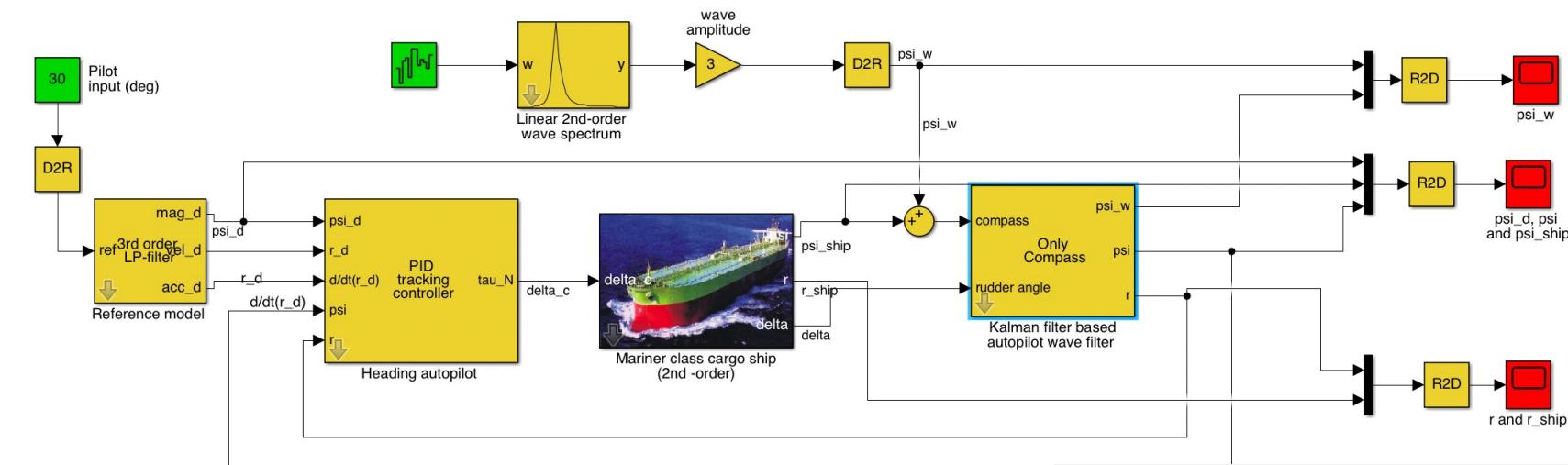
$$\delta = \frac{1}{N_\delta} \tau_N$$

In the case of several actuators we can use the generalized inverse to compute \mathbf{u} if the scalar $\mathbf{b}^\top \mathbf{b} \neq 0$ (see Section 11.2). This gives

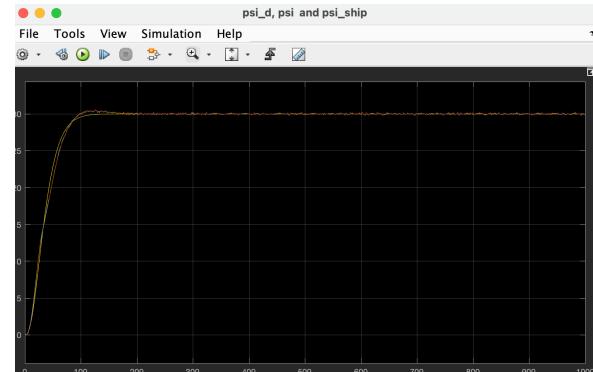
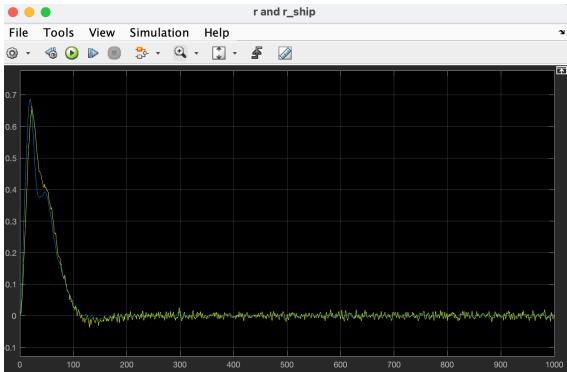
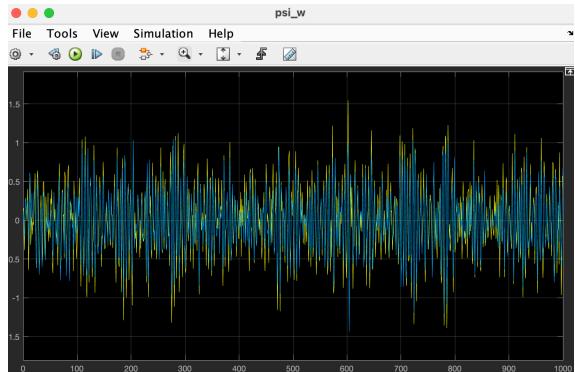
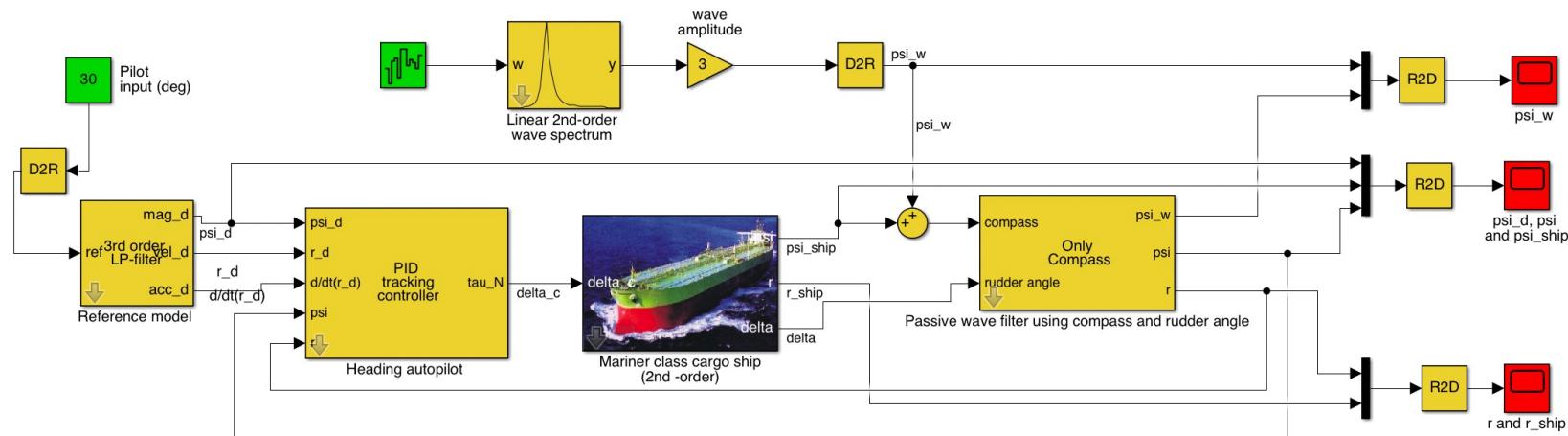
$$\mathbf{u} = \mathbf{b}(\mathbf{b}^\top \mathbf{b})^{-1} \tau_N$$



MSS Simulink: demoKalmanWavefilterAutop.slx

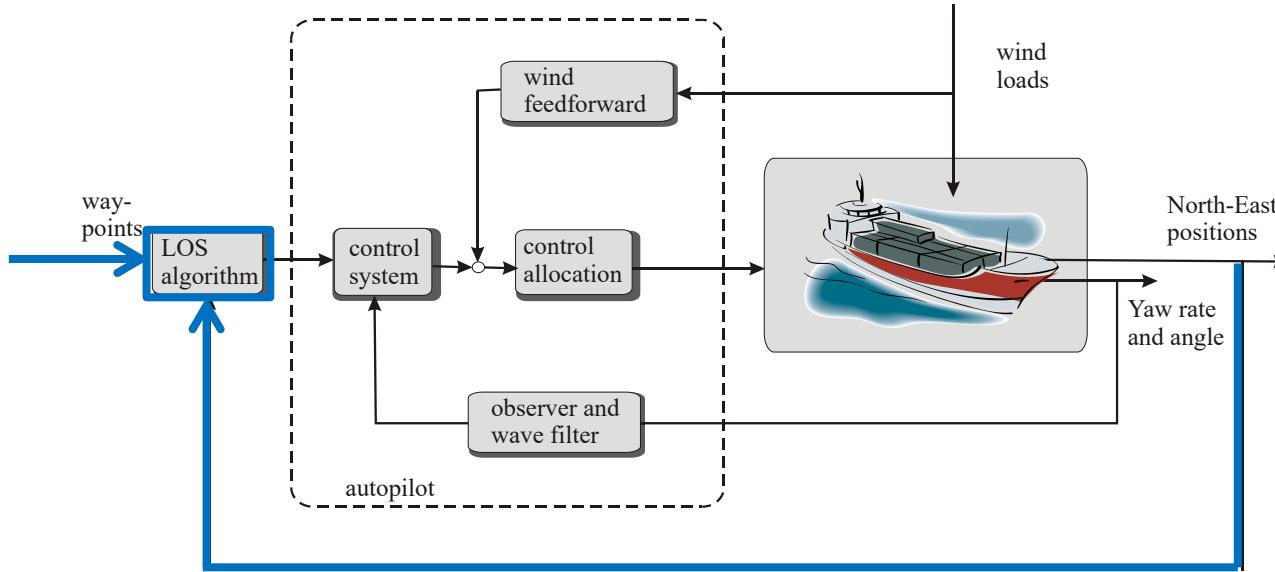


MSS Simulink: demoPassiveWavefilterAutopilot1.slx

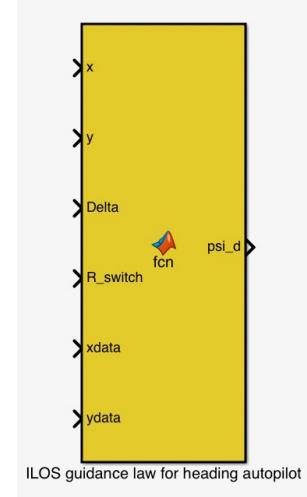


15.3.5 Case Study: LOS Path-Following Control for Marine Craft

A line-of-sight (LOS) path-following controller can be designed for conventional craft by adding a **kinematic control loop** as shown in the figure.



mssSimulink



15.3.5 Case Study: LOS Path-Following Control for Marine Craft

Consider a path defined in terms of two waypoints using the Cartesian coordinates $(x_i, y_i) \in \mathbb{R}^2$

Waypoint guidance systems can be designed as trajectory tracking controllers. In its simplest form this involves the use of a classical autopilot system where the yaw angle command ψ_d is generated by ILOS such that the cross-track error is minimized.

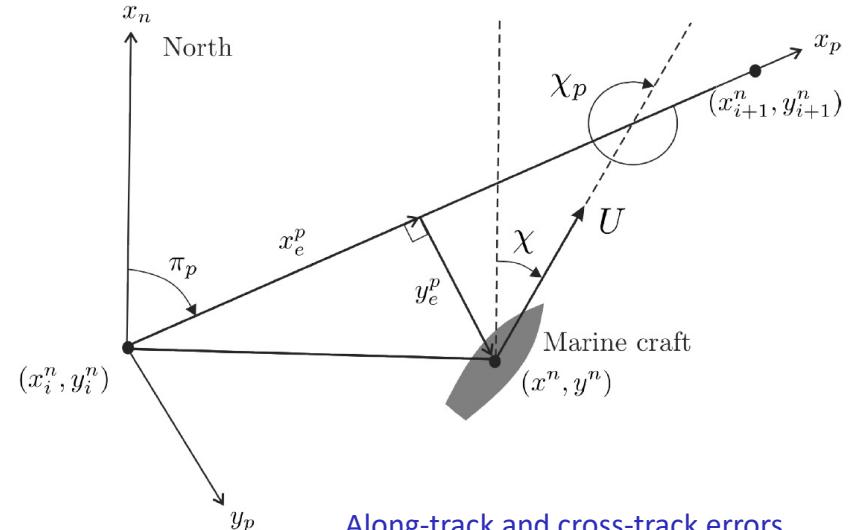
$$\begin{aligned}\psi_d &= \pi_p - \tan^{-1}(K_p y_e^p + K_i y_{\text{int}}^p) \\ \dot{y}_{\text{int}}^p &= \frac{\Delta y_e^p}{\Delta^2 + (y_e^p + \kappa y_{\text{int}}^p)^2}\end{aligned}$$

Path-fixed reference frame $\{p\}$ with origin in p_i and x -axis rotated a positive angle

$$\pi_p = \text{atan2}(y_{i+1}^n - y_i^n, x_{i+1}^n - x_i^n)$$

Consequently, the error term y_e^p represents the deviation to the path in the y -direction expressed in $\{p\}$

$$y_e^p = -\sin(\pi_p)(x^n - x_i^n) + \cos(\pi_p)(y^n - y_i^n)$$



Along-track and cross-track errors

$$\begin{bmatrix} x_e^p \\ y_e^p \end{bmatrix} = \mathbf{R}_p^n(\pi_p)^\top \left(\begin{bmatrix} x^n \\ y^n \end{bmatrix} - \begin{bmatrix} x_i^n \\ y_i^n \end{bmatrix} \right)$$

15.3.5 Case Study: LOS Path-Following Control for Marine Craft

Heading autopilot

The heading autopilot is usually chosen as a PID controller with wind and reference feedforward (see Section 15.3.4)

$$\begin{aligned}\tau_N &= -\hat{\tau}_{\text{wind}} + \tau_{\text{FF}} - K_p \text{ssa}(\tilde{\psi}) - K_d \dot{\tilde{\psi}} - K_i \int_0^t \text{ssa}(\tilde{\psi}(\tau)) d\tau \\ \tau_{\text{FF}} &= m \left(\dot{r}_d + \frac{1}{T} r_d \right)\end{aligned}$$

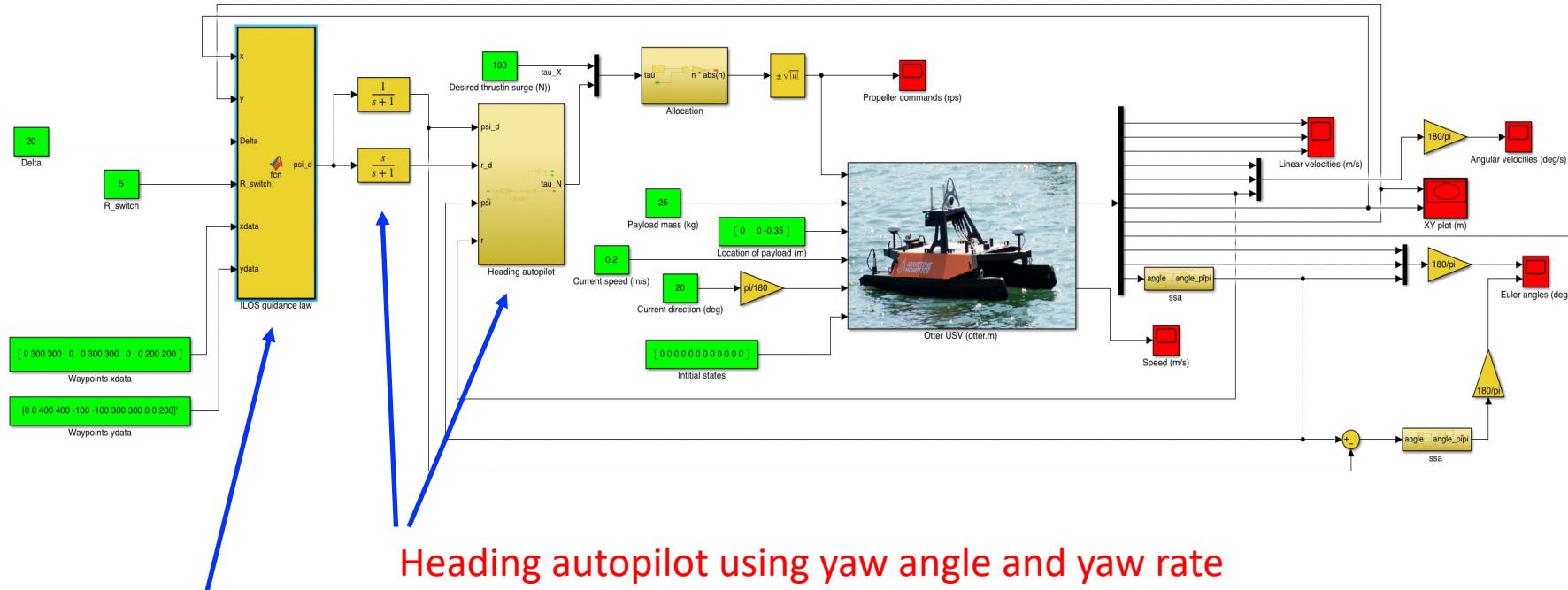
Waypoint switching mechanism

When moving along a piecewise linear path made up of N straight-line segments connected by $N + 1$ waypoints, a switching mechanism for selecting the next waypoint is needed. Hence, if the craft is located inside the circle

$$(x_{i+1}^n - x^n)^2 + (y_{i+1}^n - y^n)^2 \leq R_{i+1}^2$$

the next waypoint is chosen.

MSS Simulink: demoOtterUSVPathFollowingHeadingControl.slx



ILOS guidance law for heading autopilot

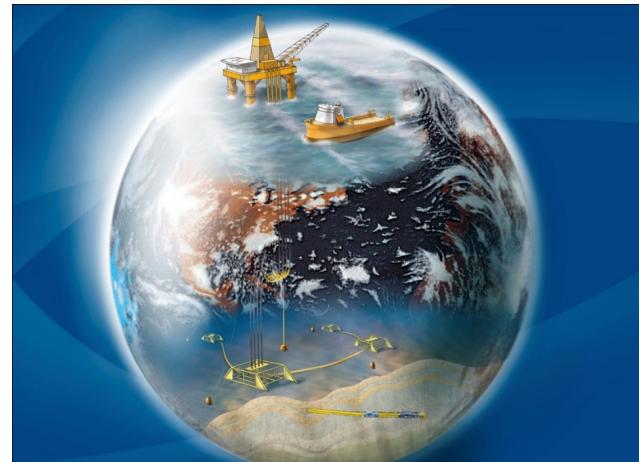
15.3.6 Case Study: Dynamic Positioning System for Surface Vessels

Control systems for stationkeeping and low-speed maneuvering are commonly known as **dynamic positioning (DP) systems**. They are designed for simultaneously control of the three horizontal motions (surge, sway, and yaw) and they are used in a wide range of marine operations such as stationkeeping, drilling and offloading.

The classification society DNV GL defines a dynamically positioned vessel as a free-floating vessel which maintains its position (fixed location or predetermined track) exclusively by means of thruster.

It is, however, possible to exploit rudder forces in DP by using the propeller to generate rudder lift forces.

The first DP systems were designed using conventional PID controllers in cascade with LP and/or notch filters to suppress the wave-induced forces. This was done by neglecting coupling terms (Sargent and Cowgill 1976, Morgan 1978). From the middle of the 1970's a new model-based control concept utilizing stochastic optimal control theory and Kalman filtering techniques was employed with the DP problem by Balchen (1976).



15.3.6 Case Study: Dynamic Positioning System for Surface Vessels

LTV DP model

$$\begin{aligned}\dot{\eta} &= \mathbf{R}(t)\nu \\ M\dot{\nu} + D\nu &= \mathbf{R}^\top(t)\mathbf{b} + \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{wind}} + \boldsymbol{\tau}_{\text{wave}} \\ \dot{\mathbf{b}} &= 0\end{aligned}$$



$\mathbf{R}(t) := \mathbf{R}_{z,\psi}(\psi(t))$ is assumed known thanks to gyrocompass measurements.

DP control system with wind feedforward

$$\boldsymbol{\tau} = -\hat{\boldsymbol{\tau}}_{\text{wind}} - \mathbf{R}^\top(t)\mathbf{K}_p\tilde{\eta} - \underbrace{\mathbf{R}^\top(t)\mathbf{K}_d\mathbf{R}(t)}_{\mathbf{K}_d^*}\nu - \mathbf{R}^\top(t)\mathbf{K}_i \int_0^t \tilde{\eta}(\tau) d\tau$$

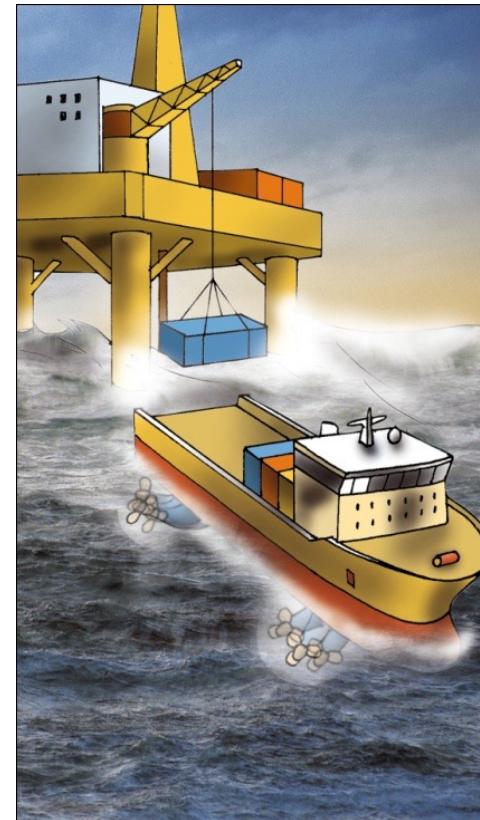
where $\mathbf{K}_d^* := \mathbf{R}^\top(t)\mathbf{K}_d\mathbf{R}(t)$. It is common to choose \mathbf{K}_d as a diagonal matrix and thus $\mathbf{K}_d^* \equiv \mathbf{K}_d$. For the full-state feedback case, asymptotic stability follows using Lyapunov arguments (see Section 15.3.3).



15.3.6 Case Study: Dynamic Positioning System for Surface Vessels

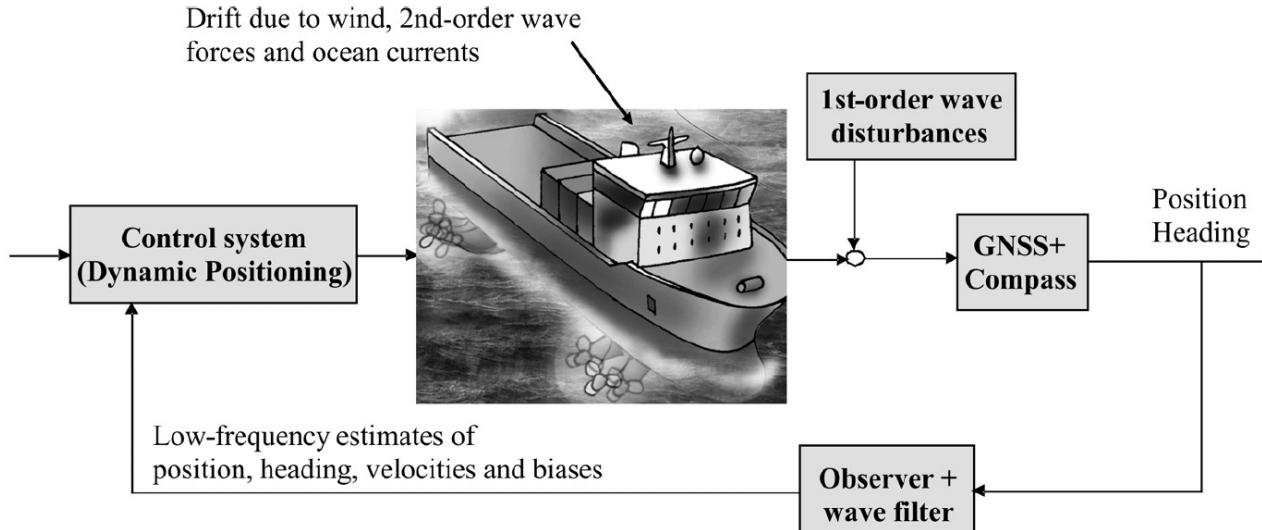
In commercial DP systems it is necessary to include the following features:

- ❖ **Integral action** to compensate for slowly-varying disturbances (bias term b) due to ocean currents and wave drift forces (2nd-order wave-induced forces).
- ❖ **Wind feedforward** control to compensate for mean wind disturbances. Wind gust cannot be compensated for since the actuators do not have the capacity of moving a large vessel in the frequency range of the wind gust.
- ❖ **Wave filtering** to avoid that 1st-order wave-induced motions feed back to the control system. This is an important feature since the actuators cannot move a large vessel fast enough to suppress the disturbances.
- ❖ **State estimator** for noise filtering and estimation of unmeasured states, for instance linear and angular velocity. The main tool for this is the Kalman filter, alternatively nonlinear and passive observers.
- ❖ **Optimal control allocation** of thrust where the main goal is to compute optimal set-points for thrusters, rudders and other actuators based on the force/moment commands generated by the DP control system.



Copyright © Bjarne Stenberg

15.3.6 Case Study: Dynamic Positioning System for Surface Vessels



In order to implement the nonlinear PID controller a state estimator and wave filter must be designed. This is straightforward for the LTV model where additional states for the WF motions can be augmented and used directly in a Kalman filter (see Section 13.4.6) or by using a passive observer (see Section 13.5):

$$\begin{aligned}
 \dot{\hat{\xi}} &= A_w \hat{\xi} + K_1(\omega_0) \tilde{y} \\
 \dot{\hat{\eta}} &= R(t) \hat{\nu} + K_2 \tilde{y} \\
 \dot{\hat{b}} &= -T^{-1} \hat{b} + K_3 \tilde{y} \\
 M \dot{\hat{\nu}} &= -D \hat{\nu} + R^\top(t) \hat{b} + \tau + \hat{\tau}_{\text{wind}} + R^\top(t) K_4 \tilde{y} \\
 \hat{y} &= \hat{\eta} + C_w \hat{\xi}
 \end{aligned}$$

15.3.6 Case Study: Dynamic Positioning System for Surface Vessels

Wind Feedforward

$$\tau = -\hat{\tau}_{\text{wind}} - \mathbf{R}^{\top}(t)\mathbf{K}_p\tilde{\boldsymbol{\eta}} - \underbrace{\mathbf{R}^{\top}(t)\mathbf{K}_d\mathbf{R}(t)}_{\mathbf{K}_d^*}\boldsymbol{\nu} - \mathbf{R}^{\top}(t)\mathbf{K}_i \int_0^t \tilde{\boldsymbol{\eta}}(\tau) d\tau$$

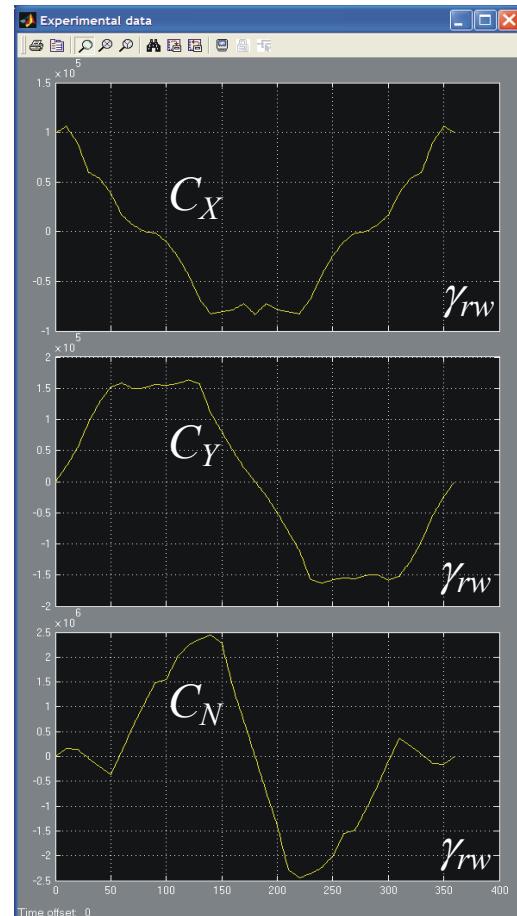
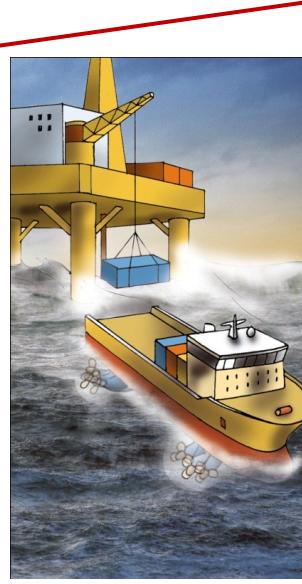
$$\hat{\tau}_{\text{wind}} = \frac{1}{2} \rho_a V_{rw}^2 \begin{bmatrix} C_X(\gamma_{rw}) A_{F_w} \\ C_Y(\gamma_{rw}) A_{L_w} \\ C_N(\gamma_{rw}) A_{L_w} L_{oa} \end{bmatrix}$$

$$V_{rw} = \sqrt{u_{rw}^2 + v_{rw}^2}$$

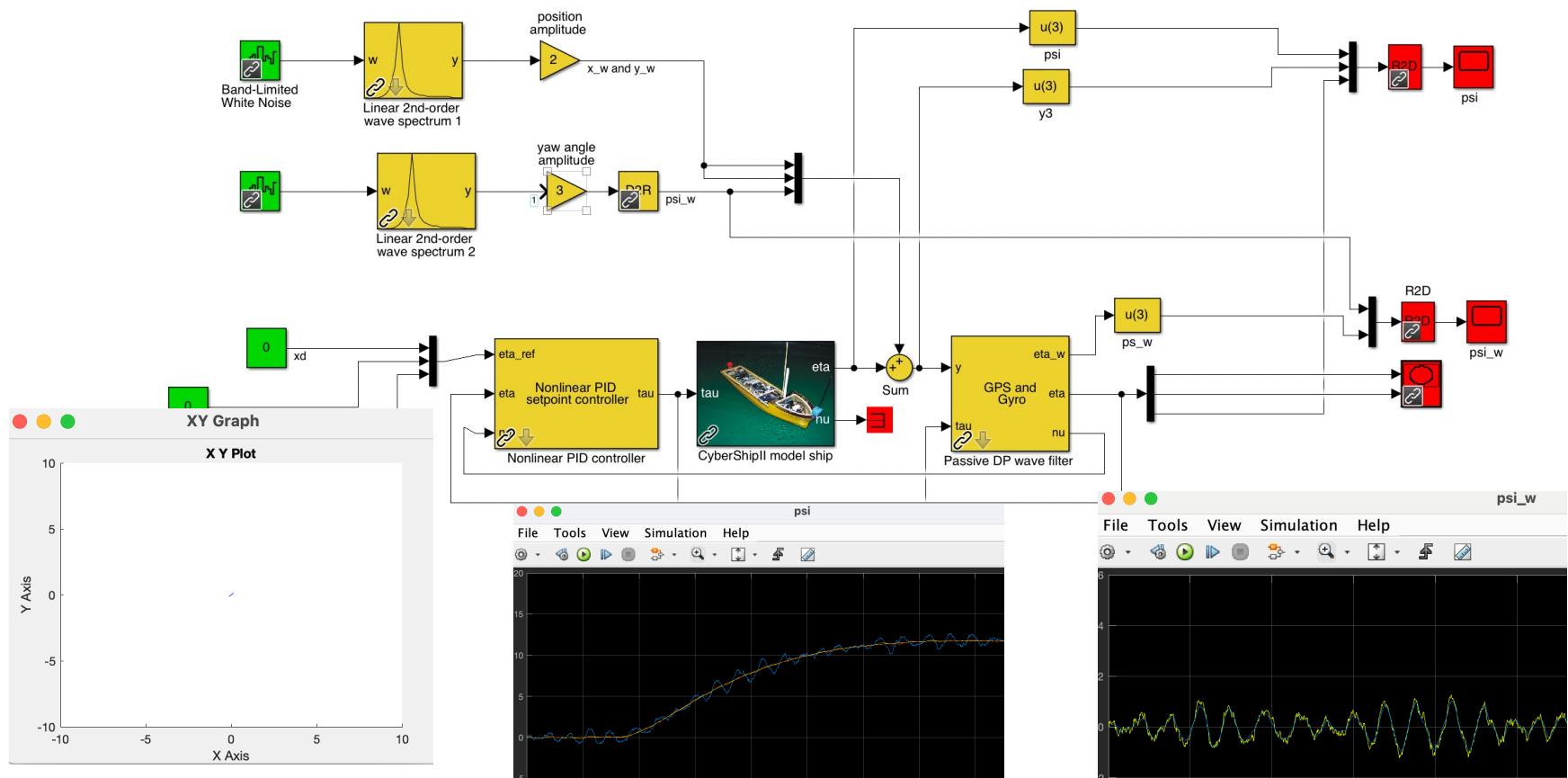
$$\gamma_{rw} = -\text{atan2}(v_{rw}, u_{rw})$$

$$u_{rw} = u - u_w = u - V_w \cos(\beta_{V_w} - \psi)$$

$$v_{rw} = v - v_w = v - V_w \sin(\beta_{V_w} - \psi)$$



MSS Simulink: demoCS2passiveObserverDP.slx



15.3.7 Case Study: Position Mooring System for Surface Vessels

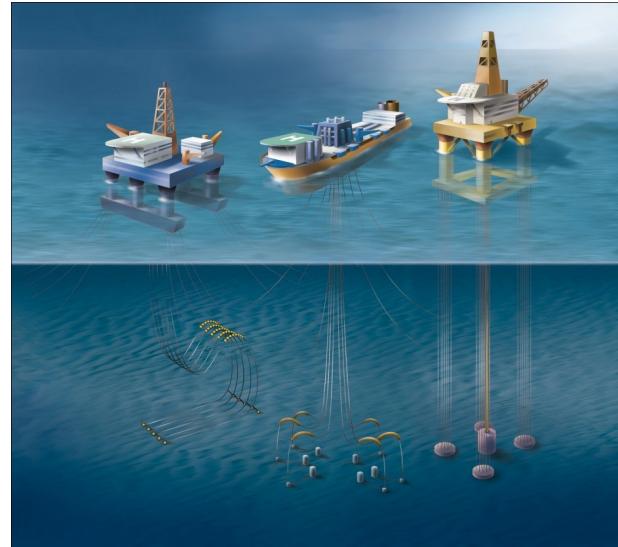
$$\underbrace{\mathbf{R}(t) M \mathbf{R}^\top(t)}_{M^*(t)} \ddot{\boldsymbol{\eta}} + \underbrace{\mathbf{R}(t) D \mathbf{R}^\top(t)}_{D^*(t)} \dot{\boldsymbol{\eta}} + \mathbf{K} \boldsymbol{\eta} = \mathbf{b} + \mathbf{R}(t) \boldsymbol{\tau} + \mathbf{R}(t) \boldsymbol{\tau}_{\text{wind}} + \mathbf{R}(t) \boldsymbol{\tau}_{\text{wave}}$$

The additional spring $\mathbf{K}\boldsymbol{\eta}$ due to the [mooring system](#) adds spring stiffness in surge, sway and yaw described by the parameters $k_{11} > 0$, $k_{22} > 0$ and $k_{33} \geq 0$.

$$\mathbf{M} = \mathbf{M}^\top = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix}$$

$$\mathbf{D} = \mathbf{D}^\top = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix}$$

$$\mathbf{K} = \text{diag}\{k_{11}, k_{22}, k_{33}\}$$

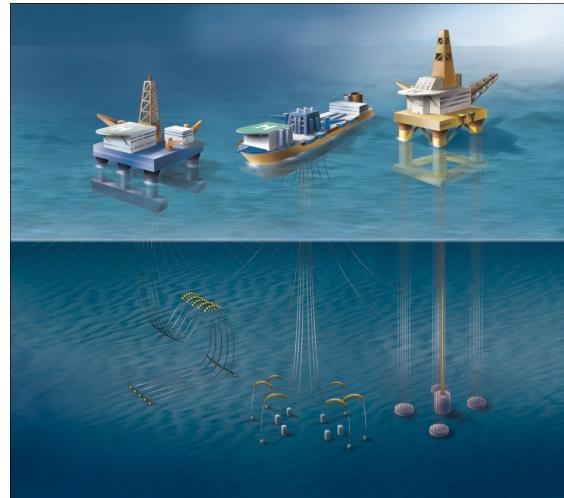


Copyright © Bjarne Stenberg

15.3.7 Case Study: Position Mooring System for Surface Vessels

Turret mooring systems have cables that are connected to the turret via bearings. This allows the vessel to rotate around the anchor legs and the rotational spring can be neglected ($k_{33} = 0$). The turret can be mounted either internally or externally. An external turret is fixed, with appropriate reinforcements, to bow or stern of the ship. In the internal case the turret is placed within the hull in a moon pool. Turret mooring systems allows the vessel to rotate in the horizontal plane (yaw) into the direction where environmental loading due to wind, waves and currents is minimal. This is referred to as **weathervaning**.

Spread mooring systems are used to moor **Floating Production, Storage and Offloading (FPSO)** units, tankers and floating platforms. The system consists of mooring lines attached somewhere to the vessel. The drawback with a spread mooring system is that it restrains the vessel from rotating ($k_{33} > 0$) and hence weathervaning is impossible.



Copyright © Bjarne Stenberg

15.3.7 Case Study: Position Mooring System for Surface Vessels

For **thruster assisted PM systems** the thrusters are complementary to the mooring system and the main idea is to provide the system with additional damping by using a **D-controller**

$$\begin{aligned}\tau &= -\mathbf{R}^\top(t)\mathbf{K}_d\dot{\boldsymbol{\eta}} \\ &= -\mathbf{R}^\top(t)\mathbf{K}_d\mathbf{R}(t)\boldsymbol{\nu}\end{aligned}$$

Closed-loop system

$$\mathbf{M}^*(t)\ddot{\boldsymbol{\eta}} + (\mathbf{D}^*(t) + \mathbf{K}_d)\dot{\boldsymbol{\eta}} + \mathbf{K}\boldsymbol{\eta} = \mathbf{b} + \mathbf{R}(t)\boldsymbol{\tau}_{\text{wind}} + \mathbf{R}(t)\boldsymbol{\tau}_{\text{wave}}$$

Integral action is not used in PM systems, since the ship is allowed to move within a limited radius about the equilibrium point or **field-zero point (FZP)**.

If the vessel, moves outside the specified radius of the mooring system, a stabilizing control system of PD-type can be used to drive the vessel inside the circle again. It is optimal to use additional thrust to stay on the circle rather than move the vessel to the FZP. In good weather, no control action is needed since the vessel is free to move within the circle.



Chapter Goals – Revisited

Open-loop stability and maneuverability

- **Stability vs. maneuverability.** Be able to explain the interplay between stability and maneuverability.
- Be able to explain the concepts of **straight-line**, **directional** and **positional motion stability** and how these classical approaches relate to eigenvalues and time constants in a feedback control system.
- Be familiar with **the classical ship maneuvers** and their use, including the:
 - Turning circle
 - Kempf's zigzag maneuver
 - Pull-out maneuver
 - Dieudonné's spiral maneuver
 - Bech's reverse spiral maneuver
 - Stopping trials

State-of-the-art autopilot design

- Be able to design **classical heading** and **course autopilots** using:
 - Successive-loop closure
 - PID control based on pole placement
- Be able to design more advanced control systems such as
 - **Depth and diving control systems**
 - **Path-following control systems**
 - **3-DOF dynamic positioning systems**