Goal Oriented Training in Theory
GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Systems & Control Fundamentals

Gianmaria De Tommasi[1]

[1]Associazione EURATOM/ENEA/CREATE
DIS – Università degli Studi di Napoli Federico II
detommas@unina.it

November 20th 2009
4th GOTiT High Level Course
Magnetic Control of Tokamak Plasmas

Goal Oriented Training in Theory

CREATE

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Outline

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

CREATE

Associazione
EURATOM–ENEA
sulla Fusione

Preliminaries   Linear Systems

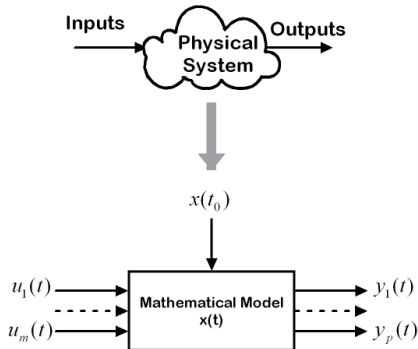## Dynamical Systems

State-space model

A **finite dimensional** *continuous-time* dynamical system can be described by the following differential equations:

$$\dot{x}(t) = f\big(x(t), u(t), t, t_0\big), x(t_0) = x_0 \quad (1a)$$

$$y(t) = \eta\big(x(t), u(t), t, t_0\big) \quad (1b)$$

where:

- $x(t) \in \mathbb{R}^n$ is the system **state**
- $x(t_0) \in \mathbb{R}^n$ is the **initial condition**
- $u(t) \in \mathbb{R}^m$ is the **input** vector
- $y(t) \in \mathbb{R}^p$ is the **output** vector

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Linear Systems

Linear time-invariant systems

A **linear time-invariant (LTI)** continuous-time system is described by

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0 \tag{2a}$$
$$y(t) = Cx(t) + Du(t) \tag{2b}$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ and $D \in \mathbb{R}^{p \times m}$.

A dynamical system with single-input ($m = 1$) and single-output ($p = 1$) is called *SISO*, otherwise it is called *MIMO*.

Matlab commands

sys = ss(A,B,C,D) creates a state-space model object.
y = lsim(sys,u,t) simulates the the time response of the LTI system **sys**.

Equilibria of nonlinear dynamical systems

Consider a nonlinear and **time-invariant** system

$$\dot{x}(t) = f(x(t), u(t)), \quad x(0) = x_0 \tag{3a}$$
$$y(t) = \eta(x(t), u(t)) \tag{3b}$$

If the input is constant, i.e. $u(t) = \bar{u}$, then the equilibria
$x_{e_1}, x_{e_2}, \ldots, x_{e_q}$ of such a system can be computed as solutions of the
homogeneous equation

$$f(x_e, \bar{u}) = 0,$$

Given an equilibrium x_{e_i} the correspondent output is given by

$$y_{e_i} = \eta(x_{e_i}, \bar{u}).$$

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione EURATOM-ENEA sulla Fusione

GOTiT

Preliminaries   Linear Systems

Linearization around a given equilibrium

If $x_0 = x_e + \delta x_0$ and $u(t) = \bar{u} + \delta u(t)$, with δx_0, $\delta u(t)$ **sufficiently small**, then the behaviour of (3) around a given equilibrium $(\bar{u}, x_e)$ is well described by the linear system
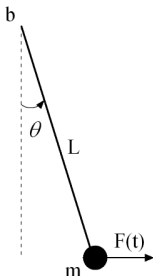
$$\delta \dot{x}(t) = \left.\frac{\partial f}{\partial x}\right|_{\substack{x = x_e \\ u = \bar{u}}} \delta x(t) + \left.\frac{\partial f}{\partial u}\right|_{\substack{x = x_e \\ u = \bar{u}}} \delta u(t), \quad \delta x(0) = \delta x_0 \quad \text{(4a)}$$

$$\delta y(t) = \left.\frac{\partial \eta}{\partial x}\right|_{\substack{x = x_e \\ u = \bar{u}}} \delta x(t) + \left.\frac{\partial \eta}{\partial u}\right|_{\substack{x = x_e \\ u = \bar{u}}} \delta u(t) \quad \text{(4b)}$$

The total output can be computed as

$$y(t) = \eta(x_e, \bar{u}) + \delta y(t).$$

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Preliminaries   Linear Systems

Example - Pendulum

- Mass *m*
- Length *L*
- Friction coefficient *b*

Goal Oriented Training in Theory

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Preliminaries   Linear Systems

Example - Pendulum

Nonlinear model

Let

$$x(t) = \left(\begin{array}{c} \theta(t) \\ \dot{\theta}(t) \end{array} \right) \qquad u(t) = F(t) \qquad y(t) = \theta(t)$$

then

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{g}{L} \sin x_1(t) - \frac{b}{mL^2} x_2(t) + \frac{1}{mL} \cos x_1(t) u(t)$$
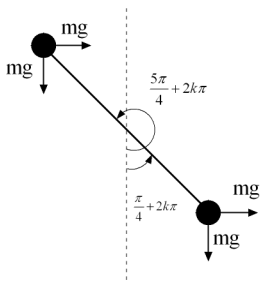$$y(t) = x_1(t)$$

Example - Pendulum

Equilibria

If $\bar{u} = mg$, solving $f(x_e, \bar{u}) = 0$ we get

$$x_{e_k} = \left(\begin{array}{c} \frac{\pi}{4} + k\pi \\ 0 \end{array} \right)$$

Example - Pendulum

Around the equilibria $x_e = \left(\frac{\pi}{4} \ 0\right)^T$ the behaviour of the pendulum is well described by the linear system

$$
\begin{aligned}
\delta \dot{x}_1(t) &= \delta x_2(t) \\
\delta \dot{x}_2(t) &= -\frac{\sqrt{2}g}{L}\delta x_1(t) - \frac{b}{mL^2}\delta x_2(t) + \frac{1}{\sqrt{2}mL}\delta u(t) \\
\delta y(t) &= \delta x_1(t)
\end{aligned}
$$

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Preliminaries   Linear Systems

The script `pendulum.m` - 1

```
% System parameters
m = 10;
b = 15;
g = 9.81;
L = 1;
% Equilibria
xbar = [pi/4 0];
ubar = m*g;
% Output at the equilibria
ybar = xbar(1);
% Variation of the initial conditions
dx0 = [0.3 3];
% Input variation:  Am*sin(t)
Am = 10;
% Simulation time interval
tfin = 15;
```

The script `pendulum.m` - 2

```
% Linearized model matrices
A = [ 0 1 ; -sqrt(2)*g/L -b/(m*L^2) ];
B = [ 0 ; 1/(sqrt(2)*m*L) ];
C = [ 1 0 ];
D = 0;
% Linearized model
sys_l = ss(A,B,C,D);
% Time vector
tlin = 0:.001:tfin;
% Linear simulation
du = Am*sin(tlin);
ylin= lsim(sys_l,du,tlin,dx0);
```

The script `pendulum.m` - 3

```
% Nonlinear simulation
[ t,x_nl ]=ode45('p_nl',[0 tfin],xbar+dx0,[],m,b,g,L,ubar,Am);
y_nl = x_nl(:,1);
% Plots
figure(1)
plot(t,y_nl*180/pi,'-',tlin,(ylin+ybar)*180/pi,'-')
grid on
ylabel('[deg]')
xlabel('tempo [s]')
title('theta')
legend('NL','L')
```

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Linear Systems

The function `p_nl.m`

```
function xdot = p_nl(t,x,flag,m,b,g,L,ubar,Am)
%
% Returns the state derivative
%
u = ubar + Am*sin(t);
xdot = [x(2); -g/L*sin(x(1)) - b/(m*L^2)*x(2) +...
u/(m*L)*cos(x(1))];
```
▶ Download Matlab example

Asymptotic stability of LTI systems

Asymptotic stability

This property roughly asserts that every solution of $\dot{x}(t) = Ax(t)$ tends to zero as $t \rightarrow \infty$.

Note that for LTI systems the stability property is related to the system and not to a specific equilibrium.

Theorem - System (2) is **asymptotically stable iff** A is <u>Hurwitz</u>, that is if every eigenvalue λ_i of A has strictly negative real part

$$\Re(\lambda_i) < 0 \,, \forall\, \lambda_i \,.$$

Theorem - System (2) is **unstable if** A has at least one eigenvalue $\bar{\lambda}$ with strictly positive real part, that is

$$\exists\, \bar{\lambda} \text{ s.t. } \Re(\bar{\lambda}) > 0 \,.$$

Theorem - Suppose that A has all eigenvalues λ_i such that $\Re(\lambda_i) \leq 0$, then system (2) is **unstable if** there is at least one eigenvalue $\bar{\lambda}$ such that $\Re(\bar{\lambda}) = 0$ which corresponds to a Jordan block with size > 1.

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM–ENEA
sulla Fusione

GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Preliminaries   Linear Systems

Equilibrium stability for nonlinear systems

For nonlinear system the stability property is related to the specific equilibrium.

Theorem - The equilibrium state x_e corresponding to the constant input $\bar{u}$ a nonlinear system (3) is **asymptotically stable if** all the eigenvalues of the correspondent linearized system (4) have strictly negative real part.

Theorem - The equilibrium state x_e corresponding to the constant input $\bar{u}$ a nonlinear system (3) is **unstable if** there exists at least one eigenvalue of the correspondent linearized system (4) which has strictly positive real part.

Goal Oriented Training in Theory

GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Transfer function

Transfer function of LTI systems

Given a LTI system (2) the corresponding *transfer matrix* from u to y is defined as

$$Y(s) = G(s)U(s),$$

with $s \in \mathbb{C}$. $U(s)$ and $Y(s)$ are the Laplace transforms of $u(t)$ and $y(t)$ with zero initial condition ($x(0) = 0$), and

$$G(s) = C(sI - A)^{-1}B + D. \qquad (5)$$

For SISO system (5) is called *transfer function* and it is equal to the Laplace transform of the **impulsive response** of system (2) with zero initial condition.

Matlab commands

`sys = tf(num,den)` creates a transfer function object.

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Transfer function

Transfer function

Given the transfer function $G(s)$ and the Laplace transform of the input $U(s)$ the time response of the system can be computed as the inverse transform of $G(s)U(s)$, without solving differential equations.

As an example, the **step response** of a system can be computed as:

$$y(t) = \mathcal{L}^{-1}\left[G(s)\frac{1}{s}\right].$$

Matlab commands

[y,t] = step(sys) computes the step response of the LTI system **sys**.

[y,t] = impulse(sys) computes the impulse response of the LTI system **sys**.

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Preliminaries   Transfer function

Poles and zeros of SISO systems

Given a SISO LTI system , its transfer function is a rational function of s

$$G(s) = \frac{N(s)}{D(s)} = \rho \frac{\Pi_i(s - z_i)}{\Pi_j(s - p_j)},$$

where $N(s)$ and $D(s)$ are polynomial in s, with
$\deg(N(s)) \leq \deg(D(s))$.
We call

- p_j **poles** of $G(s)$
- z_i **zeros** of $G(s)$

Matlab commands

sys = zpk(z,p,k) creates a zeros-poles-gain object.
p = eig(sys) or **p = pole(sys)** return the poles of the LTI
system **sys**.
z = zero(sys) returns the zeros of the LTI system **sys**.

Poles and eigenvalues of a LTI system

Every pole of $G(s)$ is an eigenvalue of the system matrix A. However, not every eigenvalue of A is a pole of $G(s)$.

If all the poles of $G(s)$ have strictly negative real part – i.e. they are located in the left half of the s-plane (LHP) – the SISO system is said to be **Bounded–Input Bounded–Output stable (BIBO)**.

A system is BIBO stable if every bounded input to the system results in a bounded output over the time interval $[0, \infty)$.

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Preliminaries   Transfer function

Time constants, natural frequencies and damping factors

A transfer function can be also specified in terms of

- time constants (τ, T)
- natural frequencies (ω_n, α_n)
- damping factors (ξ, ζ)
- gain (μ)
- system type (i.e. number of poles/zeros in 0, g)

$$G(s) = \mu \frac{\Pi_i(1 + T_i s)\Pi_j\left(1 + 2\frac{\zeta_j}{\alpha_{n_j}}s + \frac{s^2}{\alpha_{n_j}}\right)}{s^g\Pi_k(1 + \tau_k s)\Pi_l\left(1 + 2\frac{\xi_l}{\omega_{n_l}}s + \frac{s^2}{\omega_{n_l}}\right)}.$$

Goal Oriented Training in Theory

CREATE

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Block diagrams

Block diagrams

When dealing with transfer functions, it is usual to resort to *Block diagrams* which permit to graphically represent the interconnections between system in a convenient way.
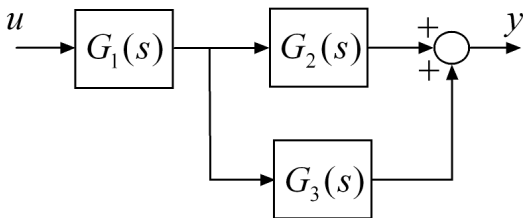
Series connection
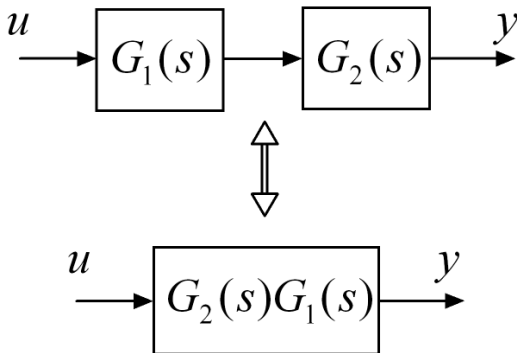
Matlab commands

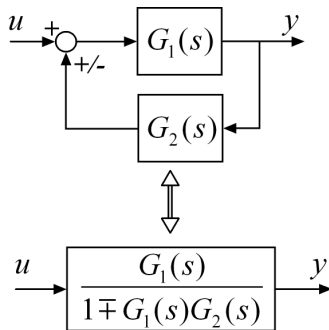`sys = series(sys1,sys2)` makes the series interconnection between `sys1` and `sys2`.

Parallel connection
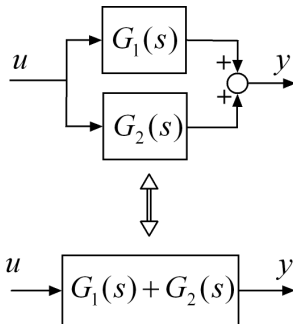
Matlab commands

`sys = parallel(sys1,sys2)` makes the parallel interconnection between `sys1` and `sys2`.

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Block diagrams

Feedback connection

Matlab commands

sys = feedback(sys1,sys2,[+1]) makes the feedback interconnection between **sys1** and **sys2**. Negative feedback is the default. If the third parameter is equal to **+1** positive feedback is applied.

Stability of interconnected systems

Given two **asymptotically stable** LTI systems $G_1(s)$ and $G_2(s)$

- the **series** connection $G_2(s)G_1(s)$ is asymptotically stable
- the **parallel** connection $G_1(s) + G_2(s)$ is asymptotically stable
- the **feedback** connection $\frac{G_1(s)}{1 \pm G_1(s)G_2(s)}$ **is not necessarely stable**

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

GOTiT

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Frequency response

Frequency response

Given a LTI system the complex function

$$G(j\omega) = C(j\omega I - A)^{-1}B + D,$$

with $\omega \in \mathbb{R}$ is called *frequency response* of the system.

$G(j\omega)$ permits to evaluate the system steady-state response to a sinusoidal input. In particular if

$$u(t) = A\sin(\bar{\omega}t + \varphi),$$

then the steady-state response of a LTI system is given by

$$y(t) = |G(j\bar{\omega})|A\sin\left(\bar{\omega}t + \varphi + \angle G(j\bar{\omega})\right).$$

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Preliminaries   Graphical representations of $G(j\omega)$

Bode plot

Given a LTI system $G(s)$ the Bode diagrams plot

- the magnitude of $G(j\omega)$ (in dB, $|G(j\omega)|_{dB} = 20 \log_{10}|G(j\omega)|$)
- and the phase of $G(j\omega)$ (in degree)

as a function of ω (in rad/s) in a semi-log scale (base 10).

Bode plots are used for both analysis and synthesis of control systems.

Matlab commands

`bode(sys)` plots the the Bode diagrams of the LTI system `sys`.
`bodemag(sys)` plots the Bode magnitude diagram of the LTI system `sys`.

Goal Oriented Training in Theory

GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Graphical representations of $G(j\omega)$

Bode plot - Example

Let consider

$$G(s) = 10\frac{1+s}{s\left(\frac{s^2}{400} + 2\frac{0.3}{20}s + 1\right)} = 10\frac{1+s}{s(0.0025s^2 + 0.03s + 1)}$$

Matlab commands

```
s = tf('s');
sys = 10*(1+s)/(s*(s^2/400+0.6*s/20+1));
bode(sys);
grid
```

Goal Oriented Training in Theory

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Graphical representations of $G(j\omega)$

Bode plot - Example
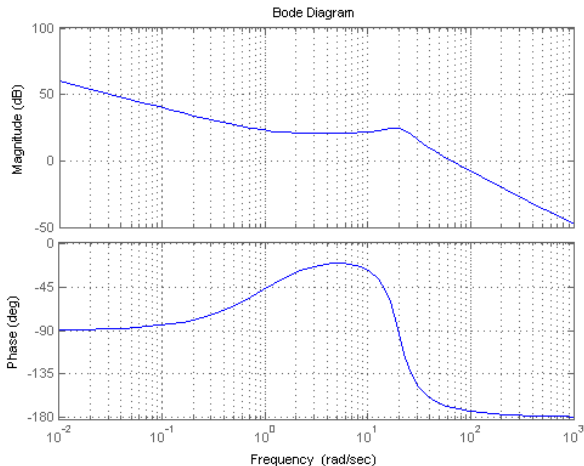
Minimum phase systems

A stable system is said to be a **minimum phase** system if **it has not time delays or right-half plane (RHP) zeros**.

For minimum phase systems there is a unique relationship between the gain and phase of the frequency response $G(j\omega)$. This may be quantified by the *Bode's gain-phase relationship*

$$\angle G(j\bar{\omega}) = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{d \ln |G(j\omega)|}{d \ln \omega} \ln \left| \frac{\omega + \bar{\omega}}{\omega - \bar{\omega}} \right| \frac{d\omega}{\omega} \, .$$

The name minimum phase refers to the fact that such a system has the minimum possible phase lag for the given magnitude response $|G(j\omega)|$.

Goal Oriented Training in Theory

GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Associazione
EURATOM-ENEA
sulla Fusione

Preliminaries   Graphical representations of $G(j\omega)$

Nyquist plot

The Nyquist is a polar plot of the frequency response $G(j\omega)$ on the complex plane.

This plot combines the two Bode plots - magnitude and phase - on a single graph, with frequency ω, which ranges in $(-\infty, +\infty)$, as a parameter along the curve.

Nyquist plots are useful to check stability of closed-loop systems (see ▸ Nyquist stability criterion).

Matlab commands

nyquist(sys) plots the Nyquist plot of the LTI system **sys**.

Nyquist plot - Example

Goal Oriented Training in Theory

GOTiT

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Associazione
EURATOM–ENEA
sulla Fusione

CREATE

Preliminaries   Graphical representations of $G(j\omega)$
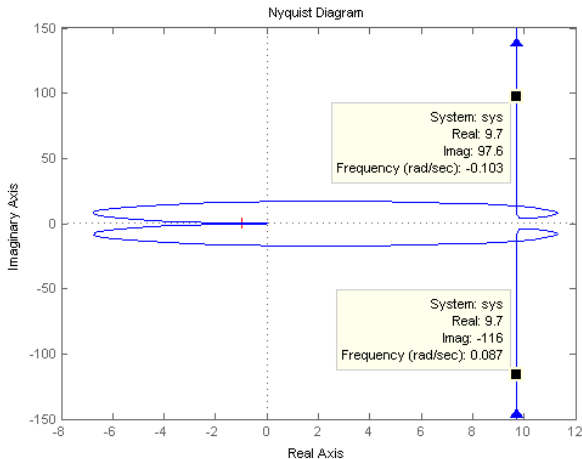
Nichols plot

It is similar to the Nyquist plot, since it plots both the magnitude and the phase of $G(j\omega)$ on a single chart, with frequency ω as a parameter along the curve.

As for the Bode plot the magnitude $|G(j\omega)|$ is expressed in dB and the phase $\angle G(j\omega)$ in degree.

Nichols charts are useful for the design of control systems, in particular for the design of lead, lag, lead-lag compensators.

Matlab commands

nichols(sys) plots the Nichols chart of the LTI system **sys**.

Nichols plot - Example
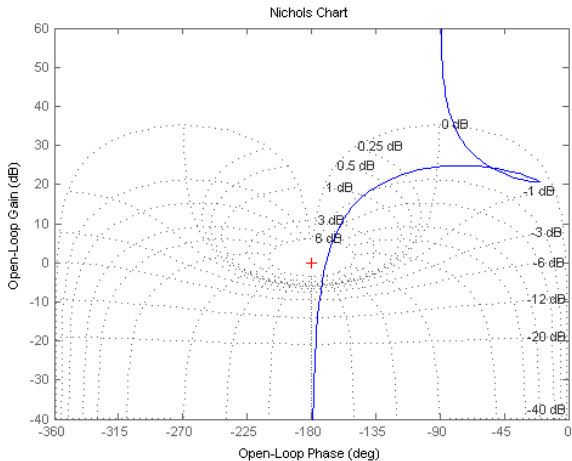
The control problem
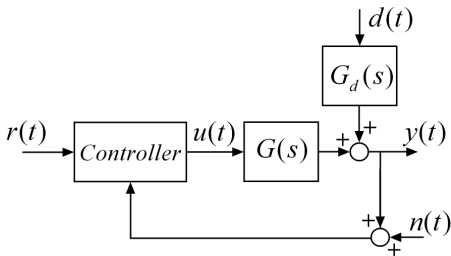
The objective of a control system is to make the output of a plant $y(t)$ behave in a desired way by manipulating the plant input $u(t)$.

A good controller should manipulate $u(t)$ so as to

- counteract the effect of a disturbance $d(t)$ (**regulator problem**)

- keep the output close to a given reference input $r(t)$ (**servo problem**)

In both cases we want the control error $e(t) = y(t) - r(t)$ to be small.

Why feedback ?
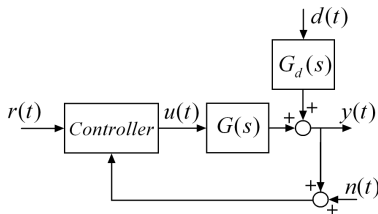
A major sources of difficulty are that

1. the plant model $G(s)$ and the disturbance model $G_d(s)$ may be affected by uncertainty and/or may change with time
2. the disturbance is not always measurable
3. the plant can be unstable

It turns out that e feed-forward approach is not robust enough and/or is not always a viable solution.

A feedback approach can guarantee the desired degree of robustness. However design a feedback control system is not straightforward: **instability is around the corner!**

Performance and stability

A good controller must guarantee:

- **Nominal stability** - The system is stable with no model uncertainty
- **Nominal Performance** - The system satisfies the performance specifications with no model uncertainty
- **Robust stability** The system is stable for all perturbed plants about the nominal model up to the worst case model uncertainty
- **Robust performance** The system satisfies the performance specifications for all perturbed plants about the nominal model up to the worst case model uncertainty

One degree-of-freedom controller
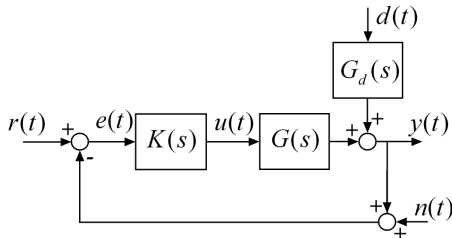
The input to the plant is given by

$$U(s) = K(s)\big(R(s) - Y(s) - N(s)\big).$$

The objective of control is to manipulate design a *controller* $K(s)$ such that the control error $e(t) = r(t) - y(t)$ remains small in spite of disturbances $d(t)$.

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Feedback Control Systems   The control problem

Terminology and Notation
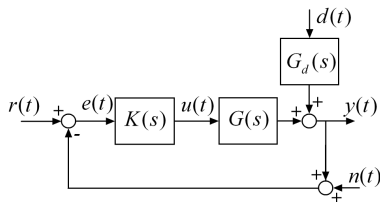
- $L(s) = G(s)K(s)$ is called **loop transfer function**
- $S(s) = (I + L(s))^{-1}$ is called **sensitivity function**
- $T(s) = (I + L(s))^{-1} L(s)$ is called **complementary sensitivity function**

It is straightforward to note that

$$T(s) + S(s) = I.$$

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Feedback Control Systems   The control problem

One DOF controller

Exploiting the composition rules for block diagrams, it turns out that

$$Y(s) = T \cdot R(s) + SG_d \cdot D(s) - T \cdot N(s) \tag{6a}$$

$$E(s) = -S \cdot R(s) + SG_d \cdot D(s) - T \cdot N(s) \tag{6b}$$

$$U(s) = KS \cdot R(s) - K(s)S(s)G_d \cdot D(s) - KS \cdot N(s) \tag{6c}$$

One DOF controller

Remark - $S(s)$ is called sensitivity because it gives the relative sensitivity of the closed-loop transfer function $T(s)$ to the relative plant model error. In particular, **at a given frequency** ω we have for a SISO plant that

$$\frac{\frac{dT}{T}}{\frac{dG}{G}} = S.$$

Remark - Equations (6) are written in matrix form because they apply to MIMO systems. For SISO systems we may write

$$S(s) = \frac{1}{1 + L(s)},$$

$$T(s) = \frac{L(s)}{1 + L(s)}.$$

Goal Oriented Training in Theory

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

Feedback Control Systems   The control problem

The control dilemma

Let consider

$$Y(s) = T \cdot R(s) + SG_d \cdot D(s) - T \cdot N(s)\,.$$

- In order to reduce the effect of the disturbance $d(t)$ on the output $y(t)$, the sensitivity function $S(s)$ should be made small (particularly in the *low frequency* range)
- In order to reduce the effect of the measurement noise $n(t)$ on the output $y(t)$, the complementary sensitivity function $T(s)$ should be made small (particularly in the *high frequency* range)

However, for all frequencies it is

$$T + S = I\,.$$

Thus a trade-off solution must be achieved.

Feedback may cause instability

One of the main issues in designing feedback controllers is stability.

If the feedback gain is too large then the controller may *overreact* and the closed-loop system becomes unstable.

▸ Download Simulink example
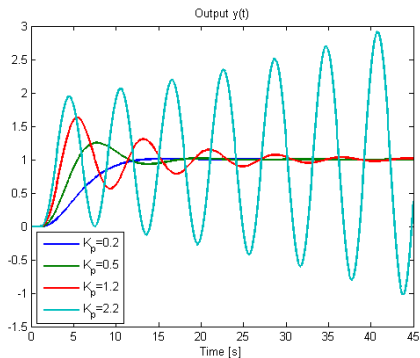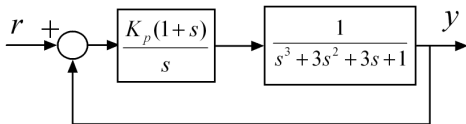
Bandwidth and crossover frequency

Usually the frequency response of loop transfer function $|L(j\omega)|$ has a low-pass behaviour.

The *crossover frequency* ω_c is the frequency such that $|L(j\omega_c)| = 1$.

In most of the cases the crossover frequency is a good estimation of the closed-loop bandwidth
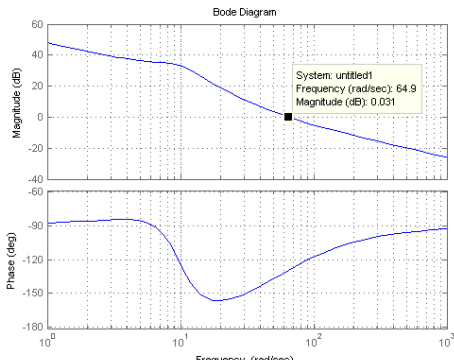
Gain and phase margin

The frequency response of the loop transfer function $L(j\omega)$ can be used to estimate the **stability margins**.
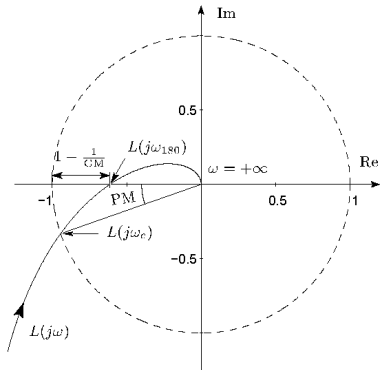
Gain margin (GM)

$$1/|L(j\omega_{180})|,$$

where ω_{180} is the *phase corssover frequency*.

Phase margin (PM)

$$\angle L(j\omega_c) + 180^\circ,$$

where ω_c is the *crossover frequency*.

Goal Oriented Training in Theory
GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Feedback Control Systems   Stability margins

Gain Margin

The GM is the factor by which the loop gain $|L(j\omega)|$ may be increased before the closed-loop system becomes unstable.

The GM is thus a direct safeguard against steady-state gain uncertainty.

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Feedback Control Systems   Stability margins

Phase Margin

The phase margin tells how much phase lag can added to $L(s)$ at frequency ω_c before the phase at this frequency becomes $180°$ which corresponds to closed-loop instability (see ▸ Nyquist stability criterion).

The PM is a direct safeguard against time delay uncertainty.

Nyquist Criterion - Preliminaries

The Nyquist Criterion permits to check the stability of a closed loop system by using the Nyquist plot of the loop frequency response $L(j\omega)$.

The criterion is based on the fact the the *close-loop poles are equal to the zeros* of the transfer function

$$D(s) = 1 + L(s).$$

Hence, if $D(s)$ has at least one zero $\bar{z}$ such that $\Re(\bar{z}) > 0$ the closed-loop system is unstable.

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Goal Oriented Training in Theory

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Feedback Control Systems   Nyquist Criterion

Nyquist Criterion

Consider a loop frequency response $L(j\omega)$ and let

- P be the number of poles of $L(s)$ with strictly positive real part
- Z be the number of zeros of $L(s)$ with strictly positive real part

The Nyquist plot of $L(j\omega)$ makes a number of encirclements N (**clockwise**) about the point $(-1, j0)$ equal to

$$N = Z - P.$$

It turns out that the closed-loop system is asymptotically **if and only if** the Nyquist plot of $L(j\omega)$ encircle (**counter clockwise**) the point $(-1, j0)$ a number of times equal to P.

The criterion is valid if the Nyquist plot of $L(j\omega)$ do not intersect the point $(-1, j0)$.

Nyquist plot - Example

$$G(s) = \frac{1}{s^3 + 3s^2 + 3s + 1} \,,$$

$$K(s) = K_p \frac{1 + s}{s} \,.$$

▸ Download Simulink example

Nyquist Criterion - Remarks

1. If the loop transfer function $L(s)$ has a zero pole of multiplicity l, then the Nyquist plot has a discontinuity at $\omega = 0$. Further analysis indicates that the zero poles should be neglected, hence if there are no other unstable poles, then the loop transfer function $L(s)$ should be considered stable, i.e. $P = 0$.

Nyquist Criterion - Remarks

1. If the loop transfer function $L(s)$ has a zero pole of multiplicity l, then the Nyquist plot has a discontinuity at $\omega = 0$. Further analysis indicates that the zero poles should be neglected, hence if there are no other unstable poles, then the loop transfer function $L(s)$ should be considered stable, i.e. $P = 0$.

2. If the loop transfer function $L(s)$ is stable, then the closed-loop system is unstable for any encirclement (clockwise) of the point -1.

Goal Oriented Training in Theory

Associazione
EURATOM–ENEA
sulla Fusione

GOTiT

Feedback Control Systems   Nyquist Criterion

Nyquist Criterion - Remarks

1. If the loop transfer function $L(s)$ has a zero pole of multiplicity l, then the Nyquist plot has a discontinuity at $\omega = 0$. Further analysis indicates that the zero poles should be neglected, hence if there are no other unstable poles, then the loop transfer function $L(s)$ should be considered stable, i.e. $P = 0$.

2. If the loop transfer function $L(s)$ is stable, then the closed-loop system is unstable for any encirclement (clockwise) of the point -1.

3. If the loop transfer function $L(s)$ is unstable, then there must be one counter clockwise encirclement of -1 for each pole of $L(s)$ in the right-half of the complex plane.

	Goal Oriented Training in Theory	*CREATE*	
EFDA			
EUROPEAN FUSION DEVELOPMENT AGREEMENT	**GOTiT**	Associazione EURATOM-ENEA sulla Fusione	

Feedback Control Systems   Nyquist Criterion

Nyquist Criterion - Remarks

1. If the loop transfer function $L(s)$ has a zero pole of multiplicity l, then the Nyquist plot has a discontinuity at $\omega = 0$. Further analysis indicates that the zero poles should be neglected, hence if there are no other unstable poles, then the loop transfer function $L(s)$ should be considered stable, i.e. $P = 0$.

2. If the loop transfer function $L(s)$ is stable, then the closed-loop system is unstable for any encirclement (clockwise) of the point -1.

3. If the loop transfer function $L(s)$ is unstable, then there must be one counter clockwise encirclement of -1 for each pole of $L(s)$ in the right-half of the complex plane.

4. If the Nyquist plot of $L(j\omega)$ intersect the point $(-1, j0)$, then deciding upon even the marginal stability of the system becomes difficult and the only conclusion that can be drawn from the graph is that there exist zeros on the imaginary axis.

Location of the poles of a closed-loop system

The time behaviour of a closed-loop system is strictly related to the position of its poles on the complex plane.

For example, for a second order closed-loop system it is possible to relate the features of the step response such as

- rise time
- overshoot
- settling time

to the location of its poles.

The Root Locus design method permits to evaluate how changes in the loop transfer function $L(s)$ affect the position of the closed-loop poles.

Goal Oriented Training in Theory

EUROPEAN FUSION DEVELOPMENT AGREEMENT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

GOTiT

Feedback Control Systems   Root locus

The Root Locus

The closed-loop poles are given by the roots of

$$1 + L(s). \tag{7}$$

Assuming that $L(s) = \rho L'(s)$ the Root Locus plot the locus of all possible roots of (7) as ρ varies in the range $[0, \infty)$.

The Root Locus can be used to study the effect of additional poles and zeros in $L'(s)$, i.e. in the controller $K(s)$.

The Root Locus can be effectively used to design SISO controllers.

Matlab commands

rlocus(sys) - plots the root locus for the loop transfer function specified by **sys**.

Root Locus - Example

$$L'(s) = \frac{1+s}{s} \cdot \frac{1}{s^3 + 3s^2 + 3s + 1} = \frac{1}{s(s+1)^2} \, .$$

Goal Oriented Training in Theory

GOTiT

CREATE

Associazione
EURATOM-ENEA
sulla Fusione

EUROPEAN FUSION DEVELOPMENT AGREEMENT

Feedback Control Systems   Root locus
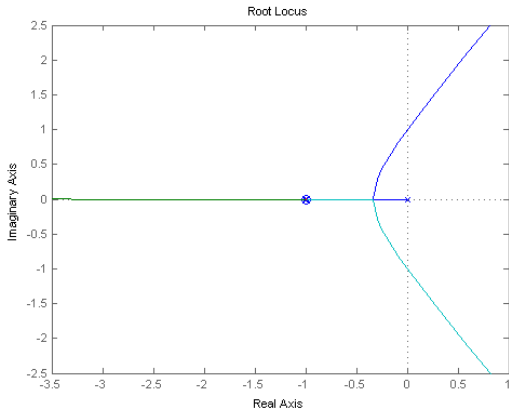
Root Locus - Example

Consider the unstable loop transfer function

$$L'(s) = \frac{1}{(s-2)^2}$$

It is not possible to stabilize the system with a simple proportional controller.
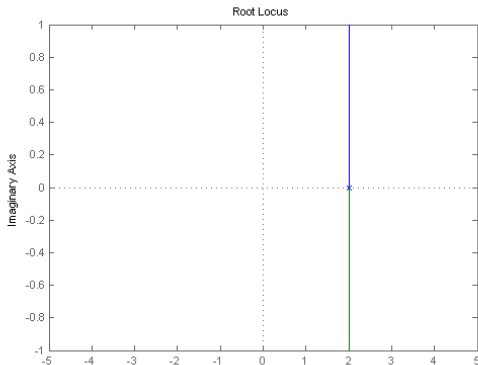
Root Locus

Root Locus - Example

It is not possible to stabilize the system with a simple proportional controller.

Root Locus - Example

Add a pole in 0 to have zero steady-state error

$$L'(s) = \frac{1}{s(s-2)^2}$$

It is still not possible to stabilize the system with a simple proportional controller.
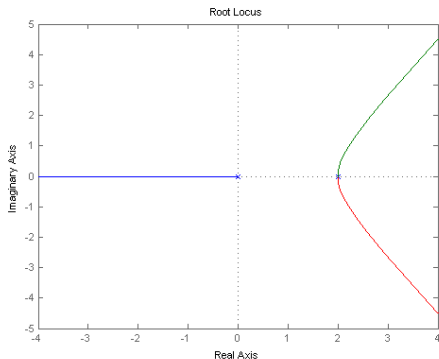
Root Locus - Example

Add two zeros to *draw* the poles in the LHP

$$L'(s) = \frac{(s + 10)^2}{s(s - 2)^2}$$

The controller $K(s) = \rho \frac{(s+10)^2}{s}$ can stabilize the plant but is **not causal**.
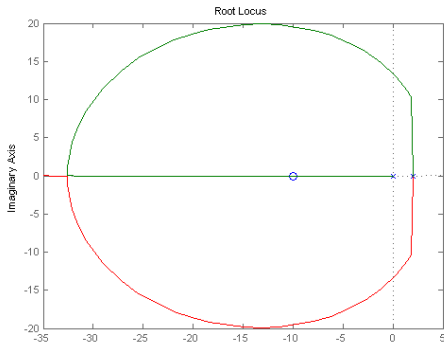
Root Locus - Example

Add an *high frequency* pole to have a proper controller

$$L'(s) = \frac{(s+10)^2}{s(s+100)(s-2)^2}$$

The controller $K(s) = \rho \frac{(s+10)^2}{s(s+100)}$ can stabilize.
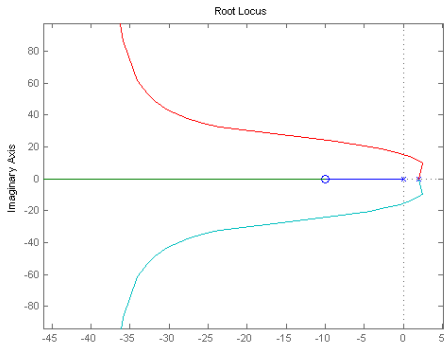
Hands on system

Let's try to design our own controller (with Matlab)!

Suggested textbooks

📕 F. M. Callier and C. A. Desoer
Linear System Theory
Springer-Verlag, 1991

📕 G. F. Franklin, J. D. Powell and A. Emami-Naeini
Feedback Control of Dynamic Systems
Pearson Prentice Hall, 2008

📕 S. Skogestad and I. Postlethwaite
Multivariable Feedback Control - Analysis and Design
John Wiley and Sons, 2006

📕 K. Zhou and J. C. Doyle
Essentials of Robust Control
Prentice Hall, 1998