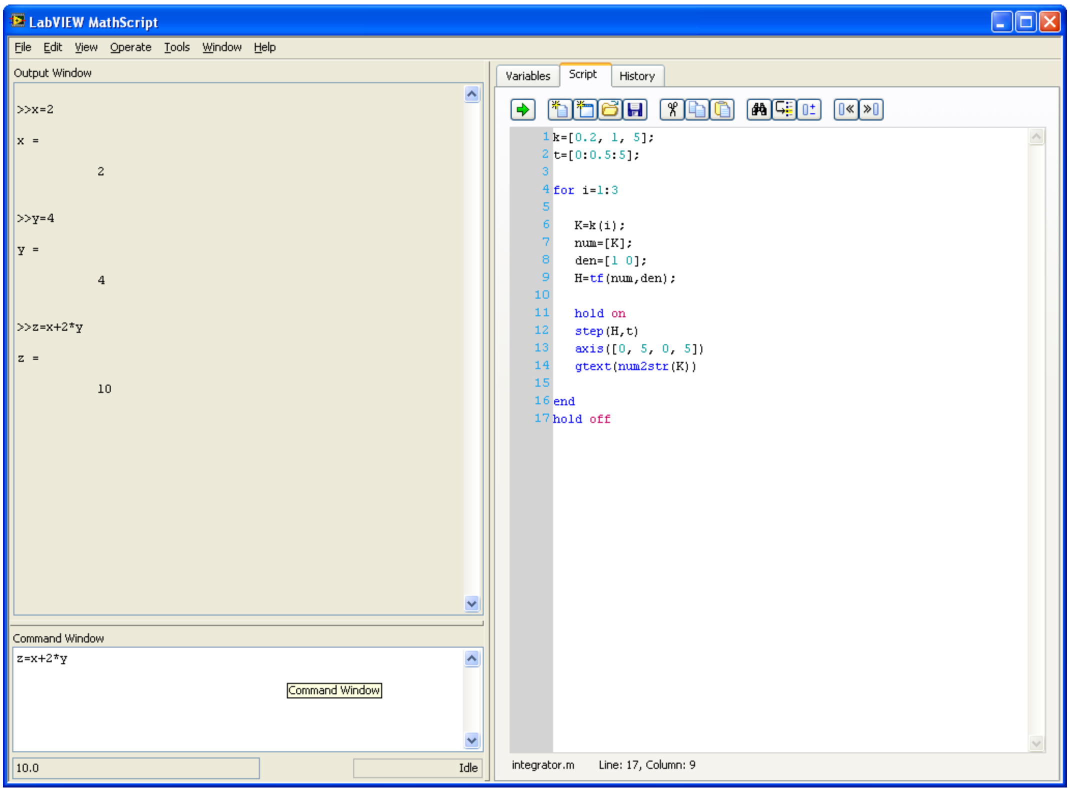
University College of Southeast Norway

Lecture Notes

Reguleringsteknikk vha. MathScript

Hans-Petter Halvorsen, 2016.10.26



http://home.hit.no/~hansha

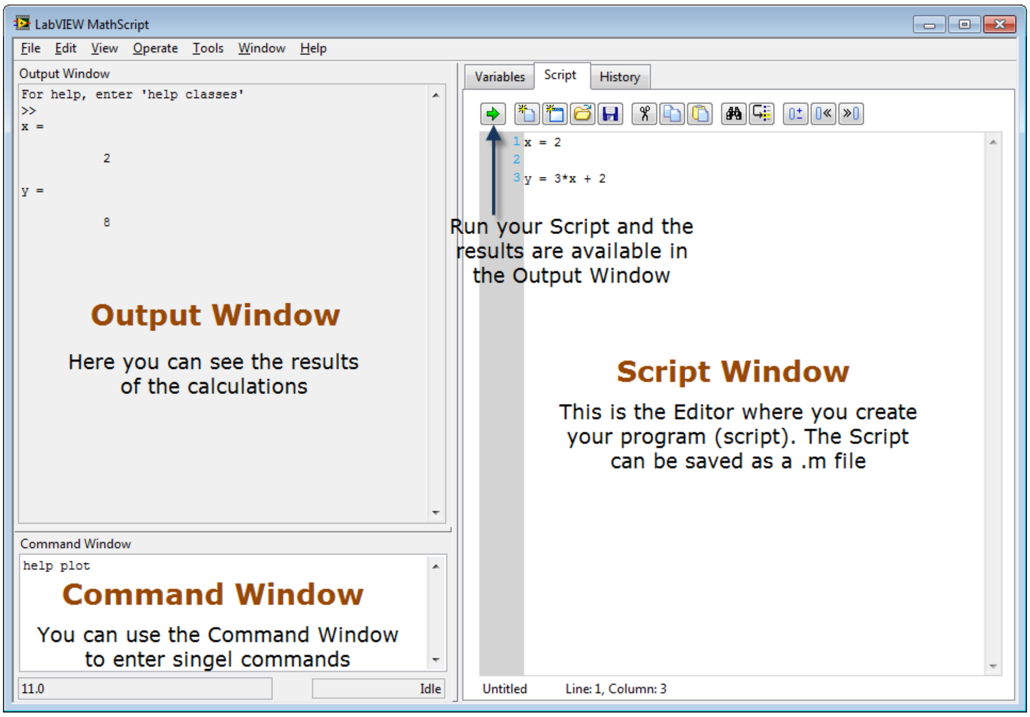
Innholdsfortegnelse

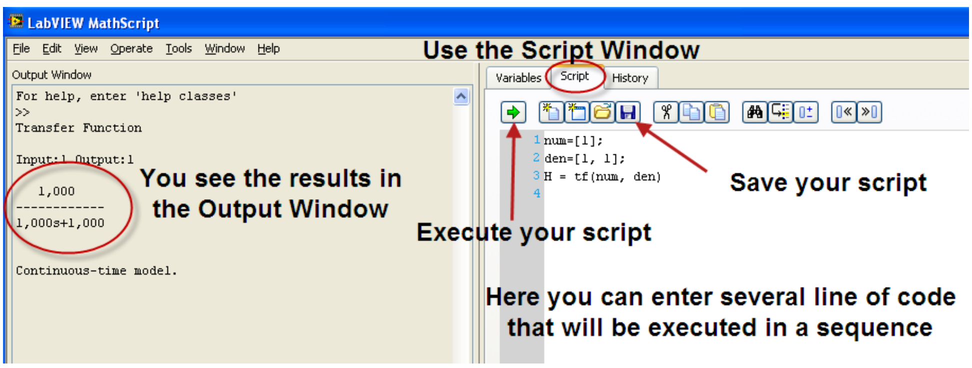
MathScript

Innledning

MathScript er en tilleggspakke til LabVIEW. Syntaksen er identisk som MATLAB. MathScript er velegnet ifm. design, analyse og simulering av reguleringssystemer da det finnes mange innebygde funksjoner for dette.

Vi åpner MathScript fra menyen i LabVIEW: Tools → MathScript Window





Exempel:

Gitt følgende funksjon:

$$y(x) = 3x + 2$$

Vi ønsker å finne $y(2)$

Først må vi definere x:

```
x = 2
```

Deretter kan vi definere funksjonen:

```
y = 3*x + 2
```

MathScript gir da følgende svar:

```
y =

      8
```

Merk! Maple kan utføre symbolsk matematikk, mens MathScript (og VB/C#) er et numerisk verktøy. Dette betyr at du alltid må definere verdier for variablene dine før du bruker dem i et matematisk uttrykk.

Merk! Syntaksen i MathScript er lik syntaksen i VB/C#, men MathScript er mye enklere på den måten at du ikke trenger å deklarere variablene og bestemme en datatype før du bruker dem. Dette blir automatisk håndtert at kompilatoren.

Plotting

MATLAB er veldig anvendelig til å plotte verdier, funksjoner, m.m.

Eksempel:

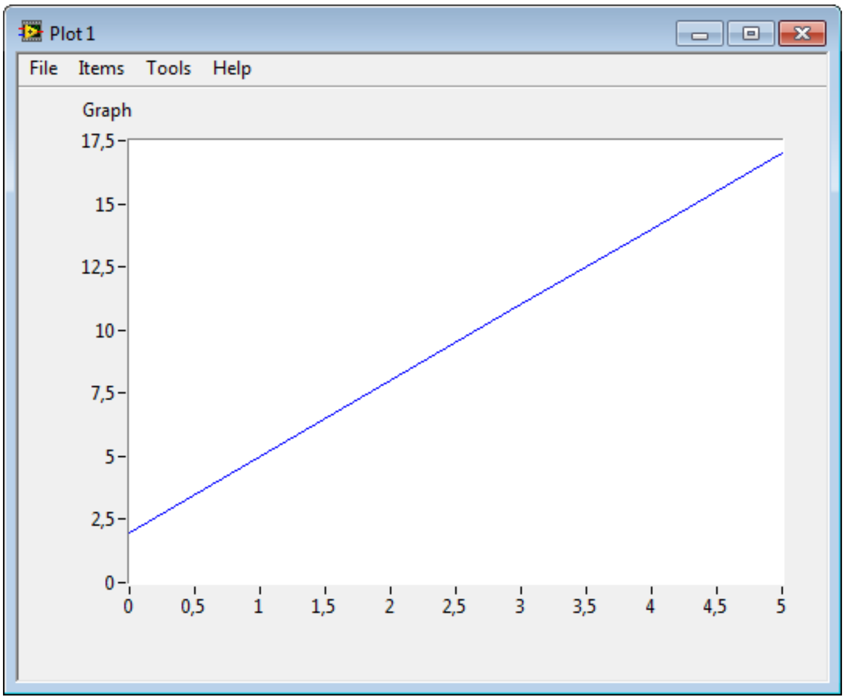
Vi ønsker å plotte følgende uttrykk vha. MathScript:

$$y(x) = 3x + 2$$

Vi bruker da **plot()** funksjonen i MathScript.

```
x = 0:5;
y = 3*x + 2;
plot(x,y)
```

I eksemplet over har vi plottet funksjonen i intervallet $[0, 5]$. Resultatet blir da som følger:



Følgene funksjoner blir brukt mye ifm plotting:.

Function	Description	Example
plot	Generates a plot. plot(y) plots the columns of **y** against the indexes of the columns.	`>X = [0:0.01:1];` `>Y = X.*X;` `>plot(X, Y)`
figure	Create a new figure window	`>>figure` `>>figure(1)`
subplot	Create subplots in a Figure. subplot(m,n,p) or subplot(mnp), breaks the Figure window into an m-by-n matrix of small axes, selects the p-th axes for the current plot. The axes are counted along the top row of the Figure window, then the second row, etc.	`>>subplot(2,2,1)`
grid	Creates grid lines in a plot. “grid on” adds major grid lines to the current plot. “grid off” removes major and minor grid lines from the current plot.	`>>grid` `>>grid on` `>>grid off`
axis	Control axis scaling and appearance. “axis([xmin xmax ymin ymax])” sets the limits for the x- and y-axis of the current axes.	`>>axis([xmin xmax ymin ymax])` `>>axis off` `>>axis on`
title	Add title to current plot title('string')	`>>title('this is a title')`
xlabel	Add xlabel to current plot xlabel('string')	`>> xlabel('time')`
ylabel	Add ylabel to current plot ylabel('string')	`>> ylabel('temperature')`
legend	Creates a legend in the corner (or at a specified position) of the plot	`>> legend('temperature')`
hold	Freezes the current plot, so that additional plots can be overlaid	`>>hold on` `>>hold off`

Tips & Triks

→ Bruk **Script** window – ikke Command window. Du kan bruke Command window til enkle ting, for eksempel til help <function>. I Script window kan du lagre koden i såkalte .m filer.

I Command window kan du bare skrive en og en kommando av gangen, mens i Script window kan du skrive flere kommandoer adskilt med linjeskift.

→ **Kommentarer** - %

```
% Dette er en kommentar
x=2;
y=3*x
```

Bruk kommentarer aktivt for å øke lesbarheten i programmet ditt!

→ **Desimaltegn**: Bruk punktum – ikke komma! Dvs. y=3.2 – ikke y=3,2

→ Ikke bruk mellomrom (space) i filnavn eller navn på funksjoner!

→ Bruk piltaster (Pil opp og Pil Ned) for å bla i tidligere brukte kommandoer i Command Window

→ En grei regel: En oppgave – en fil, dvs. <u>ikke</u> putt alle oppgavene i en fil!!

→ Funksjoner:

- **Kun en funksjon i hver fil!**
- **Filnavnet (.m) og navnet på funksjonen må være det samme!**

→ Bruk semikolon ";" etter kommandoer/funksjoner hvis du ikke trenger å vise svaret på skjermen

```
a=2;
b=4;
y=a+b
```

→ Bruk engelske navn på variable, funksjoner, filer, m.m. Dette er vanlig praksis i programmering!

Bruk alltid variable - ikke sett inn tall direkte i uttrykkene

```
a=2;
b=4;
y=a+b
```

Ikke:

```
y=2+4
```

→ **Bruk hjelp** for å finne ut mer om de funksjonene du skal bruke

For å få hjelp om tf funksjonen skriver du følgende i Command window:

```
help tf
```

→I toppen av Scriptet ditt, legg alltid til følgende kommandoer:

```
clear
clc
close all
…
```

Dette sørger for at du ikke får problemer med gamle variable, m.m.

→ Greske bokstaver: I matematikken og i reguleringsteknikk er det vanlige å bruke greske

Bokstaver i formler, m.m. Disse kan ikke brukes direkte i MathScript, så finn på gode variabelnavn for disse. Eksempler:

ω_0 – w0

ζ – zeta eller eventuelt bare z

osv.

→ Matematiske uttrykk: Bruk følgende i MathScript:

x^2	`x^2`
$\sqrt{x}$	`sqrt(x)`
$ln(x)$	`log(x)`
$\log(x)$	`Log10(x)`

e^x	**exp**(x)
π	**pi**

Eksempel:

Anta følgende matematiske funksjon:

$$z = 3x^2 + \sqrt{x^2 + y^2} + e^{\ln(x)}$$

Hvor z er en funksjon av x og y, dvs. $z(x, y)$.

Vi ønsker å bruke MathScript til å finne $z(2,2)$.

MathScript koden blir som følger:

```
x = 2;
y = 2;
z = 3*x^2 + sqrt(x^2 + y^2)+ exp(log(x))
```

MathScript gir følgende svar:

```
z = 16.8284
```

Merk! Vi må alltid definere verdier for variablene (x og y) før vi bruker dem i et uttrykk.

Reguleringsteknikk

Reguleringsteknikk beskriver hvordan man kan regulere og stabilisere et dynamisk system.

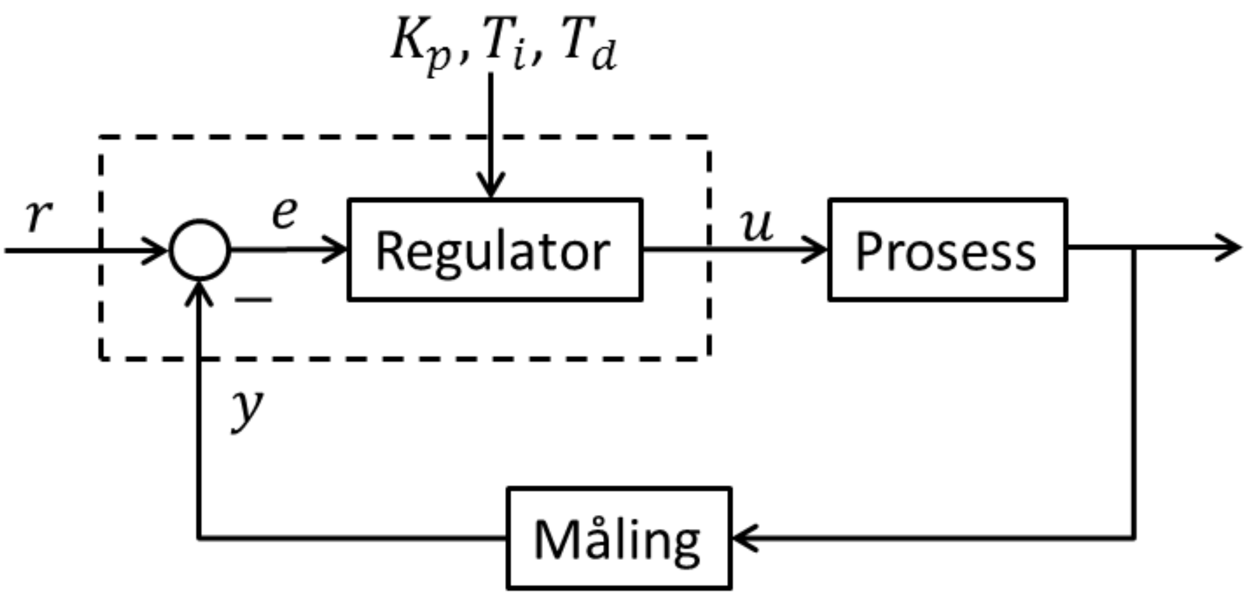
Et dynamisk system kan reguleres eller styres på flere måter. Et system som involverer en person som styrer en maskin, f.eks. en person som kjører bil, kalles manuell styring.

Regulering av vannivå i en tank, regulering av temperatur, eller regulering av gjennomstrømning, regulering av trykk, fart, osv.

Det er ubegrenset hva du kan regulere, og alle disse kan bli regulert på samme måte; med en **regulator**.

Regulator

Oppgaven til en regulator er å endre pådraget i forhold til måleverdien fra prosessen, det gjør den med noen parameterer som kalles P, I og D (PID regulator). Disse variablene varierer fra prosess til prosess. Formålet til disse variablene er å skape den optimale reguleringsalgoritmen til en viss prosess, for å få en mest mulig stabil reguleringen, dvs at utgangen y følger referansen r best mulig.



PID Algoritme

En PID regulator er gitt ved:

$$u(t) = K_p e + \frac{K_p}{T_i}\int_0^t e d\tau + K_p T_d \dot{e}$$

Der e er avviket mellom referansen r og utgangen y ($e = r - y$), mens u er pådraget.

P-ledd (Proporsjonal):

$$u_P(t) = K_p e$$

Der K_p er (proporsjonal) forsterkningen

I-ledd (Integral):

$$u_I(t) = \frac{K_p}{T_i}\int_0^t e d\tau$$

Der T_i er integraltiden

→ I-leddet sørger for at regulatoren gir null avvik (stasjonært) (Statisk ytelse)

D-ledd (Deriverte):

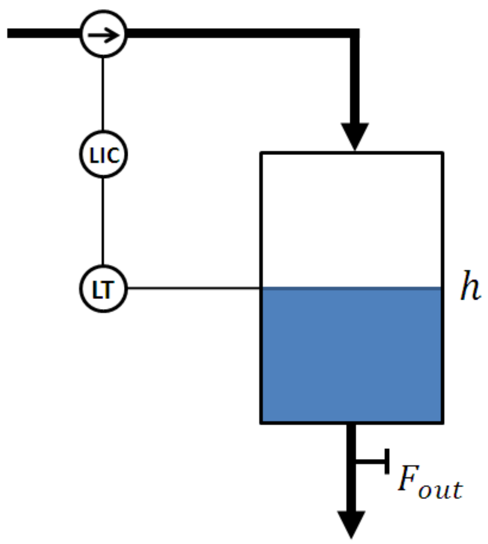
$$u_D(t) = K_p T_d \dot{e}$$

Der T_d er derivattiden

→ D-leddet sørger for at regulatoren reagerer raskt (Dynamisk ytelse)

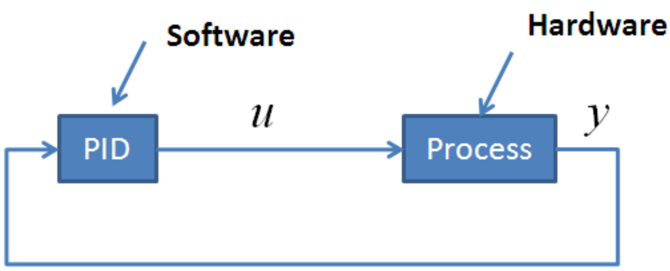
Eksempel

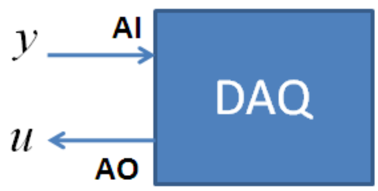
Nedenfor ser vi reguleringssystemet for en vanntank. Formålet er å regulere nivået i tanken (h) på et gitt nivå (referanseverdien). Dette gjøres i dette tilfellet ved en PID regulator som styrer en pumpe på tankens innløp.



I/O (DAQ) enhet

Vanligvis blir regulatoren implementert i software, dvs. vha. en datamaskin eller liknende, mens prosessen er en fysisk enhet (f.eks. en vanntank eller likende). Vi må da sende signaler mellom regulatoren og prosessen. Får å få til dette må vi bruke en DAQ (I/O) enhet. En del av DAQ enhets oppgave vil da være å konvertere mellom analoge og digitale signaler.





Et eksempel på en DAQ enhet kan være NI USB-6008:

Denne har 8 analoge innganger og 2 analoge utganger, i tillegg til 12 analoge inn/utganger.

Denne blir brukt i mange fag ved Høgskolen i Telemark.

Transferfunksjoner

Transferfunksjoner er nyttige å bruk ved analyse og design av dynamiske systemer.

Transferfunksjoner er modeller basert på Laplace-transformasjonen. Transferfunksjonen gir altså en modellbeskrivelse i s-planet (der s er Laplace-operatoren).

En transferfunksjon kan skrives på følgende generelle form:

$$H(S) = \frac{y(s)}{u(s)}$$

Hvor $y(s)$ er utgangen mens $u(s)$ er inngangen.

Transferfunksjonen er altså forholdet mellom inngangen og utgangen når alle andre inngangsvariable, samt alle initialbetingelser er satt lik null.

Merk! Transferfunksjoner gjelder bare for lineære systemer.

En transferfunksjon kan skrives på følgende generelle polynomform:

$$\boxed{H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}}$$

Der telleren til transferfunksjonen beskriver nullpunktene til systemet, mens nevneren beskriver polene til systemet.

MathScript

I MathScript definerer vi transferfunksjoner vha. den innebygde **tf** funksjonen:

```
num=[bm, bm_1, bm_2, … , b1, b0];
den=[an, an_1, an_2, … , a1, a0];
H = tf(num, den)
```

Eksempel:

1. Gitt følgende transferfunksjon:

$$H(s) = \frac{2s^2 + 3s + 4}{5s + 9}$$

MathScript kode:

```
num=[2, 3, 4];
den=[5, 9];
H = tf(num, den)
```

2. Gitt følgende transferfunksjon:

$$H(s) = \frac{4s^4 + 3s + 4}{5s^2 + 9}$$

Merk! Hvis noen ledd mangler, setter vi disse lik 0. Transferfunksjonen kan bli omskrevet som:

$$H(s) = \frac{4s^4 + 0 \cdot s^3 + 0 \cdot s^2 + 3s + 4}{5s^2 + 0 \cdot s + 9}$$

MathScript kode:

```
num=[4, 0, 0, 3, 4];
den=[5, 0, 9];
H = tf(num, den)
```

3. Gitt følgende transferfunksjon:

$$H(s) = \frac{7 + 3s + 2s^2}{5s + 6s^2}$$

Merk! Hvis noen ledd mangler, setter vi disse lik 0, samt passer på at de kommer i riktig rekkefølge. Transferfunksjonen kan bli omskrevet som:

$$H(s) = \frac{2s^2 + 3s + 7}{6s^2 + 5s}$$

MathScript kode:

```
num=[2, 3, 7];
den=[6, 5, 0];
H = tf(num, den)
```

[Slutt på eksempel]

Differensiallikninger

Vi kan si at transferfunksjonen gir en ekstern inn-ut representasjon av et system. Differensiallikningene derimot gir en intern representasjon av systemet fordi differensiallikningene beskriver systemets indre liv.

Vi kan finne transferfunksjonen fra differensiallikningen(e) vha. Laplace transformasjon.

Vi har følgende Transformasjonspar:

Derivasjon:

$$\boxed{\dot{x} \Leftrightarrow sx(s)}$$

Integrasjon:

$$\boxed{\int x \Leftrightarrow \frac{1}{s}x(s)}$$

Tidsforsinkelse:

$$\boxed{u(t-\tau) \Leftrightarrow u(s)e^{-\tau s}}$$

Eksempel:

Gitt følgende differensiallikning:

$$\dot{x} = -0.5x + 2u$$

Vi ønsker å finne transferfunksjonen:

$$H(s) = \frac{x(s)}{u(s)}$$

Laplace gir:

$$sx(s) = -0.5x(s) + 2u(s)$$

Videre:

$$sx(s) + 0.5x(s) = 2u(s)$$

Videre:

$$x(s)(s + 0.5) = 2u(s)$$

Videre:

$$\frac{x(s)}{u(s)} = \frac{2}{s + 0.5} = \frac{4}{2s + 1}$$

Dette gir:

$$H(s) = \frac{x(s)}{u(s)} = \frac{4}{2s+1}$$

[Slutt på eksempel]

Eksempel:

Gitt følgende differensiallikning (med tidsforsinkelse):

$$\dot{x} = -\frac{1}{T}[x + Ku(t-\tau)]$$

Vi ønsker å finne transferfunksjonen:

$$H(s) = \frac{x(s)}{u(s)}$$

Laplace gir:

$$sx(s) = -\frac{1}{T}x(s) + \frac{K}{T}u(s)e^{-\tau s}$$

Videre:

$$Tsx(s) + x(s) = Ku(s)e^{-\tau s}$$

Videre:

$$x(s)(Ts+1) = Ku(s)e^{-\tau s}$$

Videre:

$$\frac{x(s)}{u(s)} = \frac{K}{Ts+1}e^{-\tau s}$$

Som gir:

$$H(s) = \frac{x(s)}{u(s)} = \frac{K}{Ts+1}e^{-\tau s}$$

[Slutt på eksempel]

Motsatt kan vi finne differensiallikningen(e) fra transferfunksjonen.

Eksempel:

Gitt følgende transferfunksjon:

$$H(s) = \frac{x(s)}{u(s)} = \frac{3}{0.5s + 1}$$

Vi ønsker å finne differensiallikningen.

Vi gjør følgende:

$$x(s)(0.5s + 1) = 3u(s)$$

Videre:

$$0.5sx(s) + x(s) = 3u(s)$$

Inverse Laplace:

$$0.5\dot{x} + x = 3u$$

Dette gir følgende differensiallikning:

$$\dot{x} = -2x + 6u$$

[Slutt på eksempel]

Sprangrespons

Vi ønsker ofte å finne et systems sprangrespons, dvs. systemet dynamiske oppførsel over tid etter et sprang i pådraget.

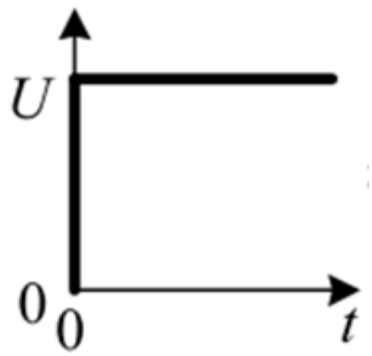
For en gitt transfer funksjon:

$$H(s) = \frac{y(s)}{u(s)}$$

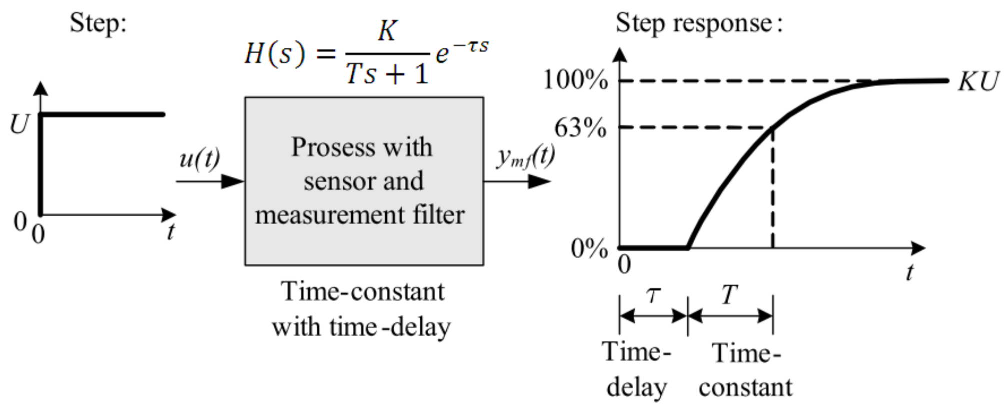
Får vi:

$$y(s) = H(s)u(s)$$

der vi vanligvis bruker et enhetssprang på inngangen, i dette tilfellet pådraget u, dvs pådraget u øker fra 0 til 1 ved $t = 0$.



Dvs. vi ønsker å plotte y(t) etter et sprang på inngangen. Nedenfor ser vi et typisk eksempel:



[Figure: F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]

For et enhetsprang har vi følgende Laplacetransformasjon:

$$\boxed{\frac{1}{s} \Leftrightarrow 1}$$

For et sprang med størrelse k (amplitude k) har vi følgende Laplacetransformasjon:

$$\frac{k}{s} \Leftrightarrow k$$

Vi finner dermed sprangresponsen for et gitt system på følgende måte:

Gitt Transfer funksjonen:

$$H(s) = \frac{y(s)}{u(s)}$$

1. Vi ordner det slik at utgangen står på høyre side:

$$y(s) = H(s)u(s)$$

2. Deretter finner vi inngangsvariabelens transformasjonspar og setter denne inn i uttrykket.

For et sprang med størrelse k (amplitude k) får vi da følgende:

$$u(s) = \frac{k}{s}$$

Dvs.:

$$y(s) = H(s) \cdot \frac{k}{s}$$

3. Tilslutt tidsresponsen ved å ta inverse laplace av uttrykket over.

Eksempel:

Gitt følgende system:

$$H = \frac{y(s)}{u(s)} = \frac{3}{2s+1}$$

Vi får da:

$$y(s) = H(s)u(s)$$

Hvor u er et enhetssprang:

$$u(s) = \frac{1}{s}$$

Dvs.:

$$y(s) = \frac{3}{2s+1} \cdot \frac{1}{s} = \frac{3}{(2s+1)s}$$

Vi broker følgende Laplace-transformasjonspar (som vi finner i en tabell):

$$\boxed{\frac{K}{(Ts+1)s} \Leftrightarrow K(1 - e^{-t/T})}$$

Vi får da:

$$\underline{y(t) = 3(1 - e^{-t/2})}$$

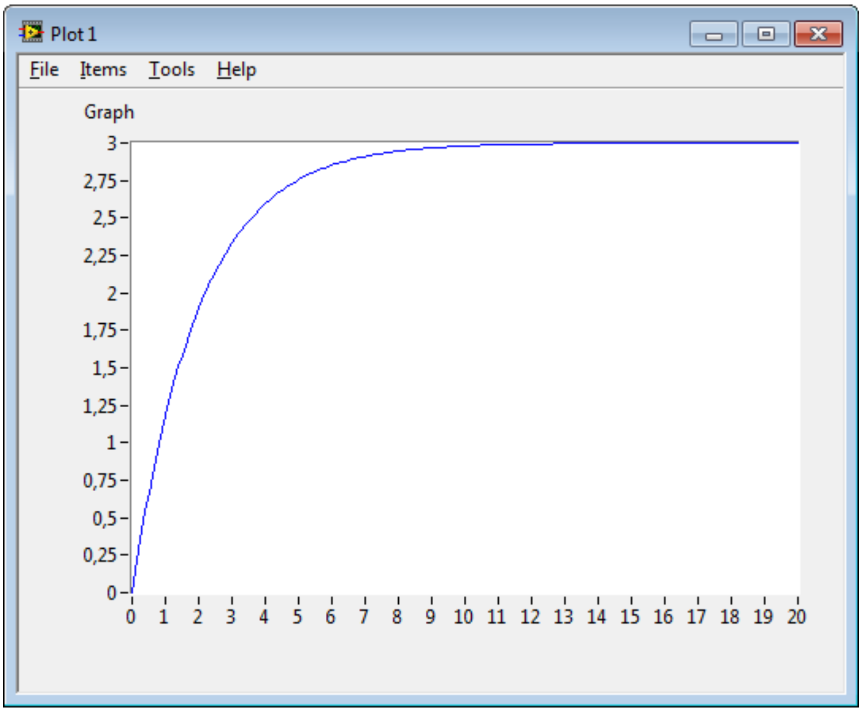
MathScript:

Vi kan da plotte denne sprangresponsen vha. MathScript:

```
t=0:0.1:20;

y = 3*(1-exp(-t/2));
```

```
plot(t,y)
```

Som gir følgende sprangrespons:



Vi ser også at den stasjonære verdien i dette tilfellet er $y_s = 3$. Den stasjonære verdien er den verdien vi får når $t \to \infty$.

Dette er jo litt tungvint siden vi må gjøre en del manuelle beregninger, men vha. MathScript kan vi gjøre det mye enklere:

```
num=[3];
den=[2, 1];
H = tf(num, den);

step(H, t)
```

Dvs. vi kan definere transfer funksjonen (vha. **tf** funksjonen) direkte i MathScript og bruke den innebygde **step** funksjonen til å beregne og eller plotte sprangresponsen. Resultatet blir det samme.

[Slutt på eksempel]

Sluttverditeoremet

I mange tilfeller er det den stasjonære responstiden man er interessert i, det vil si tidsresponsen når $t \to \infty$.

Da kan vi bruke sluttverditeoremet:

$$\boxed{y_s = \lim_{t\to\infty} y(t) = \lim_{s\to 0} s \cdot y(s)}$$

Eksempel:

Gitt følgende system:

$$H = \frac{y(s)}{u(s)} = \frac{3}{2s+1}$$

Vi får:

$$y_s = \lim_{t\to\infty} y(t) = \lim_{s\to 0} s \cdot y(s) = \lim_{s\to 0} s \frac{3}{(2s+1)s} = \frac{3}{2\cdot 0+1} = \underline{3}$$

Som du ser får vi samme resultat som vi fikk i forrige eksempel.

[Slutt på eksempel]

1.ordens systemer

1.ordens transferfunksjon:

$$\boxed{H(s) = \frac{K}{Ts + 1}}$$

K er forsterkning

T er tidskonstant

Sprangrespons

For et 1.ordens system har vi:

$$H = \frac{y(s)}{u(s)} = \frac{K}{Ts + 1}$$

Vi får da:

$$y(s) = H(s)u(s)$$

Hvor:

$$u(s) = \frac{U}{s}$$

Dvs.:

$$y(s) = \frac{K}{Ts + 1} \cdot \frac{U}{s}$$

Vi broker følgende Laplace-transformasjonspar (som vi finner i en tabell):

$$\boxed{\frac{K}{(Ts + 1)s} \Leftrightarrow K(1 - e^{-t/T})}$$

Tilslutt får vi da følgende:

$$\underline{y(t) = KU(1 - e^{-t/T})}$$

MathScript

Eksempel:

$$H(s) = \frac{K}{Ts + 1}$$

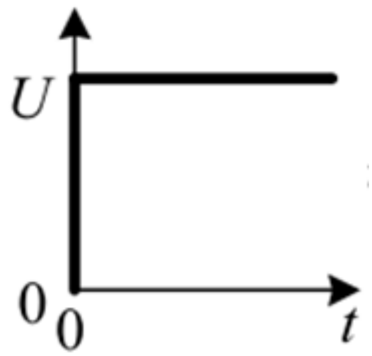
$K = 1, T = 2$

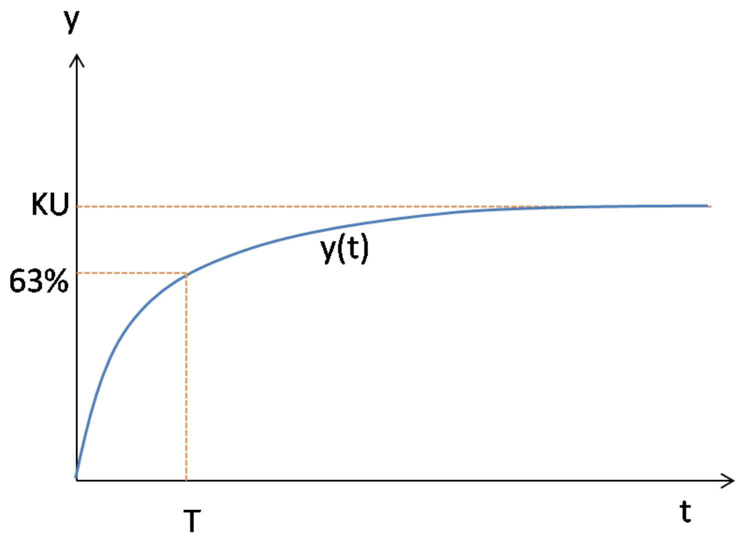
MathScript:

```
K = 1;
T = 2;
num = [K];
den = [T, 1];
H = tf(num, den)
```

Sprangrespons:

$y(s) = H(s)u(s)$





Vanligvis: er $U = 1$, dvs en enhetssprang

Mathscript:

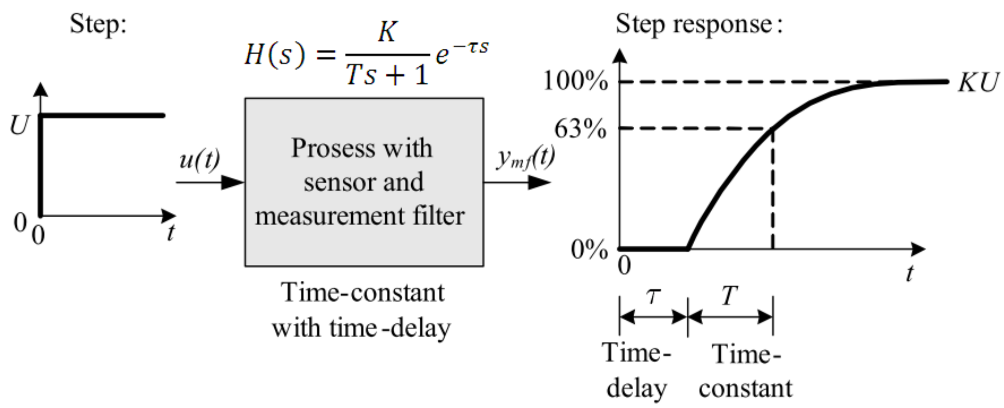
```
step(H)
```

1.ordens system med tidsforsinkelse

En 1.ordens transferfunksjon med tidsforsinkelse:

$$H(s) = \frac{K}{Ts+1}e^{-\tau s}$$

En sprangrespons for en slik transferfunksjon har følgende karakteristikk:



[Figure: F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]

Fra sprangresponsen kan vi enkelt finne K, T and τ.

MathScript

Eksempel:

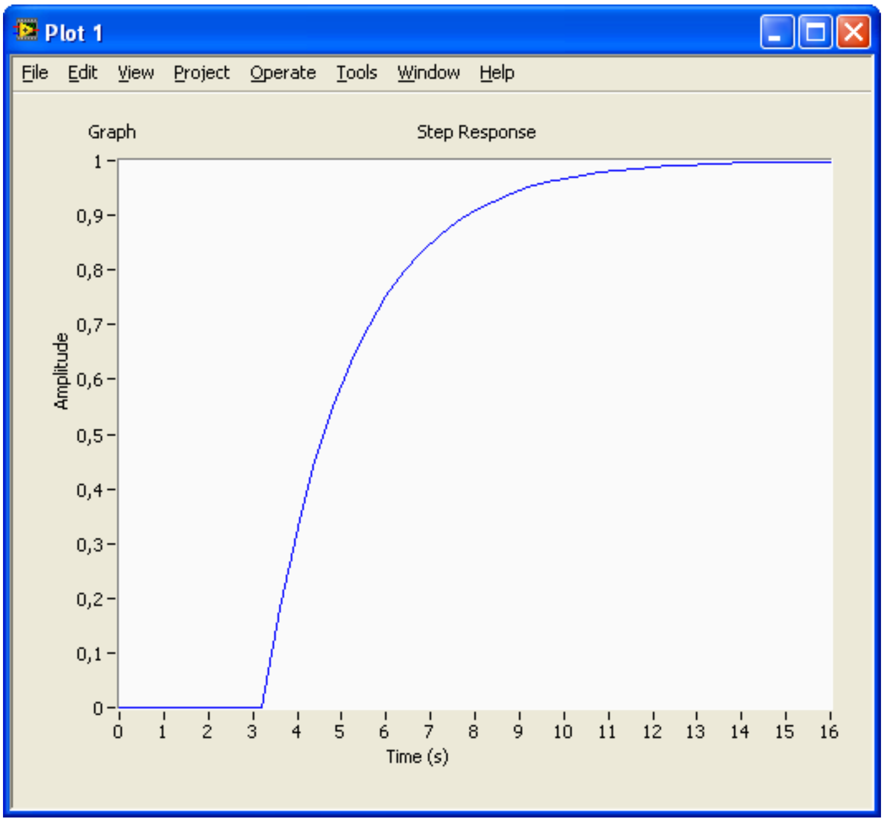
$$H(s) = \frac{K}{Ts+1}e^{-\tau s}$$

$K = 1, T = 2, \tau = 3$

MathScript:

```
K = 1;
T = 2;
delay=3;
H = sys_order1(K, T, delay)
step(H)
```

Dette gir følgende plot:



2.ordens systemer

2.ordens transferfunksjon:

$$\boxed{H(s) = \frac{K}{as^2 + bs + c} = \frac{K\omega_0^2}{s^2 + 2\zeta\omega_0 + \omega_0^2} = \frac{K}{\left(\frac{s}{\omega_0}\right)^2 + 2\zeta\frac{s}{\omega_0} + 1}}$$

K er forsterkning

ζ (zeta) er relativ dempingsfaktor

ω_0[rad/s] er udempet resonansfrekvens

Response Time

The response time for a 2.order system is approximately:

$$\boxed{T_r \approx \frac{1.5}{\omega_0}}$$

MathScript

Eksempel:

$$H(s) = \frac{1}{s^2 + s + 1}$$

Dvs: $K = 1, \omega_0 = 1, \zeta = 1$

MathScript kode:

```
K = 1;
w0 = 1;
zeta = 1;

a = 1/w0^2;
b = 2*zeta/w0;
c = 1;
```

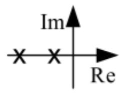
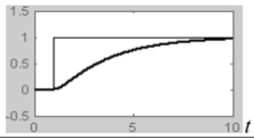
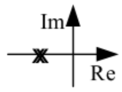
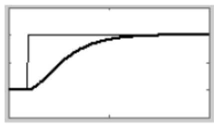
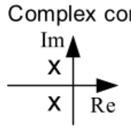
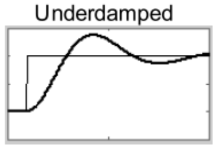
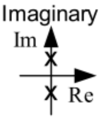
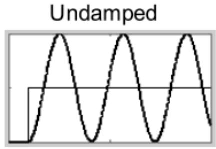
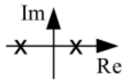
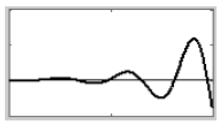
```
num = [K];
den = [a,b,c]
H = tf(num,den)
```

Sprangrespons:

```
step(H)
```

Sprangrespons og stabilitet

Verdien på ζ er avgjørende for systems sprangrespons og stabilitetsegenskaper:

Value of ζ	**Poles p_1 and p_2**	**Type of step response $y(t)$**
$\zeta > 1$	Real and distinct 	Overdamped 
$\zeta = 1$	Real and multiple 	Critically damped 
$0 < \zeta < 1$	Complex conj. 	Underdamped 
$\zeta = 0$	Imaginary 	Undamped 
$\zeta < 0$	Pos. real part 	Unstable 

[Figure: F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]

Poler

Systemets stabilitetsegenskaper er definert av polens plassering i det komplekse plan.

Gitt følgende generelle 2.ordens system:

$$H(s) = \frac{K}{as^2 + bs + c}$$

Polene finner vi ved å sette nevneren lik null:

$$as^2 + bs + c = 0$$

Dvs. vi har en standard 2.ordens likning:

$$ax^2 + bx + c = 0$$

Løsningen på denne er som kjent:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Spesialtilfelle

Når polene er reelle og forskjellige ($\zeta > 0$) har vi følgende viktige spesialtilfelle:

$$H(s) = \frac{Kp_1p_2}{(s - p_1)(s - p_2)} = \frac{K}{(T_1s + 1)(T_2s + 1)}$$

Der p_1 og p_2 er systemets poler og T_1 og T_2 er systemets tidskonstanter.

Dette kan sees på som to 1.ordens systemer i serie:

$$H(s) = H_1(s)H_1(s) = \frac{K}{(T_1s + 1)} \cdot \frac{1}{(T_2s + 1)} = \frac{K}{(T_1s + 1)(T_2s + 1)}$$

Blokkdiagrammer

Et reguleringssystem består av ulike blokker (transferfunksjoner), disse er enten koblet i serie eller i parallell. I reguleringssystemer er det også vanlig å bruke tilbakekoblinger i form av at regulatoren bruker målingen til å beregne pådragssignalet.

Serie

Seriekobling:

MathScript:

I MathScript kan vi bruke **series** funksjonen:

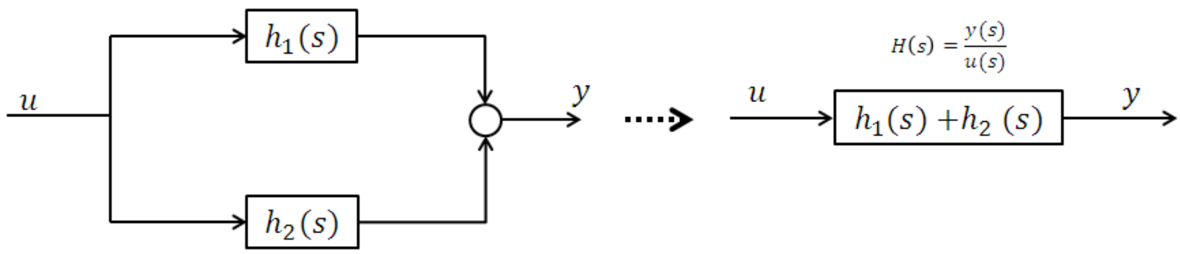
```
…
H = series(h1,h2)
```

Hvis vi har mer enn 2 kan vi gjøre følgende:

```
…
H = series(h1,series(h2,h3))
```

Parallell

Parallellkobling:



MathScript:

I MathScript kan vi bruke **parallel** funksjonen:

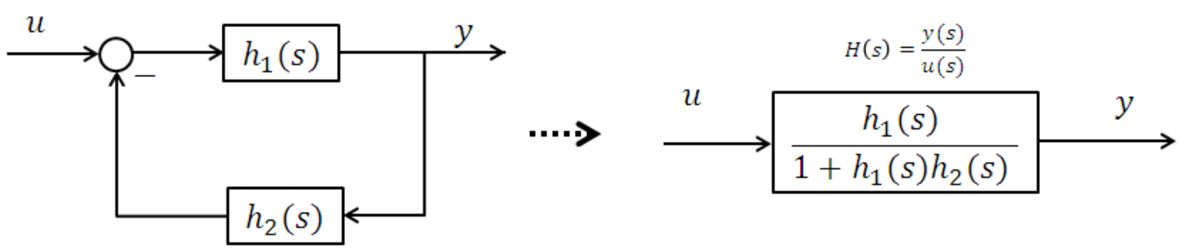
```
…
H = parallel(h1,h2)
```

Hvis vi har mer enn 2 kan vi gjøre følgende:

```
…
H = parallel(h1, parallel(h2,h3))
```

Tilbakekobling (Feedback)

Negativ tilbakekobling:



MathScript:

I MathScript kan vi bruke **feedback** funksjonen:

```
…
H = feedback(h1,h2)
```

Tilstandsrommodeller

En tilstandsrommodell er en strukturert måte å representere et sett med differensiallikninger på. Tilstandsmodell er veldig nyttige i forbindelse med reguleringsteori. Differensiallikningene blir satt på matrise-vektor form, som jo er basis-elementene i MathScript.

Anta følgende differensiallikninger:

$$\dot{x}_1 = a_{11}x_1 + a_{21}x_2 + \cdots + a_{n1}x_n + b_{11}u_1 + b_{21}u_2 + \cdots + b_{n1}u_n$$

$$\vdots$$

$$\dot{x}_n = a_{1m}x_1 + a_{2m}x_2 + \cdots + a_{nm}x_n + b_{1m}u_1 + b_{2m}u_2 + \cdots + b_{n1}u_n$$

$$\vdots$$

Disse kan skrives på følgende generelle form (tilstandsrommodell):

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} a_{11} & \cdots & a_{n1} \\ \vdots & \ddots & \vdots \\ a_{1m} & \cdots & a_{nm} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} b_{11} & \cdots & b_{n1} \\ \vdots & \ddots & \vdots \\ b_{1m} & \cdots & b_{nm} \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}}_{u}$$

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} c_{11} & \cdots & c_{n1} \\ \vdots & \ddots & \vdots \\ c_{1m} & \cdots & c_{nm} \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} d_{11} & \cdots & d_{n1} \\ \vdots & \ddots & \vdots \\ d_{1m} & \cdots & d_{nm} \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}}_{u}$$

Eller på følgende enkle form:

$$\boxed{\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}}$$

Der x – systemets interne tilstander, u – pådrag, y - måling

Eksempel:

Gitt følgende differensiallikninger:

$$\dot{x}_1 = x_1 + 2x_1$$

$$\dot{x}_1 = 3x_1 + 4x_1 + u$$

$$y = x_1$$

Dette gir:

$$\begin{bmatrix}\dot{x}_1 \\ \dot{x}_2\end{bmatrix} = \underbrace{\begin{bmatrix}1 & 2 \\ 3 & 4\end{bmatrix}}_{A}\begin{bmatrix}x_1 \\ x_2\end{bmatrix} + \underbrace{\begin{bmatrix}0 \\ 1\end{bmatrix}}_{B}u$$

$$y = \underbrace{\begin{bmatrix}1 & 0\end{bmatrix}}_{C}\begin{bmatrix}x_1 \\ x_2\end{bmatrix}$$

Merk! D matrisa er lik null i dette tilfellet, så dermed tar vi ikke med den.

MathScript

I MathScript gjør vi følgende:

```
A = [1 2; 3 4];
B = [0; 1];
C = [1, 0];
D = [0];
model = ss(A, B, C, D)
```

Sprangrespons:

```
step(model)
```

Tidsforsinkelse/Pade Appr.

Eksempler på dødtid/tidsforsinkelser i prosess -sammenheng:

- Transportbånd
- Rør, f.eks Luftrør

Transferfunskjon for tidsforsinkelse/dødtid:

$$\boxed{H(s) = \frac{y(s)}{u(s)} = e^{-\tau s}}$$

→ Transferfunksjonen $e^{-\tau s}$ er såkalt ikke-rasjonal, dvs. den har ikke teller og nevnerpolynom med potenser av s.

→ Ulike metoder for analyse og design av reguleringssystemer krever at systemet er en rasjonal transferfunksjon

Tidsplan:

$$y(t) = u(t - \tau)$$

Pade' Approksimasjon

Følgende approksimasjon brukes:

$$\boxed{e^{-\tau s} \approx \frac{1 - k_1 s + k_2 s^2 + \cdots \pm k_n s^n}{1 + k_1 s + k_2 s^2 + \cdots + k_n s^n}}$$

Hvor n er approksimasjonens orden.

$k_1, k_2, \ldots$ er konstanter

→ Jo høyere orden, jo bedre tilnærming til virkeligheten, men ulempen er at transferfunksjonen blir veldig kompleks! Så en gylden middelvei må velges

→ For 1.ordens og 2.ordens approksimasjoner er disse konstantene gitt ved:

$n = 1$	$n = 2$
$k_1 = \frac{\tau}{2}$, other $k_i = 0$	$k_1 = \frac{\tau}{2}$, $k_2 = \frac{\tau^2}{12}$, other $k_i = 0$

1.ordens Pade approksimasjon:

$$\boxed{e^{-\tau s} \approx \frac{1 - k_1 s}{1 + k_1 s}}$$

Hvor

$$k_1 = \frac{\tau}{2}$$

2.ordens Pade approksimasjon:

$$\boxed{e^{-\tau s} \approx \frac{1 - k_1 s + k_2 s^2}{1 + k_1 s + k_2 s^2}}$$

Hvor:

$$k_1 = \frac{\tau}{2}$$

$$k_2 = \frac{\tau^2}{12}$$

MathScript

Implementering av dødtid/Tidsforsinkelse.

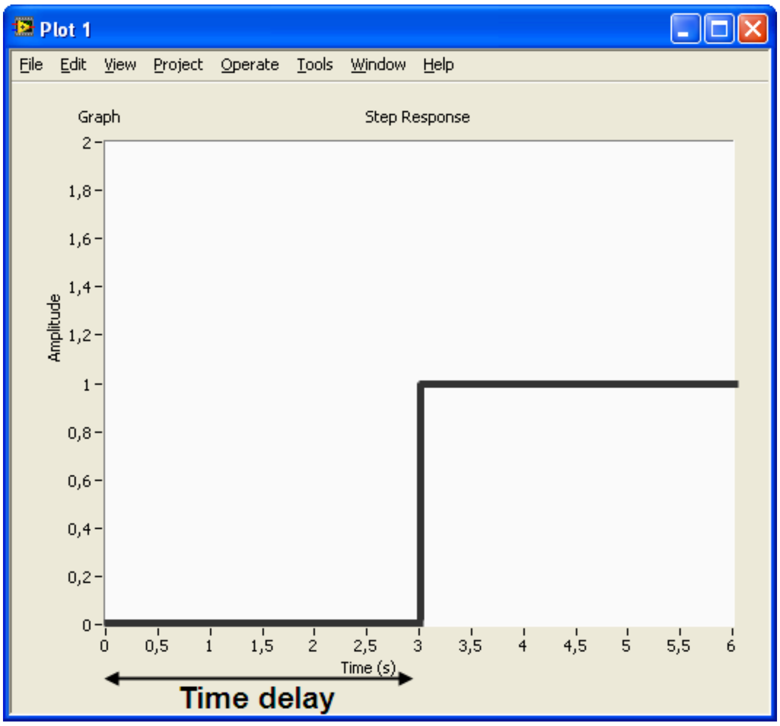
Eksempel:

$$H(s) = e^{-3s}$$

Dvs. tidsforsinkelsen er på 3 sekunder.

Sprangrespons:

I dette eksemplet viser vi ulike måter å implementere tidsforsinkelse på. For en tidsforsinkelse på 3 sekunder får vi følgende sprangrespons:



Metode 1

→ Vi bruker den innebygde funksjonen sys_order1

```
K = 1;
T = 0;
delay = 3
H = sys_order1(K,T,delay)
step(H)
```

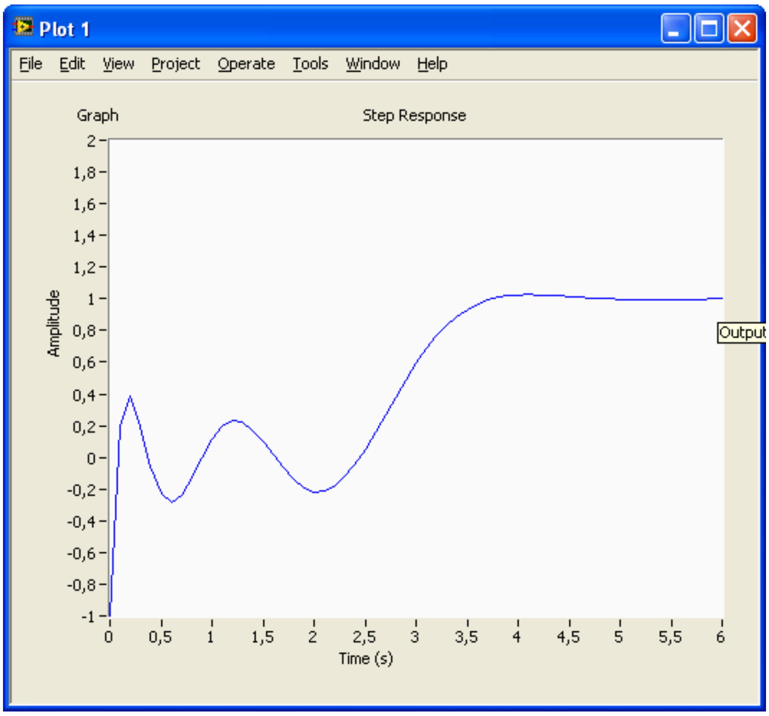
Metode 2

→ Vi bruker den innebygde pade funksjonen

Eksempel på en 2.ordens approksimasjon:

```
delay = 3;
order = 2;
H = pade(delay,order)
step(H)
```

Nedenfor ser vi sprangresponsen til en 5.ordens approksimasjon:



Metode 3

→ Vi bruker uttrykkene for Pade appr. og bruker tf funksjonen

Eksempel på en 1.ordens approksimasjon:

$$e^{-\tau s} \approx \frac{1 - k_1 s}{1 + k_1 s}$$

```
k1 = delay/2;
num = [-k1, 1];
den = [k1, 1];

H = tf(num,den)
step(H)
```

→ Det er slik den innebygde funksjonen **pade** er laget!

Merk! Pass på rekkefølgen på koeffisientene i telleren og nevneren (høyeste orden til venstre og deretter i synkende rekkefølge)!

$$e^{-\tau s} \approx \frac{1 - k_1 s}{1 + k_1 s} = \frac{-k_1 s + 1}{k_1 s + 1}$$

Stabilitetsanalyse

3 typer systemer (stabilitetsegenskaper):

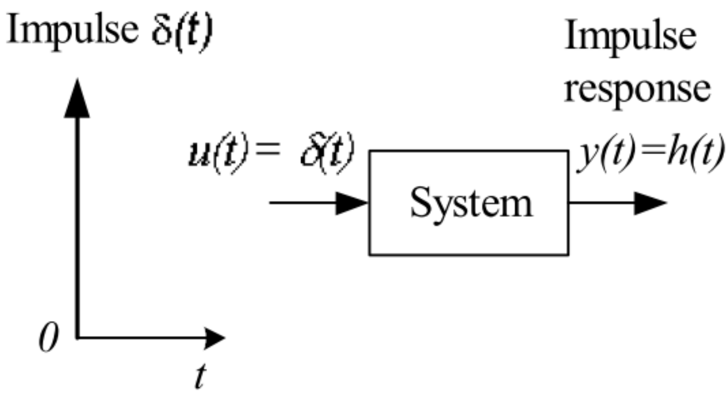
- **Asymptotisk stabilt system**
- **Marginalt stabilt system**
- **Ustabilt system**

Metoder for å finne stabilitetsegenskapene:

- **Impulsrespons**
- **Polplassering**

Impulsrespons

Vi kan definere et systems stabilitetsegenskaper utfra impulsresponsen [F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]:



Asymptotisk stabilt system

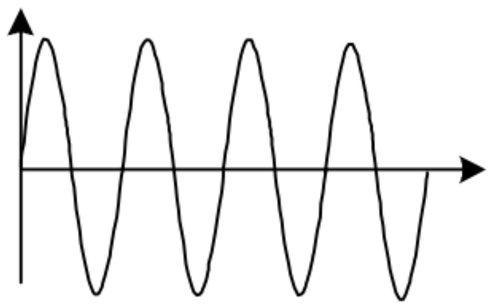
Impulsrespons:

Stasjonær Impulsrespons:

$$\lim_{t \to \infty} h(t) = 0$$

Marginalt stabilt system

Impulsrespons:

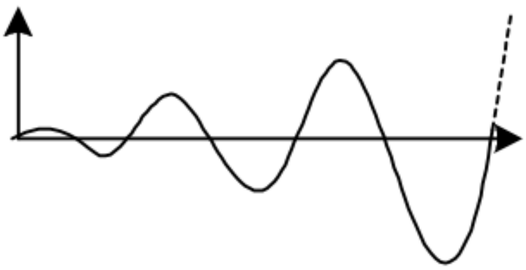


Stasjonær Impulsrespons:

$$0 < \lim_{t \to \infty} h(t) < \infty$$

Ustabilt system

Impulsrespons:



Stasjonær Impulsrespons:

$$\lim_{t \to \infty} h(t) = \infty$$

MathScript

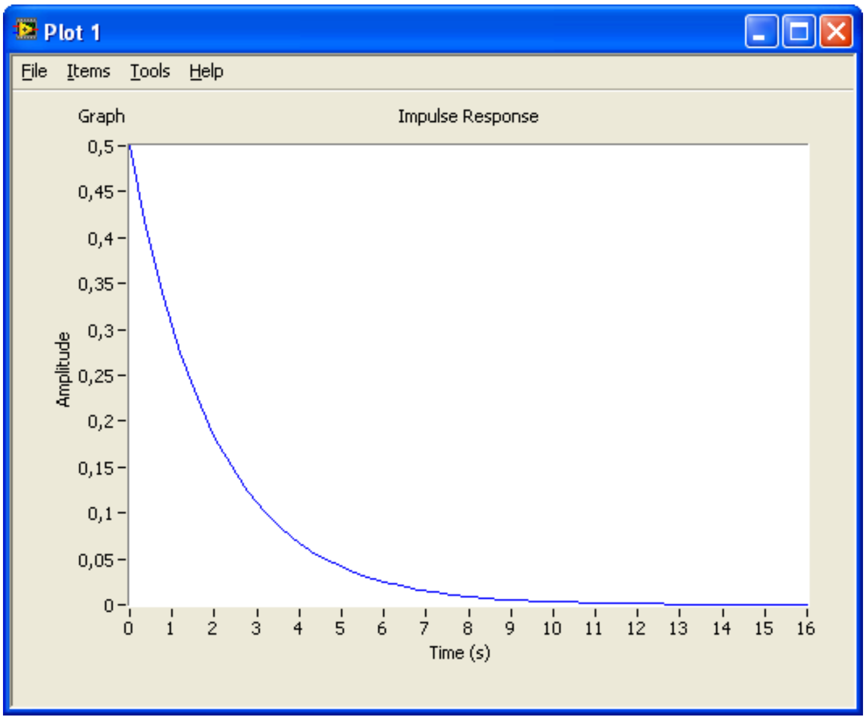
Eksempel:

$$H(s) = \frac{1}{2s + 1}$$

Impulsrespons:

```
K=1;
T=2;
num = [K];
den = [T, 1];
H = tf(num, den)

impulse(H)
```

Dette gir følgende impulsrespons:



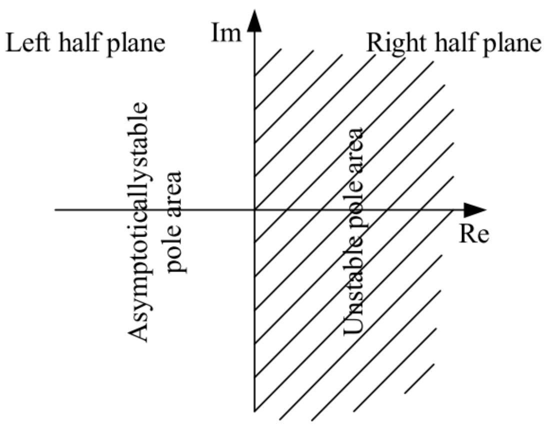
→ Asymptotisk stabilt system

Poler

Det kan være upraktisk å måtte jobbe med impulsresponser ved stabilitetsundersøkelser, så en bedre metode vil være å bruke systemets poler.

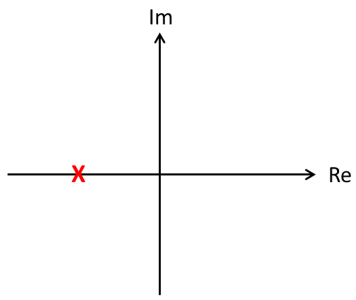
Polene til et gitt system finnes fra nevnerpolynomet i transferfunksjonen.

Det komplekse plan:



[Figure: F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]

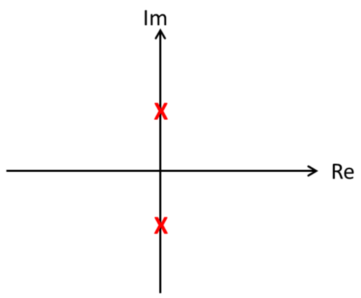
Asymptotisk stabilt system



Alle polene ligger i venstre halvplan (negativ realdel).

Ingen poler på den imaginære akse.

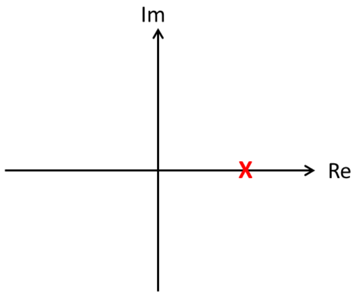
Marginalt stabilt system



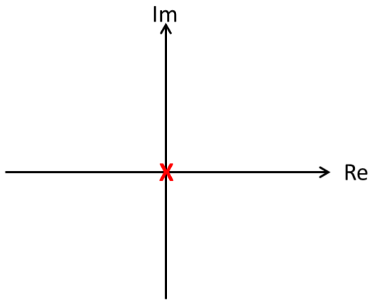
En eller flere poler ligger på den imaginære akse (har realdelen lik 0), og alle polene er forskjellige/ikke sammenfallende.

Dessuten, ingen poler i høyre halvplan

Ustabilt system



En eller flere poler ligger i høyre halvplan (har realdel større enn 0).



Eller: Det er multiple/sammenfallende poler på den imaginære akse.

F.eks: Dobbeltintegrator $H = \frac{1}{s^2}$

MathScript

Eksempel:

$$H(s) = \frac{1}{2s + 1}$$

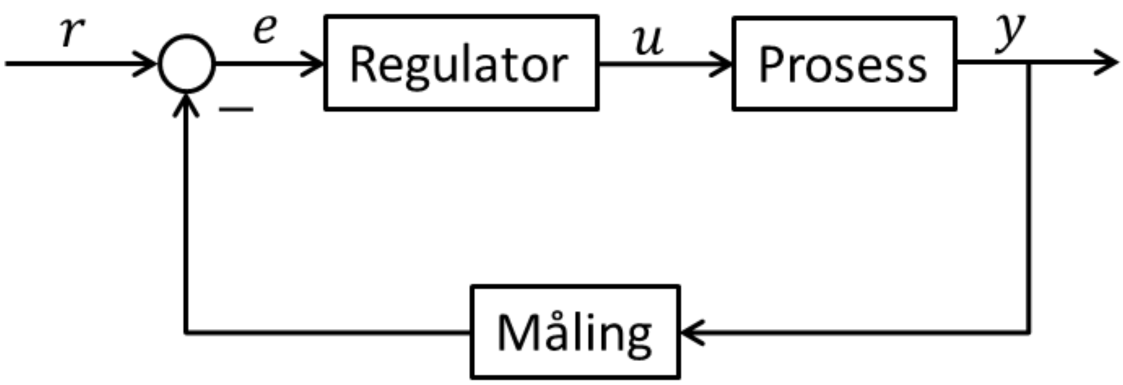
Poler:

```
K=1;
T=2;
num = [K];
den = [T, 1];
H = tf(num, den)
P = poles(H)

pzgraph(H)
```

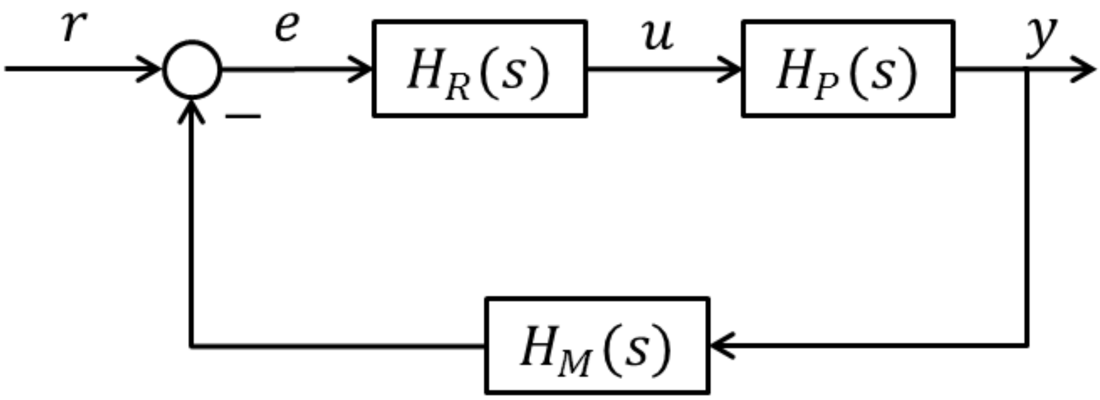
→ $p = -0.5$ → Asymptotisk stabilt system

Tilbakekoblede systemer

Skisse av et tilbakekoblet system:



Hver av disse kan beskrives vha. en transfer funksjon:



Sløyfetransferfunksjonen

(Engelsk: "Loop transfer function")

$$\boxed{L(s) = H_R H_P H_M}$$

Følgeforholdet

(Engelsk: "Tracking transfer function")

$$\boxed{T(s) = \frac{y(s)}{r(s)} = \frac{H_R H_P H_M}{1 + H_R H_P H_M} = \frac{L(s)}{1 + L(s)}}$$

Denne uttrykker hvor godt systemet følger referansen. Systemet har gode følgeegenskaper hvis $y \approx r$, dvs.:

$$|T| \approx 1$$

Sensitivitetsfunksjonen/Avviksforholdet

(Engelsk: "Sensitivity transfer function")

$$\boxed{S(s) = \frac{e(s)}{r(s)} = \frac{1}{1 + L(s)} = 1 - T(s)}$$

Denne uttrykker hvor "sensitivt" avviket er overfor referansen og denne bør derfor være "liten", dvs:

$$|S| \approx 0 \; or \; |S| \ll 1$$

Merk!

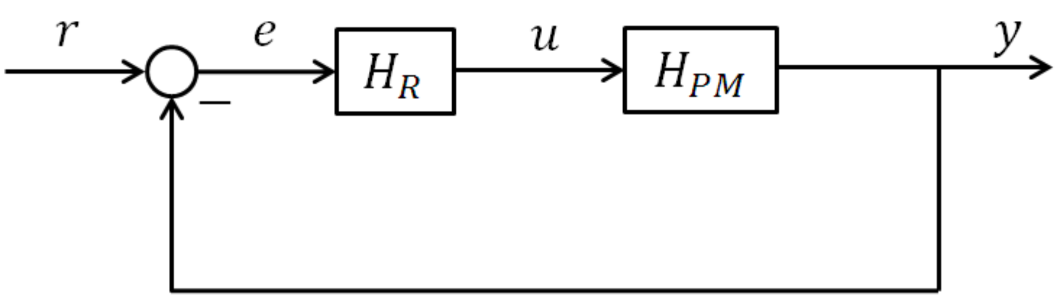
$$S(s) = 1 - T(s) \leftrightarrow T(s) = 1 - S(s)$$

og

$$T(s) + \; S(s) = \frac{L(s)}{1 + L(s)} + \frac{1}{1 + L(s)} \equiv 1$$

MathScript

Eksempel:



Sløyfetransferfunksjonen:

```
L = series(Hr, Hpm)
```

Hvis flere enn 2: M=series(H1, series(H2, H3))

Følgeforholdet:

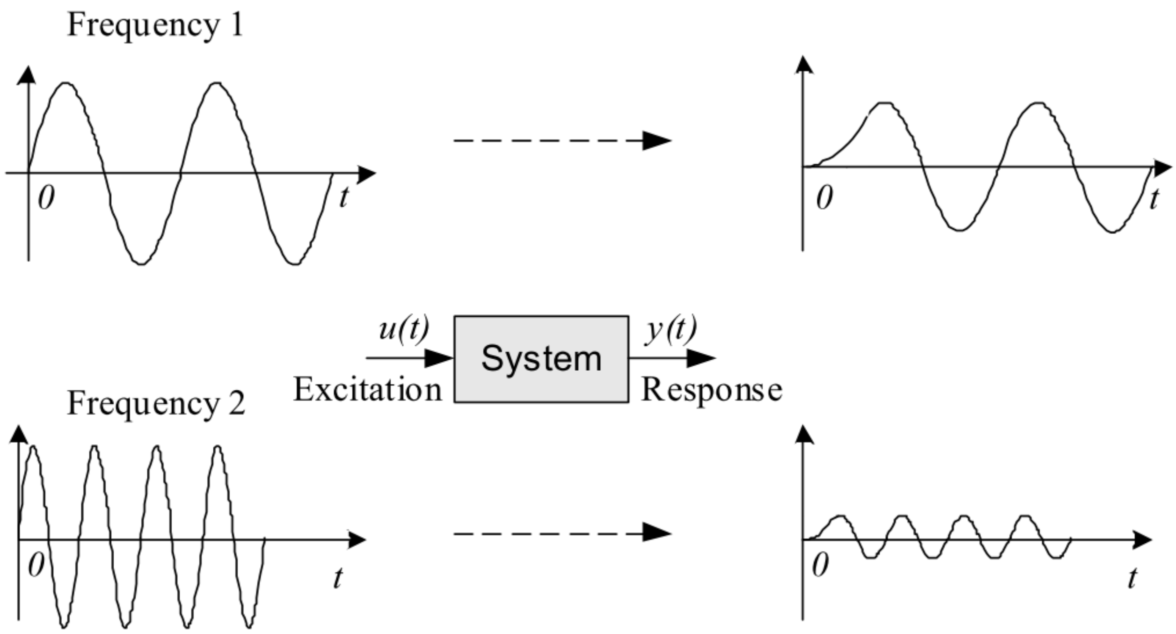
```
T = feedback(L, 1)
```

Sensitivitetsfunksjonen/Avviksforholdet

```
S = 1-T
```

Frekvensrespons

Et systems frekvensrespons uttrykker hvordan sinussignaler på systemets inngang endres gjennom systemet. Som vi ser av figuren under så vil både amplituden forandres, samt at signalet blir faseforskyvet.



[Figure: F. Haugen, Advanced Dynamics and Control: TechTeach, 2010]

Frekvensresponsen forteller hvilken amplitudeforsterkning og faseforskyvning den enkelte frekvenskomponent får gjennom systemet.

Frekvensresponsen til et system er definert som steady-state responsen til systemet hvor inngangssignalet er et sinus signal.

I steady-state vil utgangssignalet være forskjellig mtp amplitude/forsterkning (A) samt være faseforskyvet (ϕ).

Vi kan definer **inngangssignalet** som:

$$\boxed{u(t) = U\ sin\omega t}$$

Utgangssignalet i steady-state vil da være:

$$\boxed{y(t) = \underbrace{UA}_{Y}\ sin\ (\omega t + \phi)}$$

Hvor $A = \frac{Y}{U}$ er forholdet mellom utgangssignalets og inngangssignalets amplitude (i steady-state).

A og ϕ er funksjoner av frekvensen ω, så vi kan skrive $A = A(\omega), \phi = \phi(\omega)$

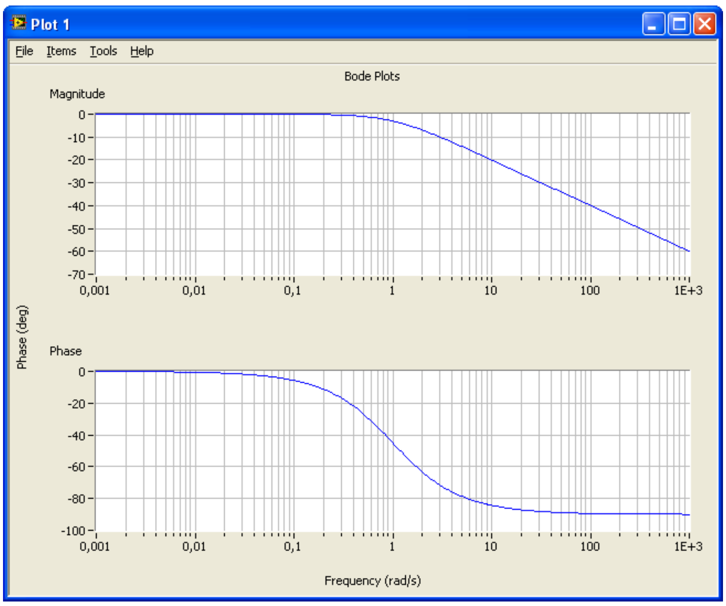
- $A(\omega)$ - Amplitudefunksjonen
- $\phi(\omega)$ – Fasefunksjonen

Bodediagram

Frekvensresponsen kan presenteres grafisk i et Bodediagram.

Bodediagrammet består av 2 deler:

- Et Amplitudediagram (Forsterkning $[dB]$ som funksjon av frekvens $[\omega]$)
- Et Fasediagram (Faseforskyvning $[grader]$ som funksjon av frekvens $[\omega]$)



Merk! Frekvensaksen er <u>logaritmisk</u> med 10-logaritmen for frekvensen som enhet

Vanligvis er enheten for frekvens Hertz $[Hz]$, mens i frekvensrespons brukes radianer $\omega\ [rad]$. Sammenhengen mellom disse er:

$$\boxed{\omega = 2\pi f}$$

MathScript

I MathScript kan vi enkelt tegne Bodediagrammer vha den innebygde funksjonen **bode**.

Eksempel:

Gitt transferfunksjon:

$$H(s) = \frac{y(s)}{u(s)} = \frac{1}{s+1}$$

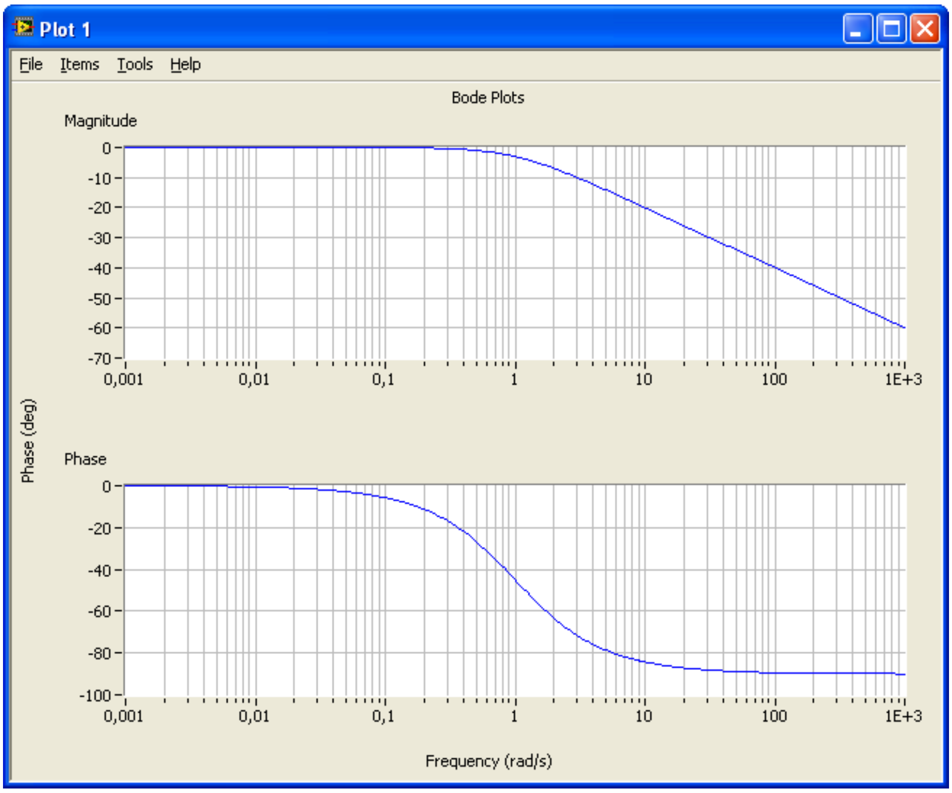
MathScript koden blir som følger:

```
K=1;
T=1;
num=[K];
den=[T, 1];
H = tf(num, den)
bode(H);
```

Det er ønskelig å ha grid på plottet:

```
subplot(2,1,1)
grid
subplot(2,1,2)
grid
```

Bode-diagrammet blir som følger:



[Slutt på eksempel]

Hvordan finne frekvensresponsen fra transferfunksjonen

For en generell transferfunksjon:

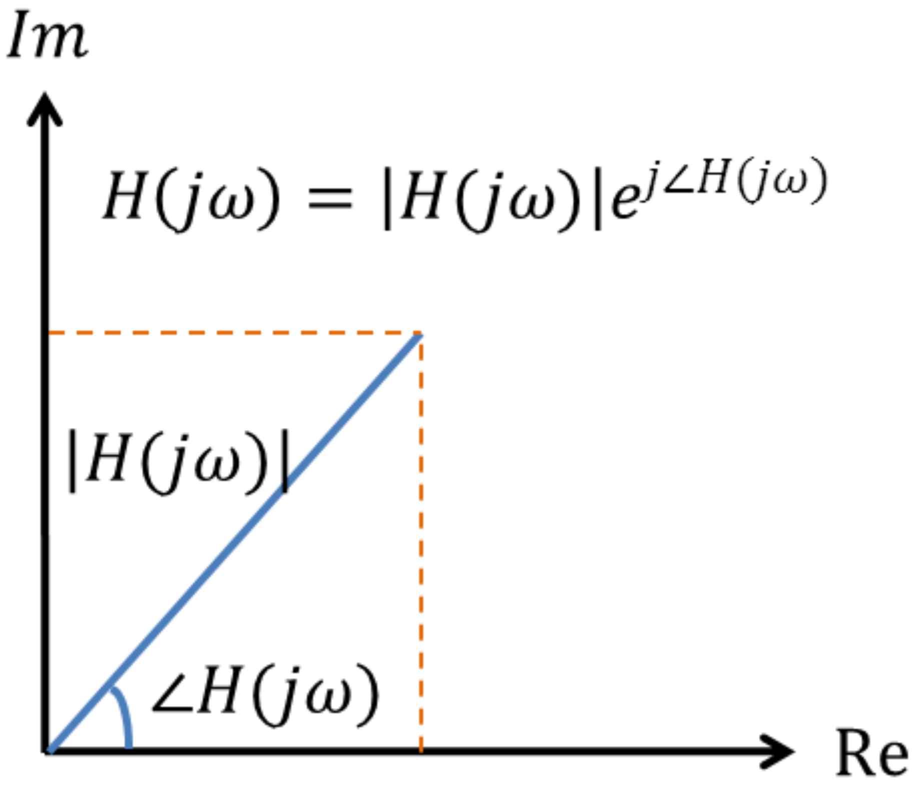
$$H(S) = \frac{y(s)}{u(s)}$$

Har vi følgende (setter $s = j\omega$):

$$\boxed{H(j\omega) = |H(j\omega)|e^{j\angle H(j\omega)}}$$

Hvor $H(j\omega)$ er frekvensresponsen til systemet, dvs. **Vi finner frekvensresponsen ved å sette $s = j\omega$ i transferfunksjonen**.

Dette kan illustreres i det komplekse planet slik:



Et Bodediagram viser en grafisk presentasjon av frekvensresponsen, og er nyttig ved analyse og design av reguleringssystemet. Bodediagrammet består av 2 forskjellige plot, Amplitudediagram, $A(\omega)$ and fasedigram, $\phi(\omega)$.

Amplitudefunksjonen:

$$\boxed{A(\omega) = |H(j\omega)|}$$

Fasefunksjonen:

$$\boxed{\phi(\omega) = \angle H(j\omega)}$$

Merk! $A(\omega)$-saksen er i desibel (dB), hvor desibelverdien av x beregnes som følger:

$$\boxed{x[dB] = 20log_{10}x}$$

Merk! $\phi(\omega)$-aksen er i grader (ikke i radianer!)

Eksempel:

Vi finner uttrykkene for $A(\omega)$ $[dB]$ og $\phi(\omega)$ for ulike transferfunksjoner:

Transferfunksjon:	$A(\omega)$ **og** $\phi(\omega)$**:**

$H(s)=\frac{y(s)}{u(s)}=\frac{1}{s+1}$	$\lvert H(j\omega)\rvert_{dB}=\underline{20log1-20log\sqrt{(\omega)^2+1}}$ $\angle H(j\omega)=\underline{-\arctan(\omega)}$
$H(s)=\frac{4}{2s+1}$	$\lvert H(j\omega)\rvert_{dB}=\underline{20log4-20log\sqrt{(2\omega)^2+1}}$ $\angle H(j\omega)=\underline{-\arctan(2\omega)}$
$H(S)=\frac{5}{(s+1)(10s+1)}$	$\lvert H(j\omega)\rvert_{dB}=\underline{20log5-20log\sqrt{(\omega)^2+1}-20log\sqrt{(10\omega)^2+1}}$ $\angle H(j\omega)=\underline{-\arctan(\omega)-\arctan(10\omega)}$
$H(S)=\frac{1}{s(s+1)^2}$	$\lvert H(j\omega)\rvert_{dB}=-20log\sqrt{(\omega)^2}-2x20log\sqrt{(\omega)^2+1}$ $=\underline{20log\omega-40log\sqrt{(\omega)^2+1}}$ $\angle H(j\omega)=\underline{-90-2\arctan(\omega)}$
$H(s)=\frac{3.2e^{-2s}}{3s+1}$	$\lvert H(j\omega)\rvert_{dB}=\underline{-20log3.2-20log\sqrt{(3\omega)^2+1}}$ $\angle H(j\omega)=\underline{-2\omega-\arctan(3\omega)}$
$H(S)$ $=\frac{5s+1}{(2s+1)(10s+1)}$	$\lvert H(j\omega)\rvert_{dB}$ $=\underline{20log\sqrt{(5\omega)^2+1}-20log\sqrt{(2\omega)^2+1}-20log\sqrt{(10\omega)^2+1}}$ $\angle H(j\omega)=\underline{\arctan(5\omega)-\arctan(2\omega)-\arctan(10\omega)}$

Merk! For å finne fasen i grader må vi multiplisere med: $\frac{180}{\pi}$

Fasekurven:

→ Ledd i telleren: $+90$ grader for hvert ledd (ledd av typen s eller $(Ts+1)$)

→Ledd i nevneren: -90 for hvert ledd (ledd av typen s eller $(Ts+1)$)

[Slutt på eksempel]

Knekkfrekvenser

Knekkfrekvensen(e) er der hvor kurven skifter retning, enten oppover eller nedover. Disse kan enkelt finnes fra transferfunksjonen og gir et godt bilde av hvordan kurven vil bli.

Eksempel:

Her er noen eksempler på knekkfrekvenser for ulike transferfunksjoner:

Transferfunksjon:	Knekkfrekvenser:
$H(s)=\frac{y(s)}{u(s)}=\frac{1}{s+1}$	$\omega=\frac{1}{T}=\frac{1}{1}=\underline{1}$

$H(s)=\frac{4}{2s+1}$	$\omega=\frac{1}{T}=\frac{1}{2}=\underline{0.5}$
$H(S)=\frac{5}{(s+1)(10s+1)}$	$\omega_1=\frac{1}{T_1}=\frac{1}{1}=\underline{1}$ $\omega_2=\frac{1}{T_2}=\frac{1}{10}=\underline{0.1}$
$H(S)=\frac{1}{s(s+1)^2}$	$\omega=\frac{1}{T}=\frac{1}{1}=\underline{1}$
$H(s)=\frac{3.2e^{-2s}}{3s+1}$	$\omega=\frac{1}{T}=\frac{1}{3}=\underline{0.33}$
$H(S)=\frac{5s+1}{(2s+1)(10s+1)}$	$\omega_1=\frac{1}{T_1}=\frac{1}{5}=\underline{0.2}$ $\omega_2=\frac{1}{T_2}=\frac{1}{2}=\underline{0.5}$ $\omega_3=\frac{1}{T_3}=\frac{1}{10}=\underline{0.1}$

[Slutt på eksempel]

MathScript

Selv om MathScript kan finne frekvensresponsen og tegne et Bodediagram direkte fra transferfunksjonen vha. den innebygde funksjonen **bode** kan det også være av interesse å vite hvordan dette gjøres "manuelt", dvs. ta utgangspunkt i de matematiske uttrykkene for $A(\omega)$ og $\phi(\omega)$.

Eksempel:

Gitt følgende transferfunksjon

$$H(s)=\frac{y(s)}{u(s)}=\frac{1}{s+1}$$

Matematiske uttrykk for $A(\omega)$ og $\phi(\omega)$ er gitt ved:

$$|H(j\omega)|_{dB}=\underline{20log1-20log\sqrt{(\omega)^2+1}}$$

$$\angle H(j\omega)=\underline{-\arctan(\omega)}$$

MathScript:

Vi finner $A(\omega)$ og $\phi(\omega)$ for $\omega=1$:

```
w=1;

gain = 20*log10(1) - 20*log10(sqrt(w^2+1))
phase = -atan(w);
phasedeg = phase * 180/pi %convert to degrees
```

Svaret blir:

```
gain =

          -3.0103

phasedeg =

          -45
```

Hvis vi ønsker å finne $A(\omega)$ og $\phi(\omega)$ for flere frekvenser kan vi f.eks. bruke en For loop:

```
w = [0.01, 0.1, 1, 10, 100];
N = length(w);

for i=1:N
     gain(i) = 20*log10(1) - 20*log10(sqrt(w(i)^2+1));
     phase(i) = -atan(w(i));
     phasedeg(i) = phase(i) * 180/pi; %convert to degrees
end
```

Alternativt kan vi gjøre det slik:

```
gain = 20*log10(1) - 20*log10(sqrt(w.^2+1))
phase = -atan(w);
phasedeg = phase * 180/pi %convert to degrees
```

MathScript er et kraftig verktøy som kan jobbe direkte med vektorer ved å bruke følgende notasjon:

```
.^

.*

./
```

Svaret blir i begge tilfellene:

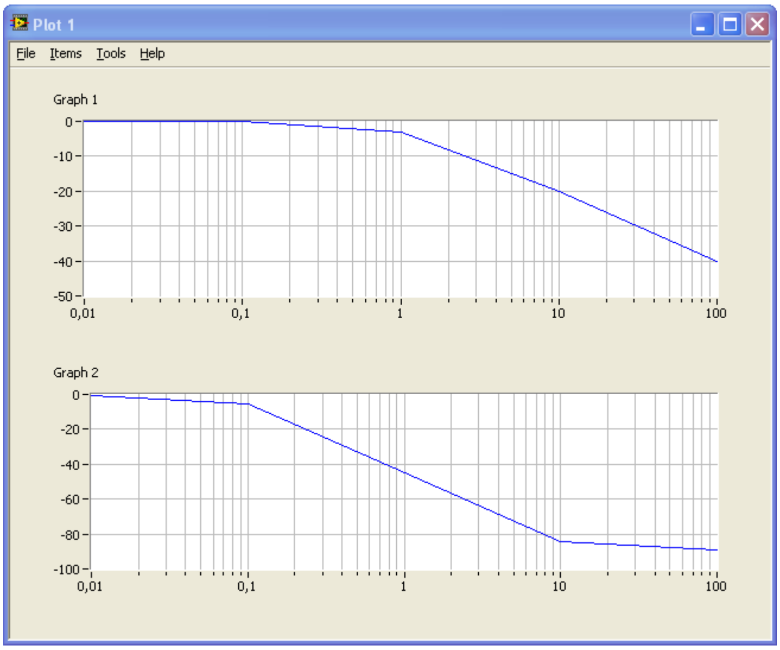
ω	$A(\omega)[dB]$	$\phi(\omega)(degrees)$
0.01	0	-0.6
0.1	0.04	-5.7
1	-3	-45

10	-20	-84
100	-40	-89

Disse kan vi plotte:

```
…
%Gain Plot
subplot(2,1,1)
semilogx(w, gain)
grid

%Phase Plot
subplot(2,1,2)
semilogx(w, phasedeg)
grid
```

Som gir:



[Slutt på eksempel]

Hvordan finne frekvensresponsen fra sinuskurver på inngangen og utgangen?

Vi kan finne frekvensresponsen for et gitt system ved å logge inngangs og utgangsdata for forskjellige frekvenser, og utfra dette beregne oss til verdier for $A(\omega)$ og $\phi(\omega)$ for utvalgte frekvenser.

Inngangssignalet er gitt som:

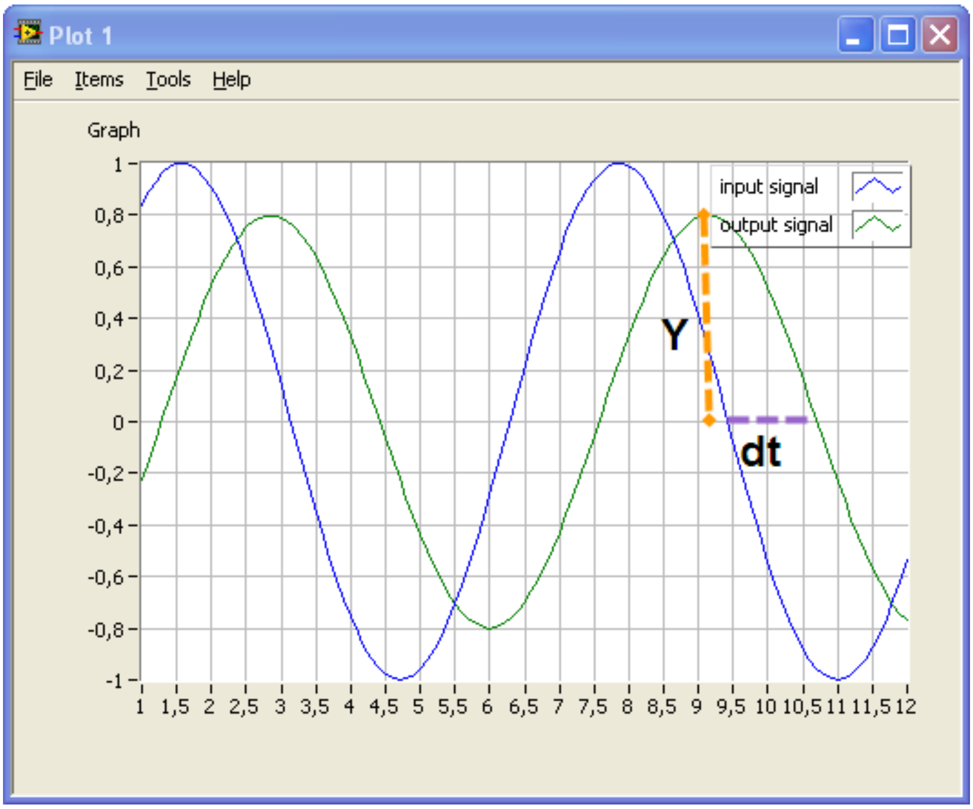
$$\boxed{u(t) = U\ sin\omega t}$$

Steady-state utgangssignalet blir da:

$$\boxed{y(t) = \underbrace{UA}_{Y}\ sin\ (\omega t + \phi)}$$

(dvs. Det er kun amplituden/forsterkningen som er forskjellig, samt at de er faseforskyvet i forhold til hverandre)

Vi vil da få et plot slik som dette for hver frekvens:



Forsterkningen er gitt ved:

$$\boxed{A = \frac{Y}{U}}$$

Der U er gitt av inngangssignalet, mens Y leses av fra plottet.

Faseforskyvningen er gitt ved:

$$\boxed{\phi = -\omega \Delta t\ [rad]}$$

Der frekvensen ω er gitt av inngangssignalet, mens Δt leses av fra plottet.

Frekvensrespons for standardfunksjoner

Vi vil se nærmere på frekvensresponsen for en del standardfunksjoner.

- Forsterker
- Integrator
- Derivator
- 1.ordens system
- 2.ordens system
- Nullpunkt
- Tidsforsinkelse

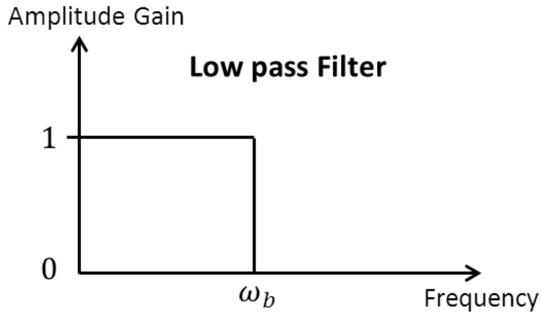
Filtere

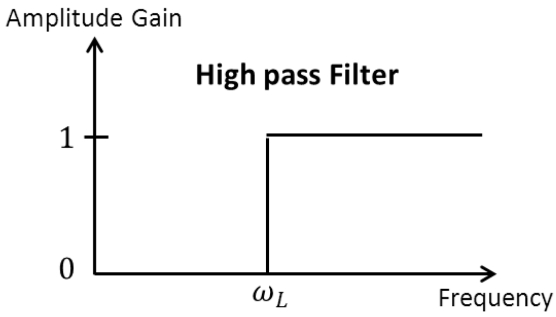
En viktig anvendelse av frekvensrespons er ifm. filtere. Et filter brukes til å fjerne gitte frekvenser eller et intervall med frekvenser fra et signal. En slik anvendelse vil typisk være å fjerne støy fra et signal. Det mest vanlige er Lavpassfilteret.

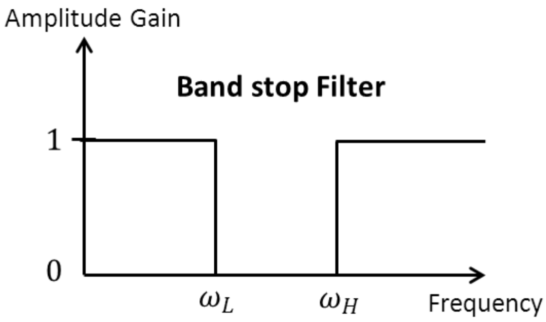
Vi har 4 typer filtere:

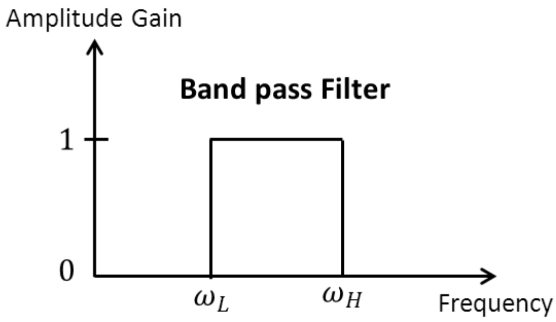
- Lavpassfilter
- Høypassfilter
- Båndstoppfilter
- Båndpassfilter

Nedenfor ser vi de ideelle filterkarakteristikkene for disse 4 typene:









Filtrene vil slippe gjennom signaler for de frekvensene hvor amplitudeforsterkningen er 1 på figuren (eller $0dB$). For virkelige filtere vil ikke overgangen være så skarpe som på bildet over (ideelle).

Lavpassfilter

Transferfunksjon for et Lavpassfilter:

$$H(s) = \frac{1}{Ts+1} = \frac{1}{\frac{1}{\omega_b}s+1}$$

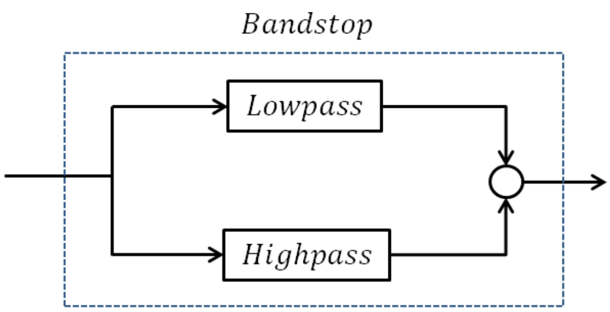
Der ω_b er definert som filterets båndbredde.

Høypassfilter

Transferfunksjon for et Høypassfilter:

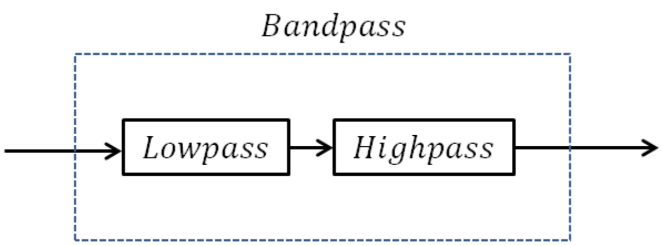
$$H(s) = \frac{Ts}{Ts+1} = \frac{\frac{1}{\omega_b}s}{\frac{1}{\omega_b}s+1}$$

Båndstopp



For dette filteret definerer vi en nedre frekvens ω_L og en øvre frekvens ω_H.

Båndpass



For dette filteret definerer vi en nedre frekvens ω_L og en øvre frekvens ω_H.

MathScript

Eksempel:

MathScript kode:

Lavpassfilter:

```
wl = 1;
Tl = 1/wl;

num = [1];
den = [Tl, 1];

H_lowpass = tf(num, den)

figure(1)
bodemag(H_lowpass)
grid
title('Lowpass Filter')
```

Høypassfilter:

```
wh = 10;
Th = 1/wh;

num = [Th, 0];
den = [Th, 1];

H_highpass = tf(num, den)

figure(2)
bodemag(H_highpass)
grid
title('Highpass Filter')
```

Båndstoppfilter:

Vi parallellkobler et Lavpassfilter og et Høypassfilter:

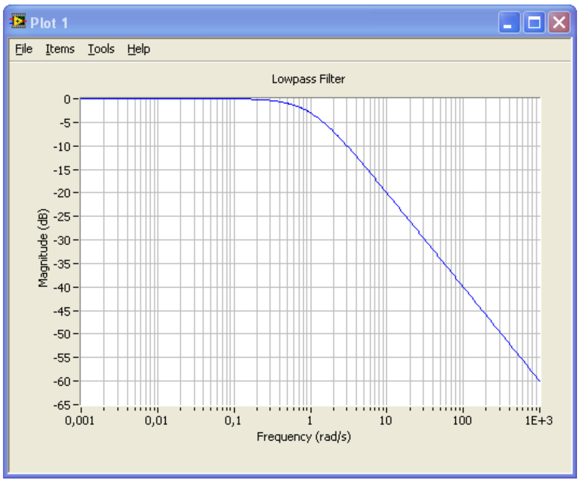
```
H_bandstop = parallel(H_lowpass, H_highpass)

figure(3)
bodemag(H_bandstop)
grid
title('Bandstop Filter')
```

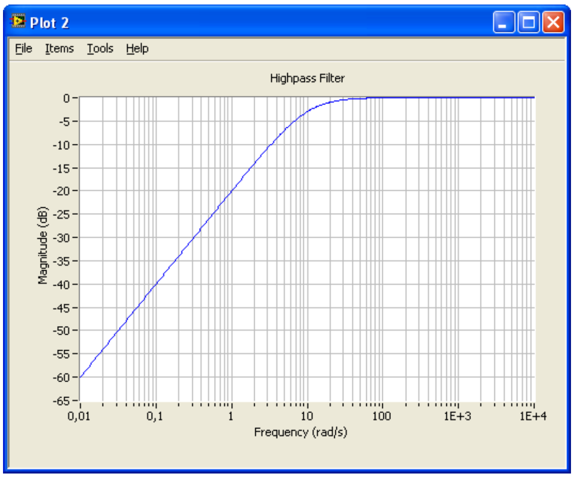
Båndpassfilter:

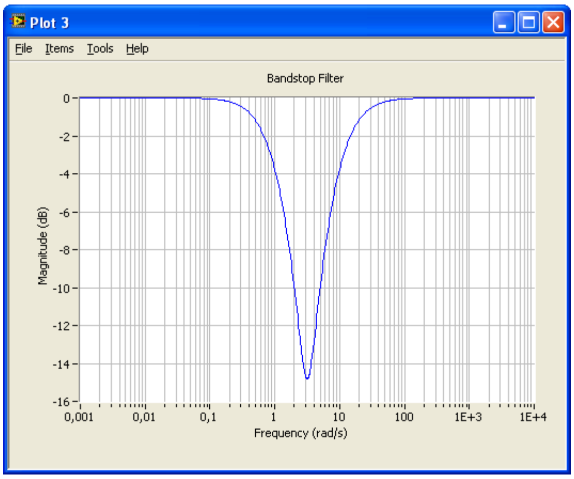
Vi seriekobler et Lavpassfilter og et Høypassfilter:

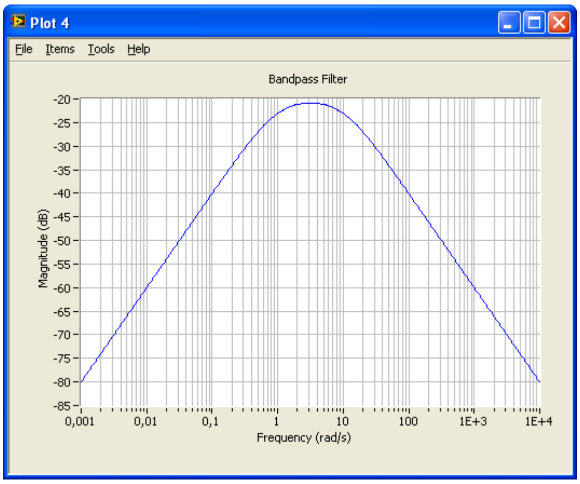
```
H_bandpass = series(H_lowpass, H_highpass)

figure(4)
bodemag(H_bandpass)
grid
title('Bandpass Filter')
```

Plot:





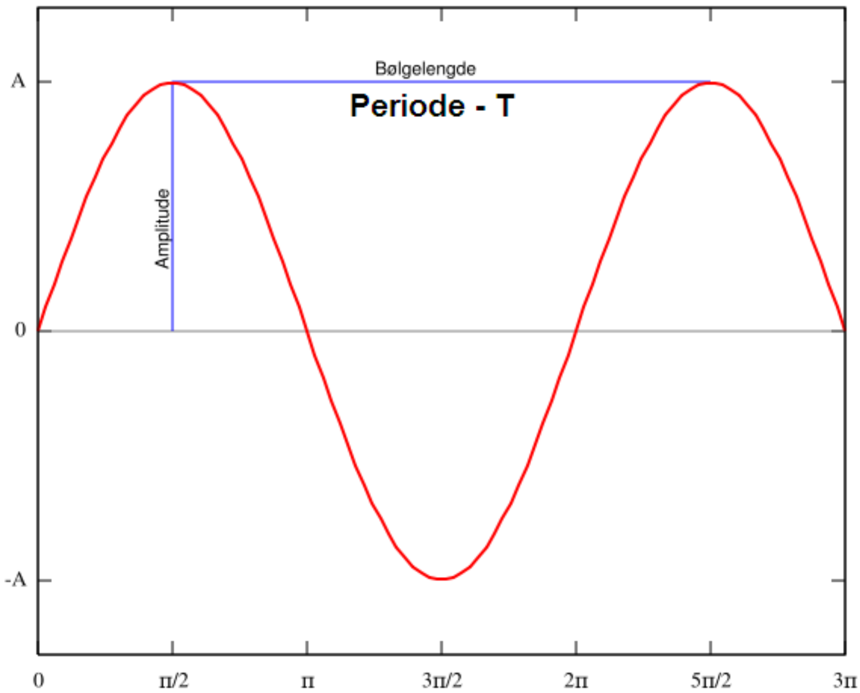




[Slutt på Eksempel]

Definisjoner

Gitt følgende sinuskurve:



Periode – T

Avstanden mellom 2 topper.

Amplitude - A

Amplitude er avstanden fra en bølges største utslag (toppunkt eller bunnpunkt) til likevektstilstanden (midten). Amplituden er altså en positiv verdi. For sinus- og cosinusbølger er amplituden halvparten av høyden mellom toppunkt og bunnpunkt.

Frekvens - f

Frekvens er et mål på antallet ganger en hendelse gjentar seg i løpet av en enhetstid. For å beregne frekvens, setter man et fast tidsintervall, teller antall ganger en hendelse inntreffer og dividerer på tidsintervallets lengde.

Så lenge hendelsene inntreffer regelmessig er en enkel metode for å beregne frekvens å måle tiden mellom to ganger hendelsen inntreffer (perioden) og så beregne frekvensen f som den inverse av denne tiden:

$$\boxed{f = \frac{1}{T}}$$

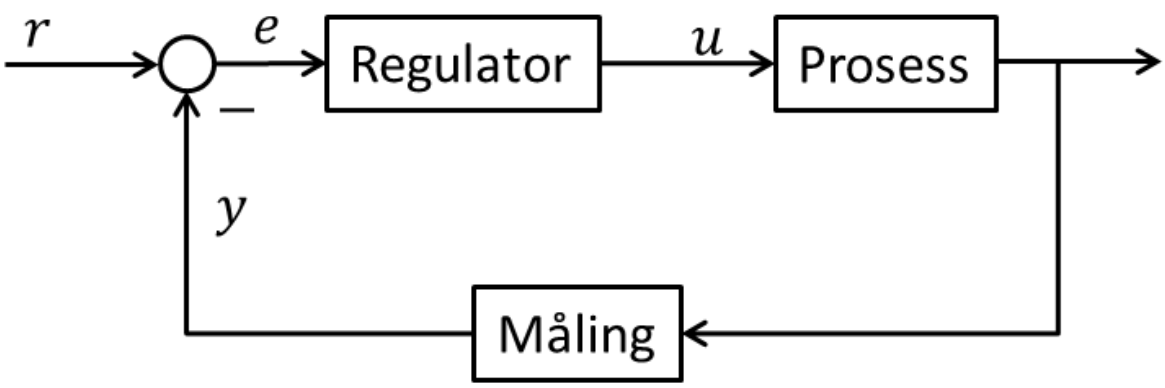
Vanligvis er enheten for frekvens $\mathrm{Hertz}\ [Hz]$, mens i frekvensrespons brukes radianer $\omega\ [rad]$. Sammenhengen mellom disse er:

$$\boxed{\omega = 2\pi f}$$

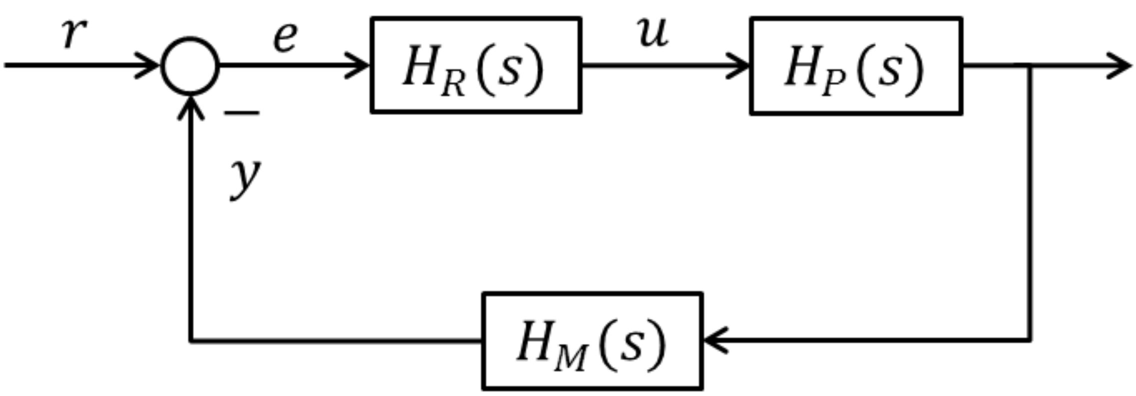
Frekvensresponsanalyse

Innledning

Skisse av et tilbakekoblet system:



Der Regulatoren, Prosessen og Måleinstrumentet er transferfunksjoner:



Vi har da følgende:

<u>Sløyfetransferfunksjonen</u> (Engelsk: "Loop transfer function"):

$$\boxed{L(s) = H_R H_P H_M}$$

Følgeforholdet (Engelsk: “Tracking transfer function”):

$$\boxed{T(s) = \frac{y(s)}{r(s)} = \frac{H_R H_P H_M}{1 + H_R H_P H_M} = \frac{L(s)}{1 + L(s)}}$$

Denne uttrykker hvor godt systemet følger referansen. Systemet har gode følgeegenskaper hvis $y \approx r$, dvs.:

$$|T| \approx 1$$

Sensitivitetsfunksjonen/Avviksforholdet (Engelsk: “Sensitivity transfer function”):

$$\boxed{S(s) = \frac{e(s)}{r(s)} = \frac{1}{1 + L(s)} = 1 - T(s)}$$

Denne uttrykker hvor “sensitivt” avviket er overfor referansen og denne bør derfor være “liten”, dvs.:

$$|S| \approx 0 \; eller \; |S| \ll 1$$

Merk!

$$S(s) = 1 - T(s) \leftrightarrow T(s) = 1 - S(s)$$

og

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

Følgeegenskaper

I frekvensplanet har vi:

Gode følgeegenskaper:

$$|T(j\omega)| \approx 1 \; eller \; |S(j\omega)| \ll 1$$

Eller:

$$\boxed{|L(j\omega)| \gg 1}$$

Dårlige følgeegenskaper:

$$|T(j\omega)| \ll 1 \; eller \; |S(j\omega)| \approx 1$$

Eller:

$$\boxed{|L(j\omega)| \ll 1}$$

Kryssfrekvensen:

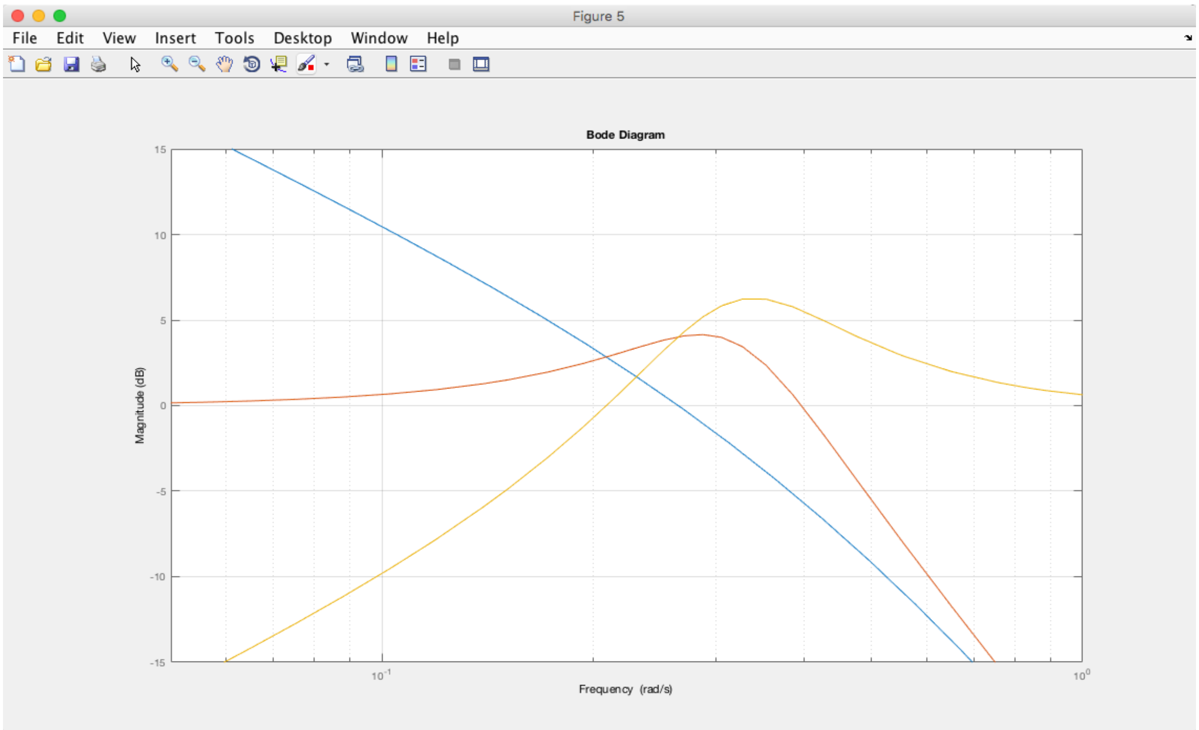
På bakgrunn av dette kan vi definere kryssfrekvensen ω_c:

$$\boxed{|L(j\omega_c)| = 1 = 0dB}$$

→ Et reguleringssystems følgeegenskaper er *gode* for frekvenser *under* kryssfrekvensen, mens de er *dårlige* for frekvenser *over* kryssfrekvensen.

Båndbredde

Når vi plotter L, T og S vil vi typisk få et Bodeplot som dette (Kun amplitudeplottet er interessant i denne sammenhengen):



Ut fra dette definerer vi følgende båndbredder:

$\boldsymbol{\omega_c}$ – Kryssfrekvensen – Frekvensen der sløyfetransferfunksjonen $L(j\omega)$ har følgende verdi:

$$1 = \underline{0dB}$$

$\boldsymbol{\omega_t}$ – Frekvensen der forsterkningen til Følgefunksjonen $T(j\omega)$ har følgende verdi:

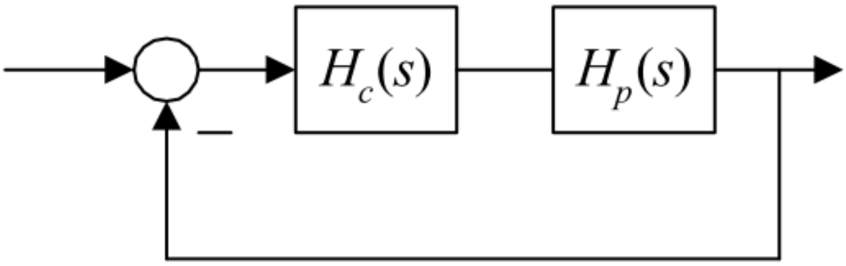
$$\frac{1}{\sqrt{2}} \approx 0.71 = \underline{-3dB}$$

$\boldsymbol{\omega_s}$ – Frekvensen hvor forsterkningen til sensitivitetsfunksjonen $S(j\omega)$ har følgende verdi:

$$1 - \frac{1}{\sqrt{2}} \approx 0.29 = \underline{-11dB}$$

MathScript

Gitt følgende reguleringssystem:



Sløyfetransferfunksjonen:

```
L = series(Hc, Hp)
```

Hvis flere enn 2: M=series(H1, series(H2, H3))

Følgeforholdet:

```
T = feedback(L, 1)
```

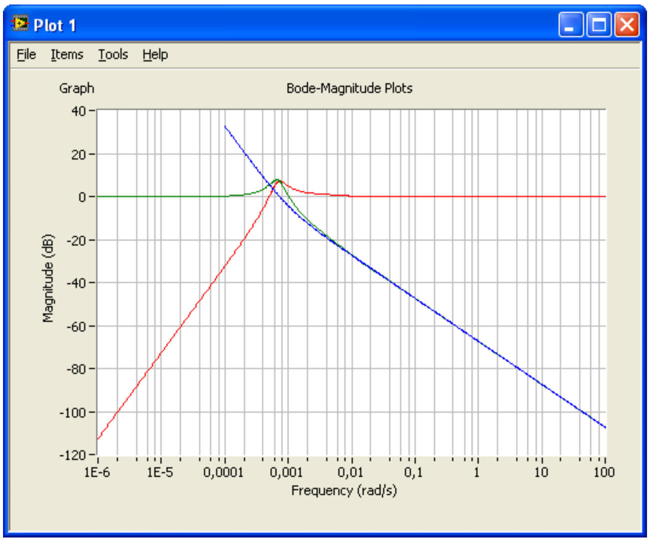
Sensitivitetsfunksjonen/Avviksforholdet:

```
S = 1-T
```

Tilslutt tegner vi disse tre funksjonene i et Bodediagram (kun amplitudeplottet):

```
bodemag(L,T,S)
```

I MathScript vil plottet typisk se slik ut:



Ut fra dette plottet kan vi lese av båndbreddene ω_c, ω_t og ω_s.

I MathScript kan vi enkelt forandre på aksene slik at det blir enklere å lese av verdiene for kurven.

Stabilitetsanalyse i Frekvensplanet

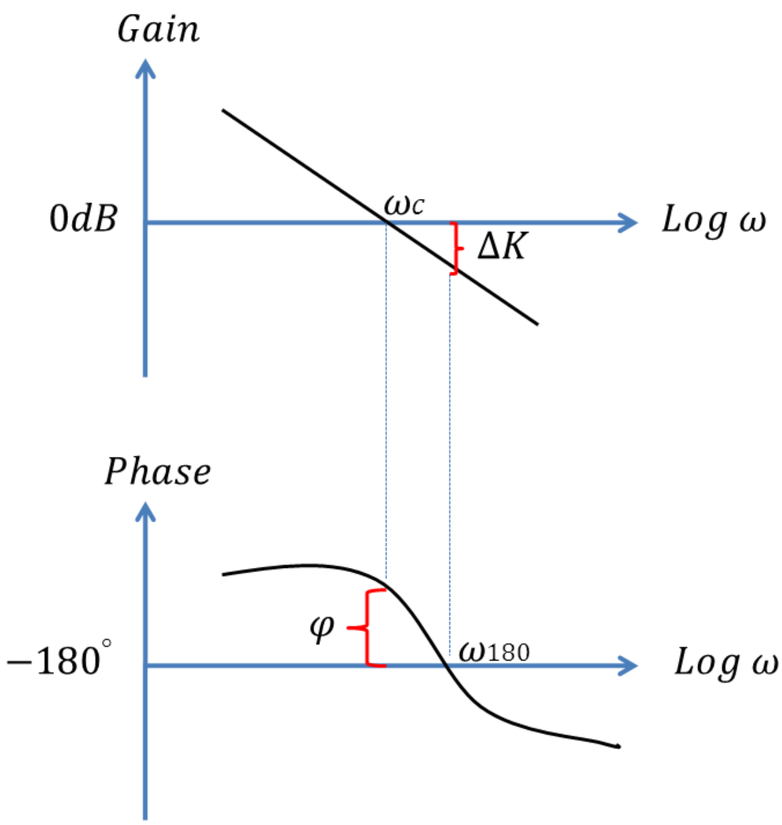
Et system kan ha en av følgende stabilitetsegenskaper:

- **Asymptotisk stabilt system**
- **Marginalt stabilt system**
- **Ustabilt system**

Stabilitetsmarginer

Forsterkningsmargin ($GM/\Delta K$) og Fasemarginen (PM/φ) er viktige mtp. analyse av stabilitetsegenskaper for tilbakekoblede systemer.

Vi tegner følgende Bodediagram:



Ut fra dette definerer vi følgende:

Kryssfrekvensen:

Kryssfrekvensen ω_c:

$$\boxed{|L(j\omega_c)| = 1 = 0dB}$$

Fasekryssfrekvensen ω_{180}:

$$\boxed{\angle L(j\omega_{180}) = -180^o}$$

Forsterkningsmargin:

Forsterkningsmargin $\mathbf{GM}/\boldsymbol{\Delta K}$:

$$\Delta K = \frac{1}{|L(j\omega_{180})|}$$

Eller:

$$\boxed{GM\ [dB] = -|L(j\omega_{180})|\ [dB]}$$

Forsterkningsmargin ($GM/\Delta K$) sier hvor mye sløyfeforsterkningen kan øke før systemet blir ustabilt.

Fasemargin:

Fasemarginen PM/φ:

$$\boxed{PM = 180^o + \angle L(j\omega_c)}$$

Fasemarginen (PM/φ) sier hvor mye faseforskyvningen kan reduseres før systemet blir ustabilt.

Stabilitetsanalyse

Ut fra dette definerer vi følgende stabilitetsegenskaper:

Asymptotisk stabilt system:	$\omega_c < \omega_{180}$
Marginalt stabilt system:	$\omega_c = \omega_{180}$
Ustabilt system:	$\omega_c > \omega_{180}$

MathScript

I MathScript kan vi lese av ω_c, ω_{180} ΔK, φ fra Bodediagrammet som vi finner ved å bruke **bode** funksjonen i MathScript.

En annen metode er å bruke funksjonen **margin**.

Eksempel:

Slyøfetransferfunkjonen er gitt som følger:

$$L(S) = \frac{K_p}{s(2s+1)(5s+1)}$$

Vi setter $K_p = 0.1$

Vi starter med å definere transferfunksjonen:

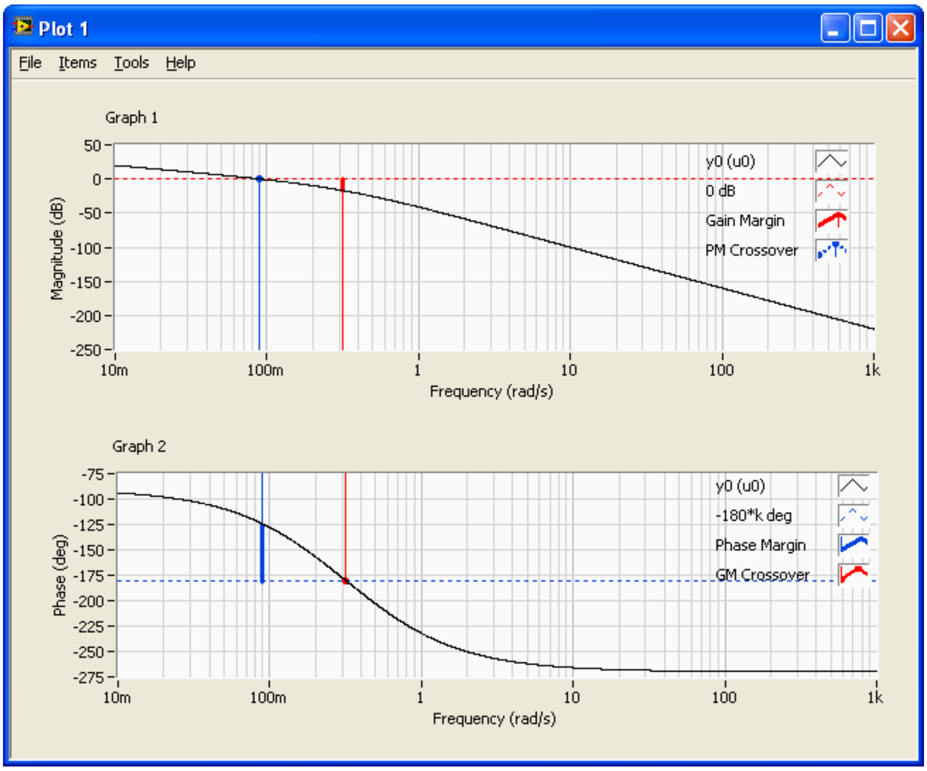
```
Kp = 0.1;
Num = [Kp;
den1 = [1,0];
den2 = [2,1];
den3 = [5,1];
den = conv(den1, conv(den2,den3));
L = tf(num,den)
```

Deretter finner vi ω_c, ω_{180} ΔK, φ:

```
margin(L)
```

der $L(s)$ er sløyfetransferfunksjonen.

Når vi bruker margin funksjonen på denne måten, får vi et Bodediagram med ω_c, ω_{180} ΔK, φ ferdig tegnet inn:



Vi kan også få tallverdier for ω_c, ω_{180} $\Delta K\ (GM)$, $\varphi\ (PM)$ direkte ved å bruke **margin** funksjonen slik:

```
[gm, pm, w180, wc] = margin(L)
```

Merk! Det er feil i hjelpeteksten som dukker opp når du skriver "help margin" i Command window i MathScript. Det riktige er rekkefølgen som er illustrert i eksemplet over!

Hvis vi ønsker GM i dB gjør vi følgende:

```
gmdB = 20*log10(gm)
```

Dette gir følgende resultat:

$$\omega_c = 0.09rad$$

$$\omega_{180} = 0.3rad$$

$$\Delta K\ (GM) = 6.99\ eller\ 16.9dB$$

$$\varphi\ (PM) = 55\ grader$$

Vi ser at $\omega_c < \omega_{180}$ → Asymptotisk stabilt system

Vi kan øke sløyfeforsterkningen med 6.99 før systemet blir ustabilt, dvs. for $K_p = 0.7$ og større vil systemet være ustabilt.

Hvor store bør stabilitetsmarginene være?

2 egenskaper som for et system som stabilitetsmarginene kan gi utrykk for:

- Innsvingningsforløpet i reguleringssystemets dynamiske responser
- Reguleringssystemets robusthet overfor visse parameterendringer i sløyfen. Med robusthet menes reguleringssystemets evne til å opprettholde visse egenskaper til tross for parameterendringer.

Gylden regel:

Stabilitetsmarginene bør ligge i området:

$$2\ (6dB) < \Delta K < 4\ (12dB)$$

$$30 grader < \varphi < 60\ grader$$

Så i eksemplet vårt kan K_p økes noe, f.eks. $K_p = 0.2$

[Slutt på eksempel]

MathScript-funksjoner

Her er en kort oversikt over MathScript funksjoner som vi har benyttet.

Funksjon	Beskrivelse	Eksempler
atan	Beregner arctangens til x	`>atan(x)`

bode	Lager et Bodediagram for en gitt transferfunksjon. Eller den kan returnere forsterkning og faseforskyvning for gitte frekvenser. Hvis du ikke spesifiserer noen utganger fra funksjonen vil funksjonen tegne et Bodediagram.	`>num = [4];`
		`>den = [2, 1];`
		`>H = tf(num, den)`
		`>bode(H)`

bodemag	Nesten same som bode funksjonen, men denne lager kun forsterkningsplottet – ikke faseforskyvningsplottet.	`>[mag, wout] = bodemag(SysIn)`
		`>[mag, wout] = bodemag(SysIn, [wmin wmax])`
		`>[mag, wout] = bodemag(SysIn, wlist)`

conv	Kan brukes når du har komplekse transferfunksjoner som du må slå sammen.	`>den1 = [1, 2, 3];`
		`>den2 = [3, 4];`
		`>den = conv(C1, C2)`

feedback	Lager transferfunksjonen for et tilbakekoblet system.	`>T = feedback(H1, H2)`
log10	Beregner 10'er logaritmen.	`>log(x)`

lsim	Brukes til å simulere et lineært system.	`>t = [0:0.1:10]`
		`>u = sin(0.1*pi*t)'`
		`>lsim(H, u, t)`

margin	Tegner et Bodediagram med forsterningsmargin og fasemargin tegnet inn.	`>num = [1]`
		`>den = [1, 5, 6]`
		`>H = tf(num, den)`
		`margin(H)`

margin	Denne funksjonen kan også beregne kryssfrekvens, fasekryssfrekvensen, forsterkningsmargin og fasemargin	`>[gmf, gm, pmf, pm] = margin(H)`
pade	Brukes til å finne transferfunksjonen til en Pade' aproksimasjon.	`>[num, den] = pade(delay, order)` `>[A, B, C, D] = pade(delay, order)`

pid	Brukes til å lage transferfunksjonen til en P, PI, PD eller PID regulator.	`>Kc = 0.5;`
		`>Ti = 0.25;`
		`>H = pid(Kc, Ti, 'academic');`

plot	Brukes til å generere et plot.	`>x = [0:0.01:1];`
		`>y = sin(x)`
		`>plot(x, y)`

poles	Finner polene for en transferfunksjon	`>num = [1]`
		`>den = [1,1]`
		`>H = tf(num,den)`
		`>poles(H)`

semilogx	Kan brukes til å tegne Bodediagram "manuelt" basert på de matematiske uttrykkene for $A(\omega)$ og $\phi(\omega)$	`>semilogx(w, gain)`
series	Sla sammen 2 eller flere transferfunksjoner som er i serie	`>H = series(H1,H2)`

ss	Lager en tilstandsrommodell fra A, B, C og D matrisene.	`>A = [1, 2; 3; 4]`
		`>B = [0; 1]`
		`>C = [1, 0]`
		`>model = ss(A, B, C)`

ssinfo	Brukes til å returnere A, B, C og D matrisene i tilstandsrommodellen.	`>A = [1, 1; -1, 2]`
		`>B = [1, 2]'`
		`>C = [2, 1]`
		`>D = 0`
		`>model = ss(A, B, C, D)`
		`>[A, B, C, D, Ts] = ssinfo(model)`

step	Sprangrespons	`>num=[1,1];`
		`>den=[1,-1,3];`
		`>H=tf(num,den);`
		`>t=[0:0.01:10];`
		`>step(H,t);`

Sys_order1	Lager en 1.ordens transferfunksjon, med eller uten tidsforsinkelse.	`>K = 1;`
		`>T = 2;`
		`>H = sys_order1(K, T)`
		
		`>delay = 3;`
		`>H = sys_order1(K, T, delay)`

Sys_order2	Lager en 2.ordens transferfunksjon.	`>z = 0.5`
		`>w = 20`
		`>[num, den] = sys_order2(w, z)`

		>H = tf(num, den) >[A, B, C, D] = **sys_order2**(wn, dr) >SysSS = **ss**(A, B, C, D)
tf	Brukes til å lage en transferfunksjon. Kan også brukes til å transformere en tilstandsrommodell til en transferfunksjon.	>num=[1]; >den=[1, 1, 1]; >H = **tf**(num, den)
tfinfo	Brukes til å returnere teler, nevner og eventuelt dødtid fra en transferfunksjon.	>[num, den, delay, Ts] = **tfinfo**(H)

Skriv "**help <funksjonsnavn>**" i Command window for å finne ut hvordan den enkelte funksjon virker i detalj.

Hans-Petter Halvorsen, M.Sc.

E-mail: hans.p.halvorsen@hit.no

Blog: http://home.hit.no/~hansha/

University College of Southeast Norway

www.usn.no