LAB-STA-5. FREQUENCY DOMAIN ANALYSIS AND DESIGN OF CONTROL SYSTEMS

A. OBJECTIVES. 1. To understand the frequency domain tools available in Matlab for analysis and design of control systems.
2. To use Matlab's frequency domain tools in the design of control systems.

B. FREQUENCY RESPONSE METHODS.

In simple, the frequency response of a system is the steady-state output response generated by a sinusoidal input signal. The necessary assumption when working with frequency response characteristics is the linearity and stability of the systems under discussion. Therefore all the treated systems are considered to be linear time-invariant (LTI). For a LTI system with sine wave input the output will always be a sine wave that only differs in magnitude and phase from the input signal.
The main reason of using frequency domain tools for control systems analysis and design is that the frequency response functions can be obtained experimentally and the design can be performed entirely in frequency domain.
Starting with the transfer function representation of a continuous time system, let it be $H(s)$, the steady-state characteristics of the output sine are completely encompassed in the complex number $H(j\omega)$, where ω is the frequency and j is the complex number. This complex number has a magnitude and a phase, both of which can be represented as a function of variable frequency ω. The corresponding variation of ω can be understood as a sweeping frequency of the input sine wave from small frequencies to high frequencies. The continuous time transfer functions case is treated here, with equivalent descriptions for discrete-time transfer functions also being available.
First we study the second order system with normalized transfer function $H(s)=\omega_n^2/(s^2+2\varsigma\,\omega_n\,s+\omega_n^2)$, where ω_n is the natural frequency and ς is the damping ratio (or factor). The reason for studying this transfer function in frequency domain is that the performances of many closed-loop control systems are prescribed in terms of the time domain response of this transfer function. The following code produces the so called Bode diagram in frequency domain.

```
s=tf('s');
wn=1;csi=0.3;
h=wn^2/(s^2+2*csi*wn*s+wn^2);
bode(h),grid
```

The Bode diagram consists of the magnitude of the frequency response as a function of the frequency expressed in decibels [dB] and the phase of the frequency response as a function of frequency ω. Basically, the magnitude and phase result from the polar expression of the complex number $H(j\omega)=|H(j\omega)|\,e^{j\arg(H(j\omega))}$. The magnitude expressed in decibels is obtained as $20\log_{10}(|H(j\omega)|)$. An example of Bode plots is given in Fig. 1; the plots are obtained by running the above Matlab code.

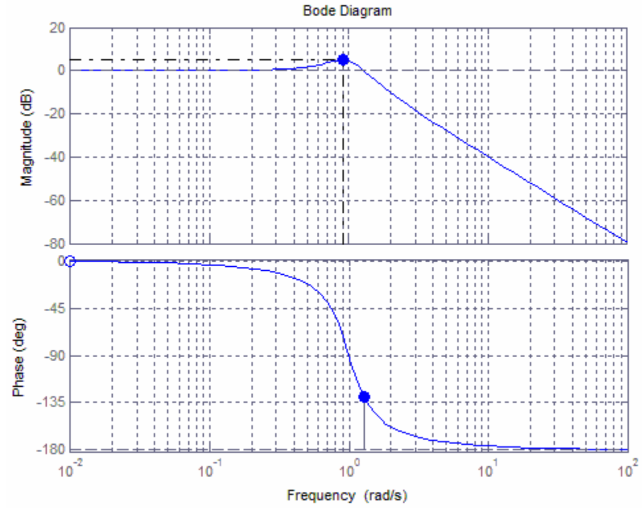

Fig. 1.

For the second order system several metrics defined on the magnitude diagram are useful. The first is the peak magnitude $M_{p\omega}$ found at the resonant frequency ω_r on the x-axis. The crossover frequency ω_c is the frequency where the magnitude plot cuts the 0 dB horizonatal line. The bandwidth is the frequency ω_b after which the magnitude drops below the –3dB threshold.
The interesting fact is that there exists some relations between the time response characteristics of the 2[nd] order system and it's frequency domain metrics. The settling time for the step-response of the system depend on the damping factor and the natural frequency and the overshoot in time-domain is related to the peak magnitude and the damping factor. The following code produces the graphics illustrated in Fig. 2 with the relations between the peak magnitude and damping factor and resonant frequency versus the damping factor:

```
w=logspace(-2,2,200);
mp=[],wr=[]; % initializing to empty vectors
csi=0.15:0.005:0.8;
for i=1:length(csi)
    h=wn^2/(s^2+2*csi(i)*wn*s+wn^2);
    [mag,phase,w]=bode(h,w);
    [mp(i),l]=max(mag);wr(i)=w(l);
end
subplot(121),plot(csi,mp),xlabel('\zeta'),ylabel('M_p_\omega')
subplot(122),plot(csi,wr),xlabel('\zeta'),ylabel('\omega_r/\omega_n')
```

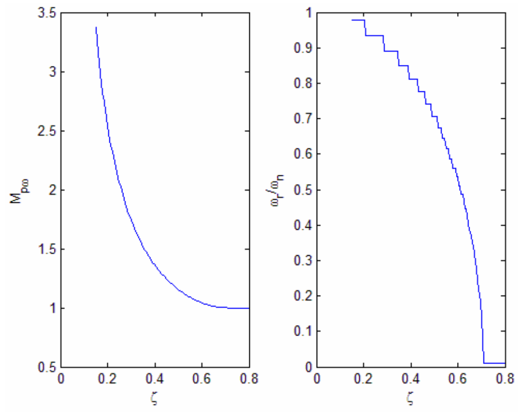

Fig. 2.

The diagrams presented in Fig. 2 are useful since the following two relations account for time performance indices such as settling time T_s and percent overshoot P.O:

$$T_s = \frac{4}{\varsigma \omega_n}, P.O. = 100 \exp\left(\frac{-\varsigma \pi}{\sqrt{1-\varsigma^2}}\right). \tag{1.1}$$

These relations are only rough approximations for higher order systems but can still lead to successful design.

Example 1: For a design example of a control system assume an engraving machine mechanism with actuator and sensor included in the transfer function $H(s) = 1/[s(s+1)(s+2)]$. A closed-loop feedback control system with unity feedback is considered, and a proportional controller is assumed of transfer function $C(s) = K$. The performance requirements are acceptable settling time under 16 sec and percent overshoot under 20%.

The following code shows for $K = 2$ the peak magnitude and the resonant frequency:

```
>> K=2;
h=1/s/(s+1)/(s+2);
cl=feedback(K*h,1,-1);
[mag,phase,w]=bode(cl,w);
[mp,l]=max(mag);wr=w(l);mp,wr

mp =
    1.8368

wr =
    0.8120
```

For the peak magnitude of 1.836 in Fig. 2 this results in a damping factor of 0.285 for which given the resonant frequency of 0.8120 rad/sec the natural frequency in Fig. 2 is ω_n =0.8120/0.9329=0.8704. According to formulas (1.1) the settling time is therefore T_s=16.12 sec and overshoot is about 39.3%. The step response using Matlab using the "**step**" command produces Fig. 3.

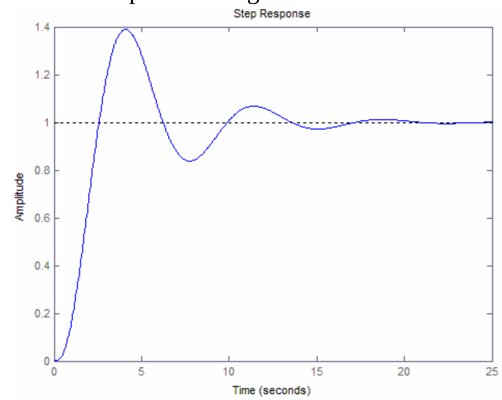

Fig. 3.

The overshoot and the settling time are very close to the estimated values using our approach. However, the current K does not meet the design specs. We can iterate over K to arrive at $K = 1$ for which the design specs are met as shown in Fig. 4.

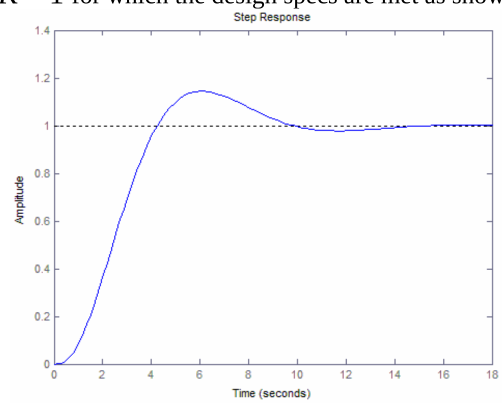

Fig. 4.

C. STABILITY ANALYSIS IN FREQUENCY DOMAIN. NYQUIST STABILITY CRITERION.

For a closed-loop control system, the stability can be inferred from the open-loop Nyquist frequency plot. Assume two cases of control systems with unity feedback, one in which the controller with the transfer function $C(s)$ is in series with the plant with the transfer function $H(s)$ on the feedforward path and the second case where

$C(s)$ is on the feedback path. The corresponding transfer functions for the closed-loop system in the two cases are:

$$T(s)=\frac{C(s)H(s)}{1+C(s)H(s)} \text{ and } T(s)=\frac{H(s)}{1+C(s)H(s)}, \text{respectively}. \qquad (1.2)$$

The denominator polynomial is in both cases the same, namely $1+C(s)H(s)$. Therefore, given the knowledge on $C(s)H(s)$, the stability of $1+C(s)H(s)$ can be deduced. Let the open-loop transfer function be denoted as $L(s)=C(s)H(s)$. The graphical representation of L(s) in the s-plane is called a *contour plot* or *locus* denoted as Γ_L. *The Nyquist stability criterion* can be stated as follows:

The feedback control system is stable if and only if the Γ_L*'s number of counterclockwise encirclements of the point* (–1,0) *is equal to the number of poles of* $L(s)$ *with positive real part.*

The Nyquist plot of a system can be obtained using the command **nyquist**. While the Bode diagram provides separated magnitude and phase plots as a function of frequency (both of which can describe the complex number as a function of frequency in polar form), the Nyquist plot uses the Cartesian plane to plot the complex numbers in the increasing direction of frequency ω. The Nyquist criterion is an absolute criterion giving a yes or no answer about closed-loop stability in terms of the loop transfer function. Some relative stability metrics can be used that underline the connection between the Bode diagam and the Nyquist plot/diagram. A Matlab function called **margin** automatically finds the gain margin and the phase margin.

- The gain margin measures how much the loop gain has to be increased until the locus $L(j\omega)$ passes through the point (–1,0) that is the moment when the closed-loop system becomes unstable.
- The phase margin measures the additional phase lag before the closed-loop system becomes unstable.

The following code plots the Bode diagram and the Nyquist plot of the same system and it leads to Fig. 5:

```
num=0.5;
den=[1 2 1 0.5];
sys1=tf(num,den);
[mag,phase,w]=bode(sys1);
subplot(121),margin(sys1);
subplot(122),nyquist(sys1,w);
```

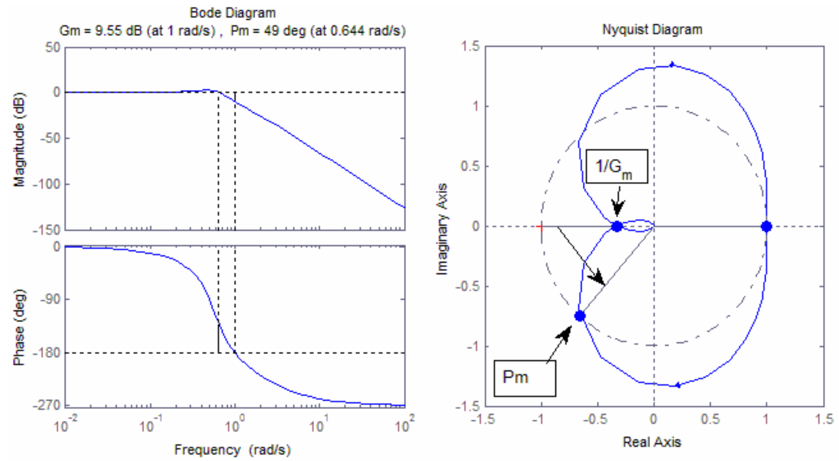

Fig. 5.

In Fig. 5, the gain margin and the phase margin from the Bode diagram are correlated with the Nyquist diagram. On the Nyquist plot, the gain margin is the reciprocal of the distance from the origin to the point on the horizontal axis where the Nyquist plot first cuts the horizontal axis in the increasing direction of the frequency. The first frequency for which the plot crosses the circle with radius 1 centered at origin is the frequency corresponding with the phase margin. The phase margin is measured in degrees as the arrow indicates, starting with the horizontal axis and in counterclockwise direction.

On the Nyquist plot presented in Fig. 5, the application of the Nyquist criterion gives zero encirclements in the counterclockwise direction of the critical point (-1,0). Given that the loop transfer function $L(s)=0.5/(s^3+2s^2+s+0.5)$ has all poles with negative real part we conclude that the system obtained by closing the feedback loop will be stable.

In another example, consider the engraving machine example previously studied in the Example 1. The machine transfer function is $H(s)=1/[s(s+1)(s+2)]$. For a proportional controller with $C(s)=K=1$ and $C(s)=K=10$, the blue and red lines presented in Fig. 6, respectively, show the Nyquist plots for the loop transfer function $L(s)=C(s)H(s)$.

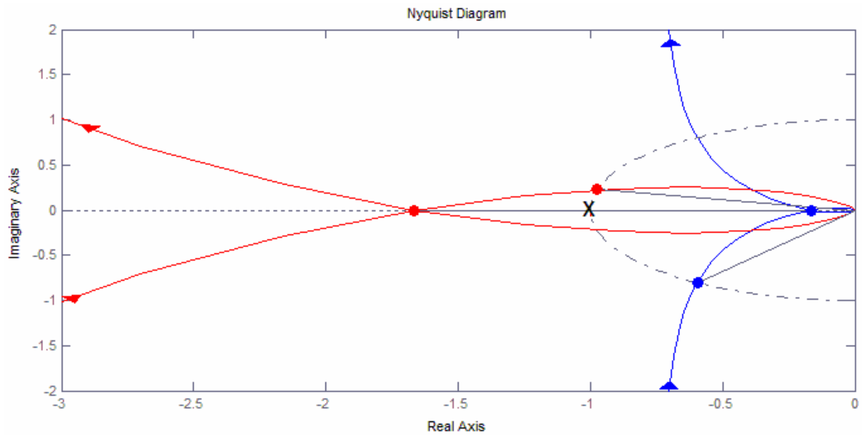

Fig. 6.

Using Fig. 6, the critical point is marked with **X**. The relative stability margins (gain margin and phase margin) are visible with colored dots. In the case where $K=1$ (the blue line), the line does not encircle the critical point so the closed loop will be stable in a feedback configuration. The same is valid for the red line corresponding to $K=10$. Because the encirclement of the critical point is not made in the counterclockwise direction but in the clockwise direction, the number of encirclements counts as zero and is equal to the number of poles of $C(s)H(s)$ in the right half-plane (RHP). Therefore, this feedback connection will also be stable.

In many physical systems, dead-time (or delay) is present and it usually appears in the continuous-time transfer functions as $\exp(-T\,s)$. In order to use the tools in the frequency domain we have to approximate the exponential with a transfer function form with polynomials in numerator and denominator. A so-called **pade** function is such an approximator. In Matlab, **pade(T,n)**, gives the Pade transfer function approximation for the delay of T seconds and of order n. The following code illustrates the approximation:

```
>> [num,den]=pade(1,2);tf(num,den)

ans =

  s^2 - 6 s + 12
  --------------
  s^2 + 6 s + 12

Continuous-time transfer function.
```

References

[1] R.H. Bishop, *Modern Control Systems Analysis and Design Using Matlab*, Addison-Wesley, Boston, MA, USA, 1993.