XMUT220 Signals and Systems

Laboratory 4: Time & Frequency Characteristics

Due Date: Not for marking (no submission required)

A. Impulse and Step Inputs

In this part of the laboratory exercise, we will evaluate the use of impulse and step inputs for analysing the time-and frequency-domain responses of an example continuous-time LTI system.

A.1. Impulse Input

An impulse input is a very high pulse applied to a system over a very short time (i.e., it is not maintained). That is, the magnitude of the input approaches infinity while the time approaches zero.

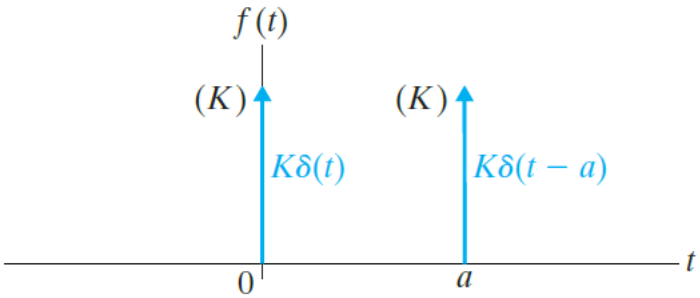


Figure 1: Transient response of an impulse input

Referring to the figure given above, the equations for impulse function at $t = 0$ and at $t = a$ are:

$$f(t) = K\delta(t) \qquad \text{and} \qquad f(t) = K\delta(t-a)$$

An example of an impulse input is a hammer striking a bell; the bell will experience a very high force but over a very short time. The impact will leave the bell swinging back and forth, hence the impulse input has delivered a finite amount of energy to the system. Another example is a car encountering a bump in the road at high speed; the suspension system will experience an impulse input.

A.2. Step Input

A step input is instantaneously applied at some time (typically taken as zero) and thereafter held at a constant level.

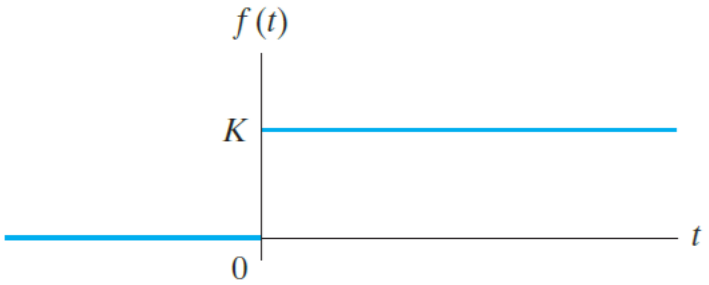


Figure 2: Transient response of a step input

As shown in the figure above, following its mathematical definition, the equation for a step function is:

$$f(t) = \begin{cases} Ku(t) = 0, & t < 0 \\ Ku(t) = K, & t > 0 \end{cases}$$

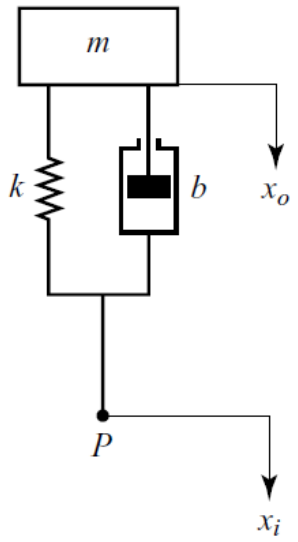
An example is the voltage applied when closing a switch in an electric circuit, or the fluid flow from a pipe when a valve is opened. Before $t = 0$, the voltage and fluid flow were zero, for $t \geq 0$, they have some constant value.

For time-domain simulation of the system in MATLAB, use `tf()` function to define the transfer function of the given system. Use `plot()` function to plot any time-domain graphs that will be required. Impulse and step inputs could be defined in MATLAB using `impulse()` and `step()` functions respectively. See Appendix for further details of these functions.

Practical Exercise 1 – Impulse and Step Response Simulation and Analysis

You are given with a simplified version of a car suspension system. Assume that the motion at point P given in the diagram where x_i is the input to the system and the vertical motion of the body is the output (x_o). Consider that the motion of the body is in the vertical direction only. Displacement is measured from the equilibrium position in the absence of input.

The forces that are acting on the mass-damper-spring parts of the mechanical system are: mass: $f_m = m\ddot{x}_o$; damper: $f_d = b(\dot{x}_o - \dot{x}_i)$; and spring: $f_s = k(x_o - x_i)$, where: m = mass, b = damper coefficient, and k = spring constant. Note: $\dot{x}$ is the first derivative and $\ddot{x}$ is the second derivative of x.



Assuming zero initial conditions and applying the second Newton laws of motion, the differential equation that governs the car suspension system is approximately:

$$m\ddot{x}_o + b(\dot{x}_o - \dot{x}_i) + k(x_o - x_i) = 0$$

a. Derive the transfer function equation of the system in term of $G(\omega) = X_o(\omega)/X_i(\omega)$. [4 marks]

b. Derive the equation for the time response of the system ($x_o(t)/x_i(t)$). [8 marks]

c. Plot the time response of the system in MATLAB when it is subjected to impulse input. Why do we use impulse response to study the natural behaviour of the system? What is the natural behaviour of the given system? Why the response is like that? [8 marks]

d. Plot the time response of the system in MATLAB when subjected to step input. Based on the step response obtained, give two reasons on why the system is a second order. [8 marks]

e. Plot frequency response of the system in MATLAB. Determine the values of the magnitude and phase at the critical frequencies of the frequency response. Explain why the overall frequency response of the system is like a combination of second-order and first-order systems? [10 marks]

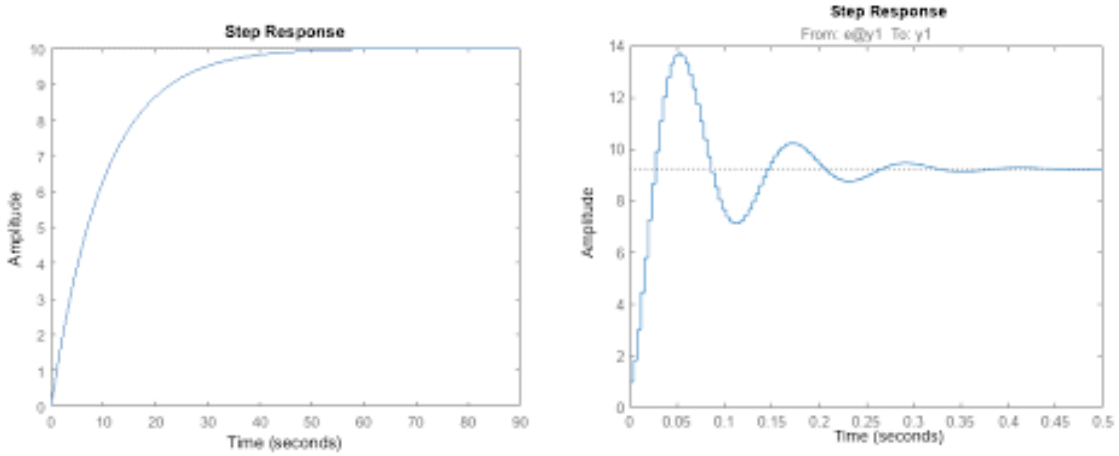
f. Sketch the frequency response of the system using the straight-line approximation method. [10 marks]

g. With regard to characteristics and behaviour of the system, is there any difference between results of time response and frequency response analysis of the system? For the given system, state which method predicts the system more accurately. [2 marks]

B. Analysis of Unstable System

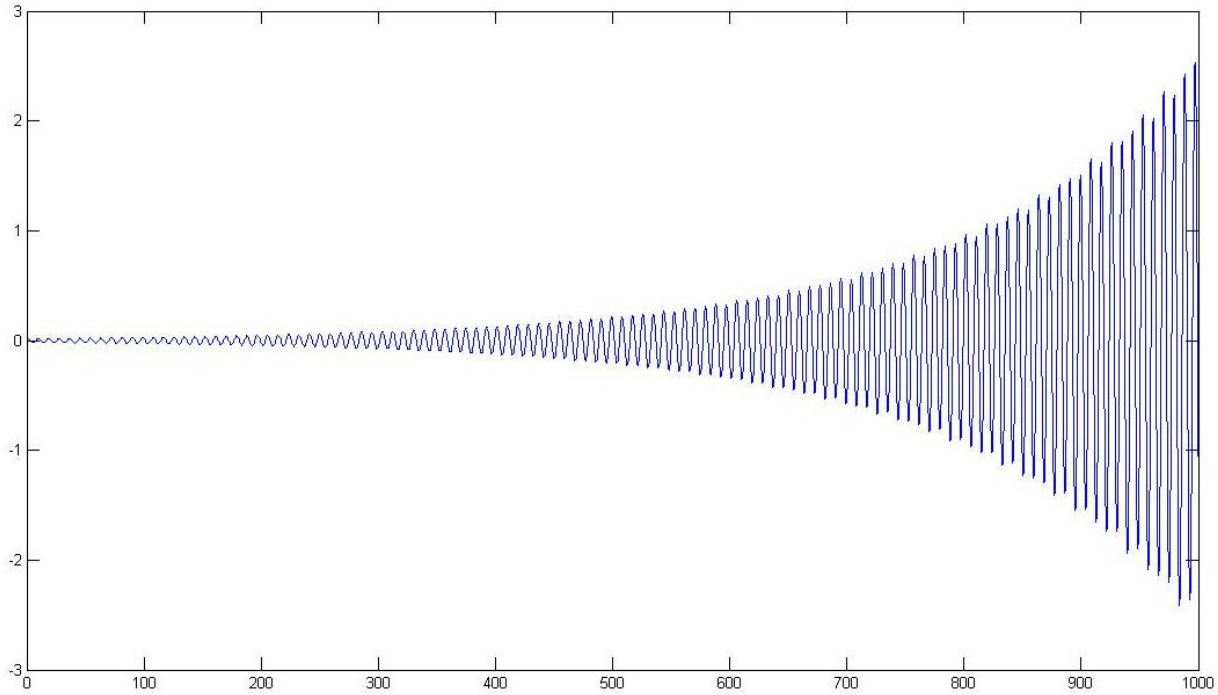
A system is said to be stable, if its output is under control. Otherwise, it is said to be unstable. A stable system produces a bounded output for a given bounded input. For a stable system, the principle is bounded input bounded output (BIBO).

The following diagram illustrates the transient responses of first-order and second-order systems for unit step input. These responses have the values between 0 and 10 and 0 and 9 respectively. So, from the result of the responses both systems have bounded outputs.



We know that the unit step signal has the value of one for all positive values of t including zero. So, it is bounded input. Therefore, the given first-order and second-order systems are stable since both the input and the output are bounded.

The following diagram illustrates an unstable system as its output is growing oscillation. In this system, the output is becoming unbounded even if it is given a bounded input.



For simulation of the system in the frequency domain, use bode( ) function to plot the frequency response of the system. This function conveniently produces magnitude and phase plots based on the transfer function equation of the system given to the function. See Appendix for further details of this function.

Practical Exercise 2 – Simulation and Analysis of Unstable Systems

Knowing that a continuous-time LTI system has the following transfer function equation:

$$\frac{d^2y(t)}{dt^2} - 2\frac{dy(t)}{dt} + 5y(t) = x(t)$$

a. Determine the time domain transfer-function equation of the system. Using MATLAB, verify the transfer function equation by finding its time-domain equation using MATLAB and simulate the time domain plot of the system using this equation. [16 marks]

b. Calculate and determine the time-domain parameters of the system e.g. time constant (τ), rise time (T_r), settling time (T_s), time to peak (T_p), percentage overshoot ($\%OS$), damping ratio (ζ) and natural frequency (ω_n) of the system. Then, verify the calculation by simulating the time-domain's parameters of the system and plot of the time response of the system. Is there any difference between the plots obtained in parts (a) with (b)? Explain why? [14 marks]

c. Plot the frequency response in MATLAB and determine the magnitude and phase at the critical frequencies of the frequency response of the system in MATLAB. [8 marks]

d. Sketch the frequency response of the system using the straight-line approximation method. Identify at least two differences between your sketch with the simulation result. [10 marks]

e. Based on the results of calculation and simulation in parts (a) to (c), describe why the time and frequency response characteristics of the system are similar to the response of an unstable system. [2 marks]

Appendix 1 - Marking Rubric

Student ID : _____

Student Name : _____

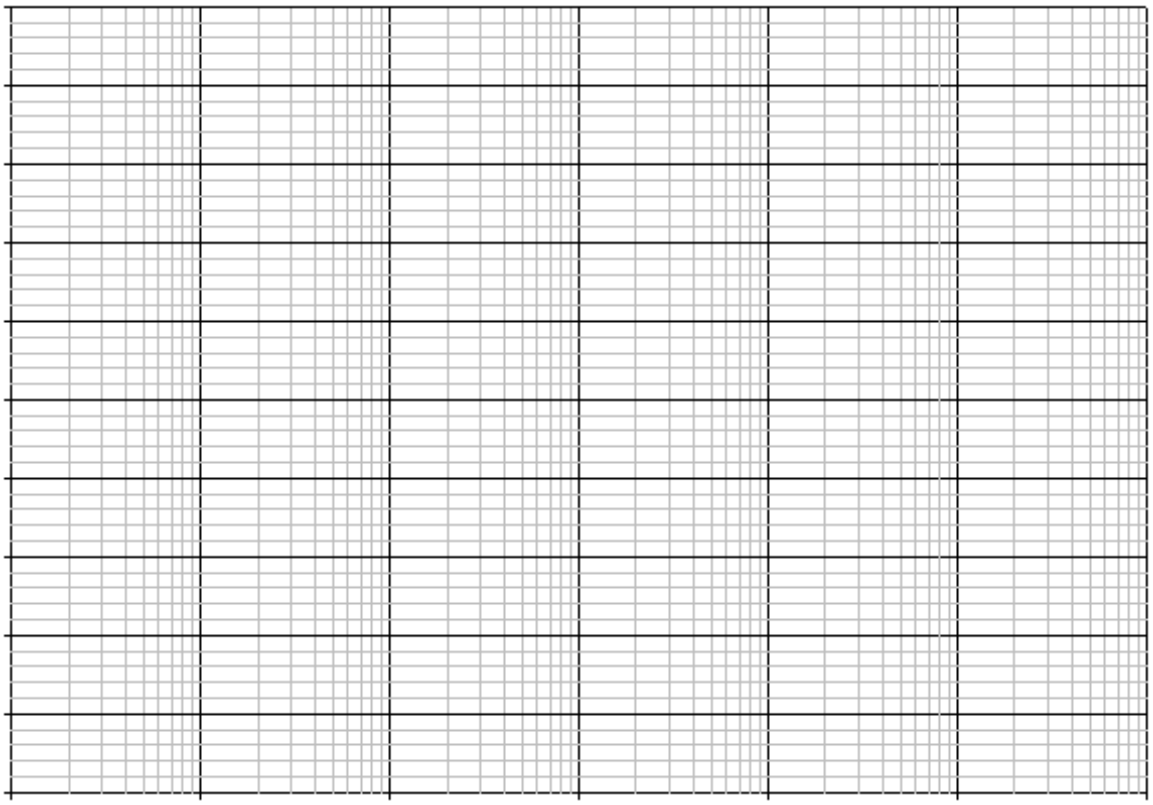
No	Description	Marks	Your Marks	Remarks
1a	Transfer function of the system in term of $G(\omega) = X_o(\omega)/X_i(\omega)$ and its time-domain equation in MATLAB.	4		
1b	Equation for the time response of the system ($x_o(t)/x_i(t)$).	8		
1c	Plot the time response of the system in MATLAB when it is subjected to impulse input. Answers to all questions	8		
1d	Plot the time response of the system in MATLAB when subjected to step input. Two reasons why the system is second order.	8		
1e	Sketch the frequency response of the system using the straight-line approximation method.	10		
1f	Plot frequency response of the system in MATLAB. Values of magnitudes and phase at critical frequencies. Explanation for overall frequency response of the system.	10		
1g	Difference between results of analysis of time response and frequency response of the system? Which method is more accurate?	2		
2a	Time domain transfer-function equation of the system. Verify the transfer function equation by simulating it in	16		

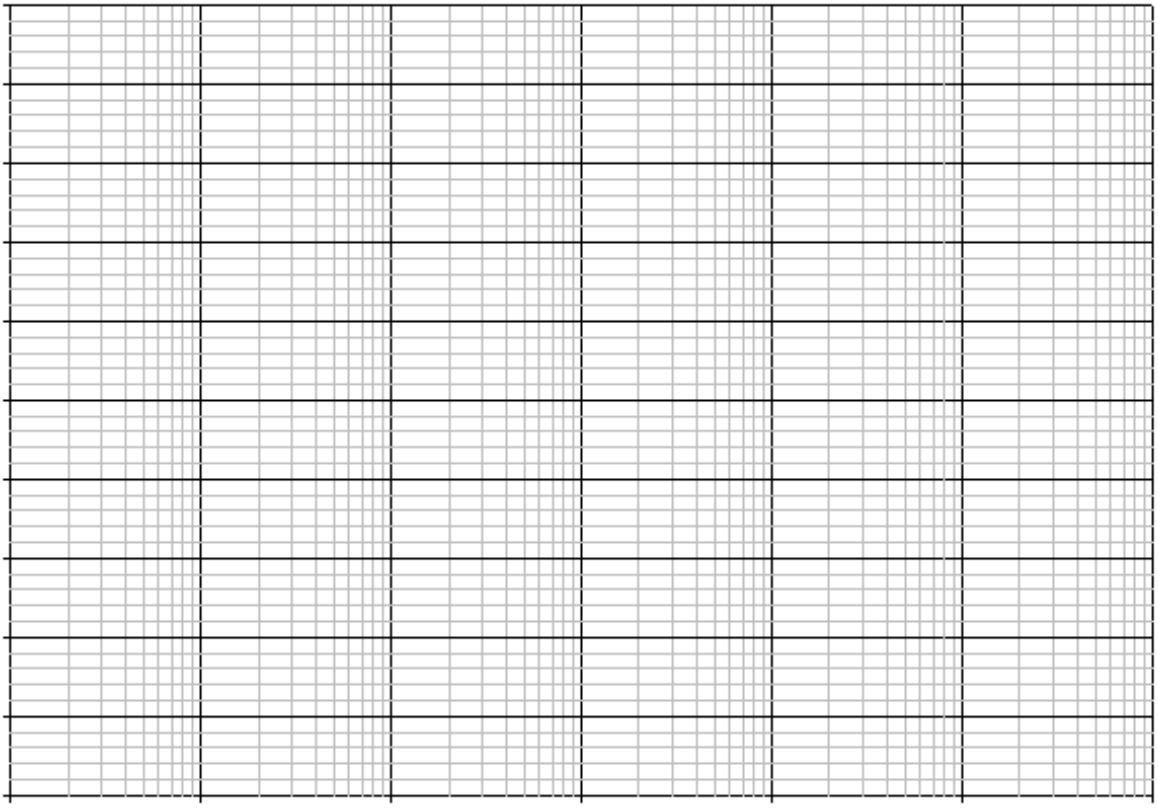
	MATLAB and simulate the time domain plot of the system using the transfer function equation			
2b	The time constant (τ), rise time (T_r), settling time (T_s), time to peak (T_p), percentage overshoot ($\%OS$), damping ratio (ζ) and natural frequency (ω_n) of the system	14		
2c	Plot of the frequency response of the system (both magnitude and phase plots)	8		
2d	Sketch the frequency response of the system using the straight-line approximation method. Two differences between manual sketch and simulation.	10		
2e	Description of frequency response characteristics of the system as in part (c)	2		
	Total	100		

Feedback:

_____
_____
_____
_____
_____

Appendix 1 – Blank Bode Graph





Appendix 2 – Useful MATLAB Functions

`tf()` - Transfer Function function

Syntax

```
sys = tf(numerator,denominator)
```

Description

`sys = tf(numerator,denominator)` creates a continuous-time transfer function model, setting the `Numerator` and `Denominator` properties. For instance, consider a continuous-time single input single output (SISO) dynamic system represented by the transfer function `sys(s) = N(s)/D(s)`, the input arguments `numerator` and `denominator` are the coefficients of `N(s)` and `D(s)`, respectively.

Example

For this example, consider the following SISO transfer function G(s)=2/(2s^2+3s+4)):

Specify the numerator and denominator coefficients ordered in descending powers of `s`, and create the transfer function model.

```
numerator = 1;

denominator = [2,3,4];

sys = tf(numerator,denominator)

sys =


         1
  ---------------
  2 s^2 + 3 s + 4


Continuous-time transfer function.
```

`plot()` - Plot a graph function

Syntax

```
plot(X,Y)

plot(X,Y,LineSpec)
```

Description

`plot(X,Y)` creates a 2-D line plot of the data in `Y` versus the corresponding values in `X`.

- If `X` and `Y` are both vectors, then they must have equal length. The `plot` function plots `Y` versus `X`.
- If `X` and `Y` are both matrices, then they must have equal size. The `plot` function plots columns of `Y` versus columns of `X`.

`plot(X,Y,LineSpec)` sets the line style, marker symbol, and colour.

Examples

Create Line Plot

Create x as a vector of linearly spaced values between 0 and 2π. Use an increment of $\pi/100$ between the values. Create y as sine values of x. Create a line plot of the data.

```
x = 0:pi/100:2*pi;
y = sin(x);
plot(x,y)
```

Plot Multiple Lines

Define x as 100 linearly spaced values between -2π and 2π. Define y1 and y2 as sine and cosine values of x. Create a line plot of both sets of data.

```
x = linspace(-2*pi,2*pi);
y1 = sin(x);
y2 = cos(x);

figure
plot(x,y1,x,y2)
```

`impulse()` - Impulse function

Syntax

```
impulse(sys)
```

Description

`impulse(sys)` plots the impulse response of the dynamic system model `sys`. This model can be continuous or discrete, and SISO or multiple input multiple output (MIMO). The impulse response of multi-input systems is the collection of impulse responses for each input channel. The duration of simulation is determined automatically to display the transient behavior of the response.

Example

```
numerator = 1;
denominator = [2,3,4];
sys = tf(numerator,denominator)
impulse(sys)
```

`step()` - Step function

Syntax

```
step(sys)
```

Description

`step(sys)` plots the response of a dynamic system model to a step input of unit amplitude. The model `sys` can be continuous- or discrete-time, and SISO or MIMO. For MIMO systems, the plot displays the step responses for each I/O channel. `step` automatically determines the time steps and duration of the simulation based on the system dynamics.

Example

Plot the step response of a system with transfer function G(s)=4/(s^2+2s+10).

```
sys = tf(4,[1 2 10]);
step(sys)
```

`bode()` – Plot frequency response

Syntax

```
bode(sys)
```

Description

`bode(sys)` creates a Bode plot of the frequency response of a dynamic system model `sys`. The plot displays the magnitude (in dB) and phase (in degrees) of the system response as a function of frequency. `bode` automatically determines frequencies to plot based on system dynamics.

Example

Create a Bode plot of a continuous-time SISO dynamic system with transfer function G(s)=(2s+1)/(s^2+2).

```
H = tf([2 1],[2 0 2]);
bode(H)
```