**Signals & Systems**

**EEE-223**

Lab # 12



.

|  |  |
| --- | --- |
| Name | Muhammad Haris Irfan |
| Registration Number | FA18-BCE-090 |
| Class | BCE-4A |
| Instructor’s Name | Muhammad Bilal Qasim |

**LAB#12**

**Open ended Lab**

**Lab 12-** **Open ended lab**

**Objectives**

Objective of this lab is to encourage the analytical and problem-solving skills in a methodical way through literature survey, design and conduct of experiments, analysis and interpretation of experimental data and synthesis of information to derive the valid conclusions.

**Pre-Lab**

**12.1 The Laplace Transform:**

The Laplace transform expresses a signal in the complex frequency domain s (or s-domain); that is, a signal is described by a function F(s). Laplace transform is defined by the symbol L{.}; that is, one can write

 12.1

In order words, the Laplace transform of a function f(t) is a function F(s). An alternative way of writing (equation 12.1) is

 12.2

There are two available forms of Laplace transform. The first is two-sided (or bilateral) Laplace transform where the Laplace transform F(s) of a function f(t) is given by

 12.3

The second form is one-sided (or unilateral) Laplace transform, which is described by the relationship

 12.4

In order to return from the s-domain back to the time domain, the inverse Laplace transform is applied. The inverse Laplace transform is defined by the symbol ; that is, one can write

, 12.5

or alternatively

12.6

**Example 1:**

Compute the (unilateral) Laplace transform of the function .

|  |  |  |
| --- | --- | --- |
| Commands | Results | Comments |
| syms t s  t  s  f=exp(-t);  laplace(f,t) | t=t  s=s  ans=1/(1+s) | The command syms is used to define the symbolic variables t and s.  The function is defined as symbolic expression and the (unilateral) Laplace transform is computed. |

**Example 2:**

Compute the inverse Laplace transform of the function .

|  |  |  |
| --- | --- | --- |
| Commands | Results | Comments |
| syms t s  F=1/(1+s);  ilaplace(F,s) | ans=exp(-t) | The command syms is used to define the symbolic variables t and s.  The function is defined as symbolic expression and the inverse Laplace transform is computed. |

**12.2 The *residue* Function**

The Matlab *residue* function converts its argument function, given in rational form, to partial fraction form. The syntax is [R, P, K] = residue (B,A), where B is the vector containing the coefficients of the numerator polynomial and A is the vector of the coefficients of the denominator polynomial. Suppose that a signal X(s) is written in the rational form



By defining the vector  and , and executing the command [R, P, K]=residue(B,A), the signal X(s) can be written in partial fraction form as



Where, ,

and K is the residue of division between the two polynomials, K is null when m<n.

The command residue is illustrated in the following examples.

**Example 3:**

Express the signal transform *X*(*s*) given below in partial fraction form



|  |  |  |
| --- | --- | --- |
| Commands | Results | Comments |
| num = [1 3 1];  den = [1 5 2 -8];  [R,P,K] = residue(num,den)  syms s  X=R(1)/(s-P(1))+R(2)/(s-P(2))+R(3)/(s-P(3))  Pretty(X) | R=0.5000 0.1667 0.3333  P=-4.0000 -2.0000 1.0000  K=[]  The signal X(s) is written as | The signal X(s) is expanded in partial fraction form. Notice that K is empty; that is, there is no residue. |

**Example 4:**

Express the signal transform *X*(*s*) given below in partial fraction form



In this case, the roots of the denominator polynomial are complex numbers. The complex roots do not really change anything in the followed procedure.

|  |  |  |
| --- | --- | --- |
| Commands | Results | Comments |
| num = [1 -3 2];  den = [1 4 5];  [R,P,K] = residue(num,den) | R=-3.50-5.50i -3.50+3.50i  P=-2.00+1.00i -2.00-1.00i  K=1 | The signal X(s) is expressed in partial fraction form as |

**Task 01: Compute the unilateral Laplace transform of the function . Also evaluate the inverse Laplace transform of your result.**

|  |
| --- |
| syms t s  f = -1.25+3.5\*t\*exp(-2\*t)+1.25\*exp(-2\*t);  l = laplace(f,s);  pretty(l);  il = ilaplace(l,t);  pretty(il);  **OUTPUT:**  5 7 5  --------- + ---------- - ---  4 (s + 2) 2(s+2)^2 4 s  exp(-2 t) 5 t exp(-2 t) 7 5  ----------- + ------------- - -  4 2 4 |

**Task 02: Compute the unilateral Laplace transform of the function .**

|  |
| --- |
| syms t s  f = 1;  l = laplace(f,s);  pretty(l);  **OUTPUT:**  1  --  s |

**Task 03: Express in the partial fraction form the signal**

****

|  |
| --- |
| syms s t  num = [ 1 0 -3 2 ];  deno = [ 1 4 5 ];  [R P K] = residue(num,deno)  X = R(1)/(s-P(1)) + R(2)/(s-P(2)) + s\*K(1) + K(2);  pretty(X);  **OUTPUT:**  4 - 3i 4 + 3i  s + --------- + ---------- - 4  s + 2 - i s + 2 + 1i |

**Task 04: Express in the partial fraction form the signal**

****

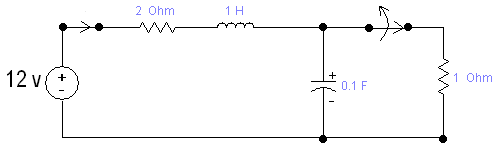
**Verify your result by computing the inverse Laplace transform from both forms (partial fraction and rational) and plot both of the results.**

|  |
| --- |
| syms s t  num = [ 1 5 4 ];  deno = [ 1 0 0 0 1 ];  [R P K] = residue(num,deno);  X = R(1)/(s-P(1)) + R(2)/(s-P(2)) + R(3)/(s-P(3)) + R(4)/(s-P(4)) ;  il = ilaplace(X,t)  ezplot(il);  title('ILT Through Partial Fraction Form');  syms s  figure();  rat = (s^2 + 5\*s + 4)/(s^4 + 1);  il2 = ilaplace(rat,t);  ezplot(il2);  title('ILT Through Rational Form');  **Inverse Laplace through Partial Fraction**  **Chart  Description automatically generated**  **Inverse Laplace through Rational Form**  Graphical user interface, chart  Description automatically generated |

**In-Lab Open ended Tasks**

**Task 01: Examine the network shown in figure 12.1 below. Assume the network is in steady state prior to.**

1. **Plot the output current for**
2. **Determine whether the system is stable or not?**

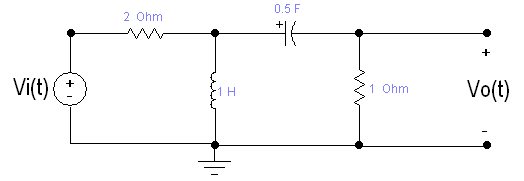
****

**Figure 12.1**

|  |
| --- |
| Diagram  Description automatically generated  **CODE:**  syms t s  I=(4\*s-1)/(2+s+10/s);  R = (2+s+10/s);  IL=ilaplace(I,t);  ezplot(IL,[-0.5 10]);  title('FA18-BCE-090');  ylim([-3 4]);  xlabel('t');  ylabel('I ( t ) ');  E=int(abs(I\*R).^2,s,0,inf)  Graphical user interface, chart, line chart  Description automatically generated **BIBO Unstable**  E=int(abs(V).^2,s,-inf,inf)  E **=**    Inf |

**Task 02: For the circuit shown in figure 12.2, the input voltage is**

1. **Plot the steady state output voltage for assuming zero initial conditions.**
2. **Determine whether the system is stable or not?**

****

**Figure 12.2**

|  |
| --- |
| **Diagram  Description automatically generated**  **CODE:**  syms t s  Vin=10\*cos(2\*t)\*heaviside(t);  lap\_Vin=laplace(Vin,s);  output=lap\_Vin\*((s\*(s+2))/(3\*(s^2)+4\*s+4));  Vo=ilaplace(output,t);  ezplot(Vo,[0 10]);  title('FA18-BCE-090');  xlabel('t');  ylabel('V ( t )');  e = int(abs(output).^2,s,-inf,inf)  **Graphical user interface, chart, line chart  Description automatically generated**  **BIBO Stable**  e = int(abs(output).^2,s,-inf,inf)  e =    (25\*pi\*2^(1/2))/16 |

## **Critical Analysis / Conclusion**

In this lab we learned about Laplace transformation as well as Inverse Laplace transformation. We used “laplace” and “ilaplace” commands in MATLAB to convert from time (t) domain to Laplace (s) domain or from Laplace (s) domain to time (t) domain, respectively. These commands were applied in RLC circuits as well. We also learned the residue function which converts a given rational function into partial fraction.