Using MatLab to solve Problems in Electrical Engineering Courses

Klaus Wuersig

University of Pittsburgh at Bradford,

Email: wuersig@pitt.edu

Abstract

Students are introduced to MatLab in their second semester at the University. Many problems are solved in all areas of Engineering and Mathematics. The ease of writing a program and executing it makes MatLab a perfect contender for solving Electrical Engineering problems in the Sophomore year and beyond. Mesh and Nodal analysis in Linear Circuits I and II can be greatly simplified with just a few lines of code and answers are quickly obtained with left array and right array and left division. Especially in Linear Circuits II with complex number manipulation any order mesh or nodal problem is solved in a few minutes and the same ease of operation applies to LaPlace transform problems. In Design of Electronic circuits Diodes, MOSFETs and BJT's can be modeled and various outputs can be obtained for different inputs.

Keywords: Electrical Engineering; MatLab; Mesh and Nodal Analysis; Diode Modeling.

1. Introduction

In the second semester at the University of Pittsburgh the powerful programming language of MatLab is introduced. In the sophomore year many of the programs learned previously are used to solve intricate problems in the various Engineering disciplines. This applies especially to courses in the Electrical Engineering course area. Recognizing that MatLab is used to solve and model Engineering problems in hundreds of companies and universities worldwide, it is most appropriate to make students aware how this language can solve intricate problems in Linear Circuits and Systems I , in Linear Circuits and Systems II and in Design of Electronic Circuits quickly and with a minimum of effort.

Students in Linear Circuits and Systems I, can be started to solve for total series or parallel Resistance with a simple program. However this program is easily modified so it can solve for any number of Impedances in series or in parallel, which of course makes it very useful for Linear Circuits and Systems II. A simple set up for mesh and nodal analysis with MatLab makes it possible to solve any order of these problems. By concentrating on how to set up the equations based on Kirchoffs Laws, instead of going over the painful methods of using determinants for solving loop currents or node voltages, gives the instructor more time to teach basic circuit theory. Since Linear Circuits and Systems I is also a requirement for Mechanical Engineering majors it makes even more sense to use Matlab programming and solutions to solve Mesh and Nodal setups because these students are more interested in the solutions than in the theory behind it.

In Linear Circuits and Systems II, it is a simple adjustment to change the previously learned procedures and setups to include complex numbers. What used to be an arduous process for any mesh or nodal setup that was just a 3*3, can now be done for any order of these setups. Both rectangular and polar answers are quickly obtained and the polar answer includes the angle. Students race each other to find out who is the quickest to get the right answer. In three phase systems with complex loads the mesh analysis method provides answers for phase and or line currents very quickly and gives students the added satisfaction of understanding where some of the multipliers for line to line voltages and phase and line currents come from. When one looks at transfer characteristics using s parameters, with a few keystrokes Bode plots of gain in dB versus frequency and phase versus frequency can be obtained. The response to a unit step function for any transfer function in terms of s parameters is obtained with a minimum amount of effort. For LaPlace transforms using partial fraction expansion solutions can be found using the residue method. In Design of Electronic Circuits, output of diodes can be modeled for numerous inputs and adjustments in the output of a diode can be obtained with various diode drops. Students always have problems when it comes to determine VGS in a MOSFET that includes a Source Resistor. The set up here includes a KVL loop and the quadratic equation.

2. Examples and Applications

The Examples and Applications shown next will deal with the use of MatLab in Linear Circuits 1, and will be restricted to the applications of series and parallel Resistors and Mesh and Nodal Analysis.

2.1 Simple Series and Parallel Resistance Examples

The Program

```
Rinput=input('How many Resistors[R1,R2,---,Rn]='); Rt=input('Choice of Rs or Rp?[Rs]:','s'); if Rt=='Rs'; Rs=sum(Rinput)
else Rp=1./Rinput; Rpstep1=sum(Rp) Rpfinal=1./Rpstep1 end
```

The Answer for Parallel Resistance

```
How many Resistors[R1,R2,---,Rn]=[10,12.6,40]   Choice of Rs or Rp?[Rs]:Rp     Rpfinal =   4.8932
```

Actually using the same program for complex numbers will generate complex results.

```
How many Resistors[R1,R2,---,Rn]=[10+10i ,5-5i]   Choice of Rs or Rp?[Rs]:Rs     Rs =  15.0000 + 5.0000i
How many Resistors[R1,R2,---,Rn]= [10+10i, 5-5i]   Choice of Rs or Rp?[Rs]:Rp    Rpfinal = 6.0000 - 2.0000i
```

2.2 Mesh and Nodal Examples

The Program

```
% Using the Format Method of writing complex Mesh equations
% (4-j5)I1 - (1+j0)I2 = (5-j5)
% -(1+j0)I1 + (3+j3)I2 = (-10+j5)
Amatrix = [(4-5i),-(1-0i);-(1+0i),(3+3i)];Bvector=[(5-5i);(-10+5i)];
LoopCurrents=Amatrix\Bvector
% Converting from Rectangular to Polar Values
PolarMagnitudes=abs(LoopCurrents)
AngleRadians=angle(LoopCurrents)
DegreeAngle1=180*(.3599)/pi
DegreeAngle2=180*(1.8383)/pi
```

The Answers

```
LoopCurrents =    0.7372 + 0.2774i   ,   -0.6642 + 2.4234i     AngleRadians =   0.3599 ,  1.8383
PolarMagnitudes =   0.7877  ,  2.5127    DegreeAngle1 =  20.6208    DegreeAngle2 =    105.3268
```

For brevities sake a simple Nodal example will be given.

The Program

```
% (1/4+1/1)V1-(1/1)V2=2   %-(1/1)V1 + (1/2+1/1)V2=-1   A=[1.25,-1;-1,1.5];    B=[2;-1];   Result=A\B
```

The Answer

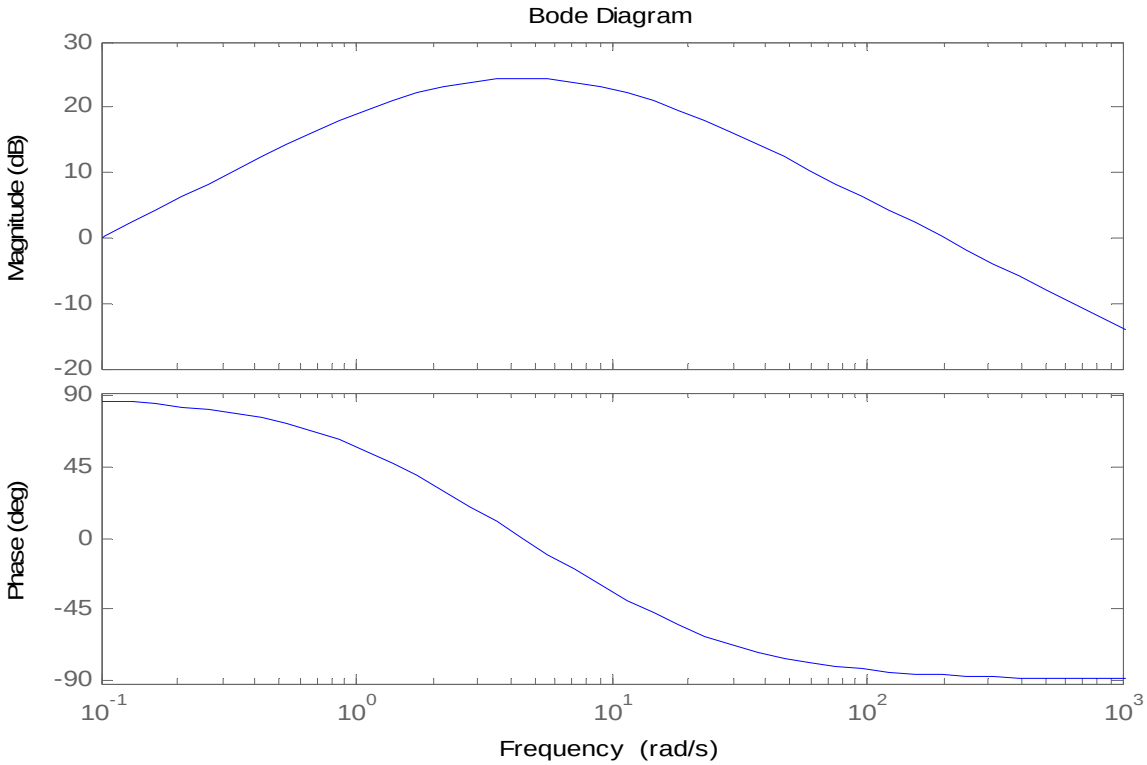
```
Result = 2.2857, 0.8571
```

2.3 Examples for Bode Plots, Responses to Step functions and exponentials and Partial Fraction Expansion

The Bode Program

```
% The given Transfer function   % H(s)=200s/(s^2+12s+20)    num=[200 0];   den=[1 12 20];   bode(num,den);
```

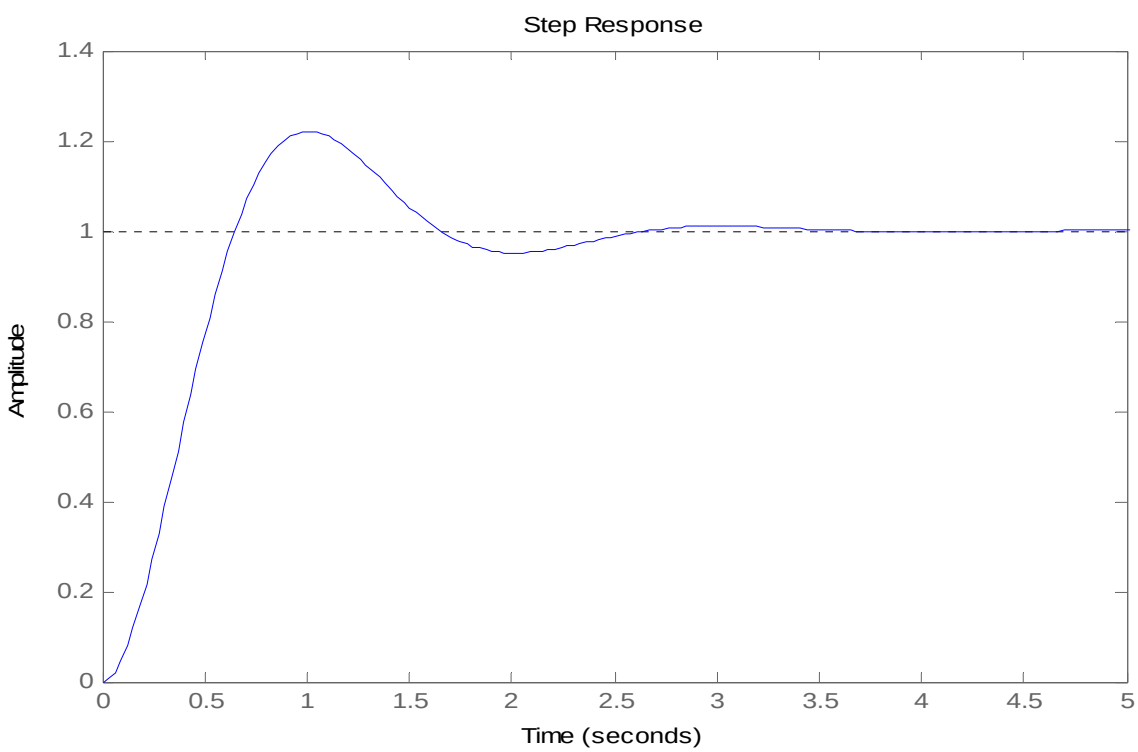
The Gain and Phase Plots



The Step Response Program

```
% The given Transfer function%    % H(s)=12/(s^2+3s+12)     n=12;     d=[1,3,12];     step(n,d);
```

The Step Response Plot

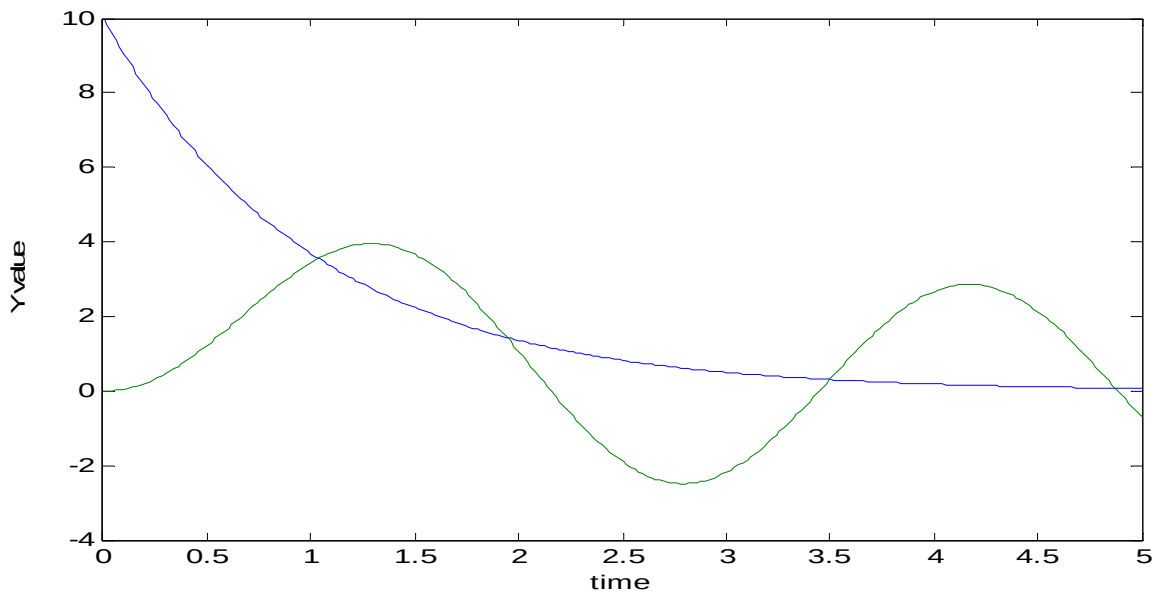


The Exponential Response Program

% The given Transfer function% % H(s)=s+4/(s^3+2s^2+5s+10) t=[0:.02:5]; x=10*exp(-t); num=[1 4]; den=[1 2 5 10];

y=lsim(num,den,x,t); plot(t,x,t,y)

The Exponential Response Plot



The Partial Fraction Expansion Program

%Partial Fraction Expansion disp=('F(s)= (s+1)/s(s+2)') n=[1 1]; d=[1 2 0] [r,p,k]=residue(n,d)

The Answers

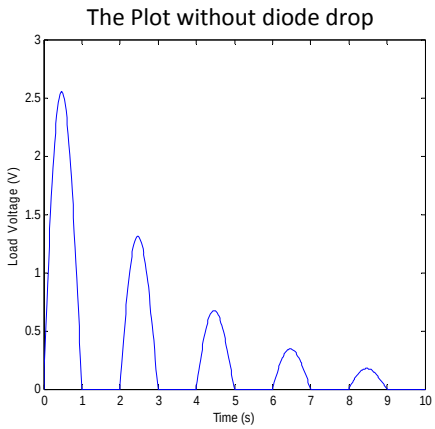
disp = F(s)= (s+1)/s(s+2) r = 0.5000 0.5000 p =-2 0 k = []

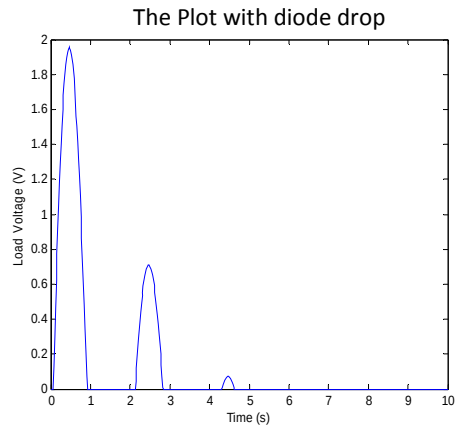
2.4 Examples for Design of Electronic Circuits

In this particular example a decaying exponential waveform and a sine wave are combined. When a forward drop across the diode of .6v is included some of the waveform is eliminated. By eliminating the exponential waveform a typical half wave rectifier circuit can be demonstrated.

The Program

```
t = linspace(0,10,500);  for k = 1:500;  vs = 3*exp(-t(k)/3)*sin(pi*t(k)); if
vs >0
vL(k) = vs;  else  vL(k) = 0;  end end
plot(t,vL),ylabel('Load Voltage (V)'),xlabel('Time (s)')
```





Calculating Vgs in a Mosfet circuit that has an Rs resistor, from which the Q point and IC can easily be calculated.

The Program

```
% Finding Vgs in a MOSFET circuit % IcQ = (VR2-Vgs)/Rs=Kn(Vgs-Vtn)^2
% Rs*Kn(Vgs^2-2VgsVtn+Vtn^2)+Vgs-VR2  Rs=input('Rs =  ');  Kn=input('Kn =
');
VR2=input('VR2 =  '); Vtn=input('Vtn =  ');  a=Rs*Kn; b=Rs*Kn*Vtn*(-2)+1;
c=Vtn*Vtn-VR2; Vgs=(-b+sqrt(b^2-4*a*c))/2*a
```

The Answer for typical given values

```
Rs =  1000   Kn =  .001   VR2 =  2  Vtn =  .8

Vgs = 1.5042
```

3. Conclusions

As can be seen from the typical examples given in the preceding sections, MatLab can play a significant role in numerous Electrical Engineering Courses to arrive at answers to sometimes difficult problems in a minimum amount of time. Since MatLab is used in thousands of companies around the world it not only gives students during the courses some quick answers and tutorials, but it prepares them to actually use a language that is a major component to actually solve Engineering problems. This will enable students to fit into the work environment very quickly. It of course is essential that the answers obtained are completely understood and in Tests and Quizzes that should be a priority.

Students really liked the approach that was presented in this paper and in the coming year more and more programs will be developed by this Professor. It would also be appropriate for on line courses.

Acknowledgements

I would like to thank my wife Celeste and my daughter Dr. Renate Wuersig to look over the manuscript and make many helpful suggestions. Thanks should also go to the Faculty Development Committee for providing some funding to attend the conference.

References

MatLab is a trademark of the MathWorks Inc.
William J. Palm, A Concise Introduction to MatLab, 1st Edition, McGraw-Hill
Charles Alexander, Matthew N. O. Sadiku, Fundamentals of Electric Circuits, 5th Edition, McGraw-Hill
Neamen, Micro Electronics Circuit Analysis and Design, 4th Edition, McGraw-Hill
Leonard S. Bobrow, Elementary Linear Circuit Analysis, 2nd Edition, Oxford Press
Klaus Wuersig, Internal Documents, University of Pittsburgh at Bradford
Klaus Wuersig, Selection of Appropriate Programming Languages for Engineering Applications, ICEE 2012