Application of MATLAB in signal and system

Qiuhui Qu*

Shandong Xiehe University, Yaoqiang Town, Licheng District, Jinan City, China

Abstract. Signal and system course is an important professional basic course for electronic information and communication majors. Signal and system are abstract concepts, which are described by mathematical models. In daily life, simple signals can be calculated or drawn manually, but complex signals are difficult to be accurately processed. Matlab contains graphics processing and symbol operation functions, which provides us with powerful tools to solve the above problems. This paper will introduce how to use Matlab to express, calculate and process signals, and realize the systematic analysis of signals.

1 Introduction

Signal and system is an important professional basic course for electronic information majors. This course involves many basic analysis methods. Its characteristic is that theoretical calculation is cumbersome, which requires a lot of time to be spent in manual calculation and drawing in the process of learning, resulting in a vague understanding of some basic concepts. The emergence of MATLAB brings convenience to the study of signal and system courses, and frees people from the previous mechanical problems. Its application in signal and system analysis includes the following aspects: time domain analysis of signals using MATLAB, time domain analysis of systems using MATLAB, Matlab implementation of frequency domain analysis of periodic signals, Matlab implementation of frequency domain analysis of aperiodic signals, frequency domain analysis of systems using MATLAB, complex frequency domain analysis of continuous systems using Matlab, z-domain analysis of discrete systems using MATLAB Use Matlab to analyze the modulation and demodulation of the signal.

2 Introduction to matlab

Matlab is the abbreviation of matrix laboratory. MATLAB is a set of high-performance numerical calculation and visualization software launched by MathWorks company in 1984. It integrates numerical analysis, matrix operation, signal processing and graphic display. It can be easily applied to mathematical calculation, algorithm development, data acquisition, system modeling and simulation, data branch and visualization, scientific and engineering drawing Application software development, etc. With its powerful data processing ability and rich toolbox, Matlab makes its programming extremely simple, which can greatly shorten the application development cycle and improve the programming efficiency. At

* Corresponding author: 465736822 @qq.com

present, MATLAB has developed into the most popular and widely used scientific and engineering computing software in the world. It is widely used in automatic control, mathematical operation, image signal processing and other industries. It is also an important tool for teaching and research in universities and research departments at home and abroad, It is internationally recognized as the best technology application software by IEEE.

Its main features are:

(1) Efficient numerical calculation and symbolic calculation functions can free users from complicated mathematical operation and analysis;

(2) It has complete graphics processing function to realize the visualization of calculation results and programming;

(3) Friendly user interface and natural language close to mathematical expressions make it easy for scholars to learn and master;

(4) It has application toolboxes such as signal processing toolbox and communication toolbox with rich functions, providing users with a large number of convenient and practical processing tools.

Matlab's powerful numerical analysis and visualization of calculation results, as well as the auxiliary teaching experiment of the function rich toolbox "signals and systems" provide strong support, which can easily realize the visualization teaching of basic theories and conclusions in teaching.

3 Application examples

Matlab's powerful graphic processing function and symbol operation function provide us with a powerful tool to realize signal visualization and system analysis. Matlab's powerful toolbox function can analyze continuous signals and connections Continuous system can also analyze discrete signals and discrete systems.

In the process of signal and system teaching, Matlab digital simulation of signal and system analysis cases can be added synchronously with the theory, which can change the boring pure principle teaching into vivid teaching closely combined with the actual signal system analysis cases, so that students can broaden their ideas and horizons and improve the teaching effect.

The following is a few specific teaching examples to illustrate that in the teaching of signal and system, the auxiliary teaching experiment using MATLAB word simulation enables students to deepen their understanding of theoretical content in active exploration and creation, and improve students' learning interest and learning efficiency.

3.1 Time domain analysis of signal

To analyze the signal in time domain, we first need to express the law of the signal changing with time with a two-dimensional curve. For simple signals, their waveforms can be drawn manually. But for complex signals, it is very difficult to draw the signal waveform manually, and it is difficult to draw accurate lines. Matlab provides a powerful tool for signal visualization and time domain analysis.

3.1.1 Basic operation of signal

In the process of signal transmission and processing, it is often necessary to calculate a limited number of signals. The basic operations of signals include signal addition, multiplication, translation, inversion and scale transformation.

1. Addition and multiplication of signals

The sum of the two signals is equal to the sum of the instantaneous values of the two signals at any time. The mathematical expression is $f(t)=f_1(t)+f_2(t)$.

The multiplication of two signals is equal to the sum of the instantaneous values of the two signals multiplied at any time. The mathematical expression is $f(t)=f_1(t)f_2(t)$.

2. Inversion and Translation

The inversion of the signal is to replace the independent variable t in f (t) with -t, and the inverted signal is f (-t). From the graph, f (t) and f (-t) are symmetrical about the Y axis, that is, rotate 180 ° with the Y axis as the rotation axis.

The translation of the signal is to replace the independent variable t in f (t) with $t-t_0$. At this time, when $t_0>0$, f (t) moves to the right; When $t_0<0$, f (t) shifts left.

3. Scale transformation

The scaling transformation of the signal is to replace the independent variable t in f (t) with at. The signal after scaling transformation is f (at), which is the signal obtained by stretching or compressing the original signal along the X axis. At that time, f (t) was compressed to times of the original; At that time, f (t) was stretched to times the original;

Example 1 transforms f (x) =sinx into f (2x) and f (0.5x).

Matlab program is as follows:

```
x=-6:0.01:6;
f=sin(x);
g=sin(2*x);
h=sin(0.5*x);
subplot(3,1,1)
plot(x,f)
title('f(x)')
subplot(3,1,2)
plot(x,g)
title('f(2x)')
subplot(3,1,3)
plot(x,h)
title('f(0.5x)')
```

3.1.2 Zero input response and zero state response

1. Zero state response

The Matlab function for finding the zero state response of the differential equation is lsim. Its calling format is:

```
y=lsim(sys,f,t)
```

Among them, t is the independent variable, f is the system input signal, and sys is the system model, which is used to represent the difference equation, differential equation and state equation. Sys should be obtained with the help of TF function. TF is the transfer function model, various system models and mutual conversion functions provided by MATLAB.

Such as differential equation $\frac{d^2y(t)}{dt^2}+3\frac{dy(t)}{dt}+2y(t)=\frac{d^2f(t)}{dt^2}+f(t)$. The corresponding coefficient vector is represented by vectors a and B, and its transfer function is represented by sys.

Enter in the command window:

```
>>a=[1 3 2];
>>b=[1 0 1];
>>sys=tf(b,a)
```

Example 2 the input signal of the known system is f (T) =10sin (2 π T), and the differential equation is:, try to calculate y (T).

Matlab program is as follows:

```
ts=0;te=5;dt=0.01;
sys=tf([1],[1 2 100]);
t=ts:dt:te;
f=10*sin(2*pi*t);
y=lsim(sys,f,t);
plot(t,y);
xlabel(‘Time（sec）’)
ylabel(‘y（t）’)
```

3.1.3 Convolution integral

The principle of convolution integration is to decompose the signal into the sum of several impulse signals, so the response of the system is the linear superposition of the impulse responses corresponding to these impulse signals.

If there are two functions f1(t) and f2(t), their convolution integral is:

$$f_1(t) * f_2(t) = \int_{-\infty}^{+\infty} f_1(\tau) f_2(t-\tau)\, d\tau$$

To realize the convolution of continuous signals f1(t) and f1(t) with MATLAB, it is necessary to sample the two signals to obtain the discrete sequences k1 (n) and k2 (n), and call the discrete convolution summation function conv to calculate the convolution of the two discrete sequences. Its calling format is:

conv(f,g)

This function calculates the convolution of two vectors f and g. If the length of F is m and the length of g is n, the length of convolution is m+n-1.

Example 3: f1(t)=e-tu(t)，f2(t)=sintu(t) are known. Calculate f1(t)*f2(t) and draw the image.

Matlab program is as follows:

```
%Create M file
function f=xconv(f1,k1,f2,k2,d)
f=0.01*conv(f1,f2);
end
% Call m file and draw image
k=0:0.01:5;
k1=0:0.01:5;
k2=0:0.01:5;
d=0.01;
f1=exp(-k1); % Take the discrete value of f1 f2=sin(k2); % Take the discrete value of f1
f=xconv(f1,k1,f2,k2,d);   % Call the function to calculate the convolution of f1 and f2
k0=k1(1)+k2(1); %Calculate initial positionk3=length(f1)+length(f2)-2; % Calculated length
k=k0: d: (k0+k3*d);
subplot (2,2,1)
plot (k1,f1)
title ('f1(t)')
subplot (2,2,2)
plot (k2,f2)
title ('f2(t)')
```

```
subplot (2,2,3)
plot (k,f)
title ('f1(t)*f2(t)')
```

3.2 frequency domain analysis

The Fourier transform analysis method of the system is also called the frequency domain analysis method. The frequency domain analysis method is based on the superposition and uniformity of the linear system. The basic unit of signal decomposition is the constant amplitude sine function. The total response of the system is obtained by calculating the response generated by the excitation of each unit, superimposing the response, and then transferring to the time domain. That is to seek the law of the response with frequency under different signal excitation.

Generally, the calculation of Fourier transform analysis method can be divided into the following steps:

(1) Calculate the Fourier transform of the excitation x (t), that is, F [x (t)] = X (w);

(2) Determine the system function H (w) of the system;

(3) Calculate the Fourier transform of the response Y (w) =H (w) X(W);

(4) Then, Y (W) is inversely transformed from the frequency domain to the time domain, so as to obtain the time function principal y (t) of the zero state response.

Example 4 calculates $f(t)=e^{-3t}u(t)$ the Fourier transform and plots f (t) and its spectrum.

Analysis: the unit step signal u(t) in the title can be expressed by Heaviside (t).

Matlab program is as follows:

```
syms t f
f=exp(-3*t)*sym('Heaviside(t)');
F=fourier(f)
subplot(2,1,1)
ezplot(f)
subplot(2,1,2)
ezplot(abs(F))
```

3.3 Complex frequency domain analysis

Complex frequency domain analysis is to transform the differential equations of continuous systems and the difference equations of discrete systems into algebraic equations in the transform domain, that is, to transform convolution operations into multiplication operations, which makes the operation more simple. Corresponding to differential equations and difference equations, the complex frequency domain analysis is s-domain and z-domain analysis respectively.

3.3.1. Laplace transform

Laplace transform is defined as $F(s)=\int_{-\infty}^{+\infty} f(t)e^{-st}dt$. Where, s= σ+ j ω .It is called complex frequency.

The inverse Laplace transform is defined as $f(t)=\frac{1}{2\pi j}\int_{\sigma-j\infty}^{\sigma+j\infty} F(s)e^{st}dt$.

Matlab gives the sentences for calculating the Laplace transform and its inverse transform of symbolic functions, which are respectively:

Fs=laplace (FT, t, s)% calculate the Laplace transform of the time domain function ft

Ft=ilaplace (FS, s, t)% calculate the inverse Laplace transform of frequency domain function FS

where t is the independent variable of the time domain function and S is the variable of the complex frequency domain function.

Example 5 $f_1(t)=e^{-t}\varepsilon(t)$, $f_2(t)=te^{-1/2t}\varepsilon(t)$ are known. Find its Laplace transform.

Matlab program is as follows:

```
syms t s
f1=exp(-t);
f2=t*exp(-1/2*t);
Fs1=laplace(f1,t,s)
Fs2=laplace(f2,t,s)
```

Program running results:

```
Fs1 =
1/(1+s)
Fs2 =
1/(1/2+s)^2
```

3.3.2. Z transformation

Z transform and Laplace transform have the same operation rules. For discrete-time signals, the z-transform of sequence f (t) is defined as:

$$F(z)=\sum_{k=-\infty}^{+\infty} f(k)\, z^{-k}$$

The inverse transformation is defined as:

$$f(k)=\frac{1}{j2\pi}\oint F(z)\, z^{k-1} dz$$

where Z is a complex variable; F (z) is the image function of sequence f (k); f (k) is the original sequence of F (z).

In MATLAB, the statements for calculating the z-transform and inverse Z-transform of symbolic functions are given as follows:

Fz=ztrans(fk,k,z) % calculate the z-transform of the time-domain function fk

fk=iztrans(Fz,z,k) % calculate the inverse Z transform of the frequency domain function Fz

where k is the independent variable of the time function and Z is the variable of the complex frequency domain function.

Example 6 calculates the z-transform of the symbolic expression f (k) $=k^4$.

Matlab program is as follows:

```
syms k z
f=k^4;
Fz=ztrans(sym('(k^4)'))
Fz=simplify(Fz); % Simplified Fz
pretty(Fz)
```

Example 7 Find the zero and pole of the system function and judge the stability of the system.

Matlab program is as follows:

```
b=[1 0.7];
a=[1 2 2 1];
sys=tf(b,a);
zeros=roots(b)                    % Find zero point
poles=roots(a)                    % Seeking pole
pzmap(sys)
```

Program running results:

```
zeros =
-0.7000
poles =
-1.0000
-0.5000 + 0.8660i
-0.5000 - 0.8660i
```

The system has three poles and one zero. Among them, the three poles are in the left half plane, so the system is stable.

4 Conclusion

Matlab simulation is introduced to expose students to engineering practice. A user graphical interface LTI system simulation and analysis platform of MATLAB is designed, which can realize the realization of typical continuous time signals, the basic operation of signals, the output waveform of system zero state response, the output waveform of impulse response, etc. The application of this platform in the teaching of signals and systems can effectively improve the teaching efficiency, help students understand abstract definitions, and stimulate students' interest in learning. Introducing Matlab into the teaching of signal and system course can visualize the course content, deepen students' understanding and stimulate their enthusiasm for learning. Programming with MATLAB is relatively simple, which greatly improves the programming efficiency and makes the running image clear and intuitive. And the program compilation and execution speed is far faster than the traditional high-level language. Using Simulink to build the system model, simulate and debug the system is not only of great significance in theory, but also very useful in engineering. In short, applying MATLAB to signals and systems can achieve twice the result with half the effort.

References

1. EdwardW.Kamen Bonnie S.Heck Fundamental of Signals and Systems Using the Web and Matlab[M]. Second Edition.Beijing: Science Press(2022)
2. Houjin Chen , signal and system [M] Beijing: Tsinghua University Press(2017)
3. Chen Huaichen, Wu Dazheng, Gao Xiquan MATLAB and its application in electronic information course [M] Beijing: Electronic Industry Press(2018)
4. Wang Jieli, Jia Sumei, Xue Fang application of MATLAB software in signal and system aided teaching [j] Modern electronic technology(2021)