**第四章Matlab作业报告**

2015010912014 耿瑞旭

[M4\_1: 1](#_Toc481441889)

[1. 原型 1](#_Toc481441890)

[2. 长度为5 2](#_Toc481441891)

[3. 长度为7 2](#_Toc481441892)

[4. 长度为9 3](#_Toc481441893)

[5. 利用GUI手动输入长度 3](#_Toc481441894)

[6. 结论 7](#_Toc481441895)

[M4\_2 7](#_Toc481441896)

[1. 实现思路 7](#_Toc481441897)

[2. 代码实现 7](#_Toc481441898)

[3. 输出结果： 8](#_Toc481441899)

[4. 实验结论 8](#_Toc481441900)

[M4\_3 8](#_Toc481441901)

[1. 设计思路 8](#_Toc481441902)

[2. 代码实现 8](#_Toc481441903)

[3. 实验结果 9](#_Toc481441904)

[M4\_5 9](#_Toc481441905)

[1. 设计思路 9](#_Toc481441906)

[2. 设计代码 9](#_Toc481441907)

[3. 运行结果 9](#_Toc481441908)

[4. 结果分析 10](#_Toc481441909)

[附录1 10](#_Toc481441910)

[M4\_1修改后代码 10](#_Toc481441911)

[M4\_1 GUI代码 11](#_Toc481441912)

M4\_1:

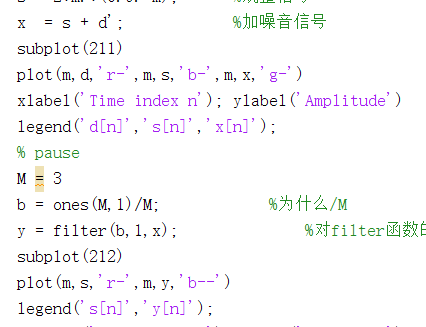
1. 原型

原程序4\_1运行结果如下所示：

（这里对源程序进行了细微的改动：利用subplot分区域显示，取消了pause，并加入了注释，修改后代码见附录1）



1. 长度为5



只需将上图中的M = 3，改为M=5即可，运行结果如下图所示



1. 长度为7

同理，将M改为7，结果如下：



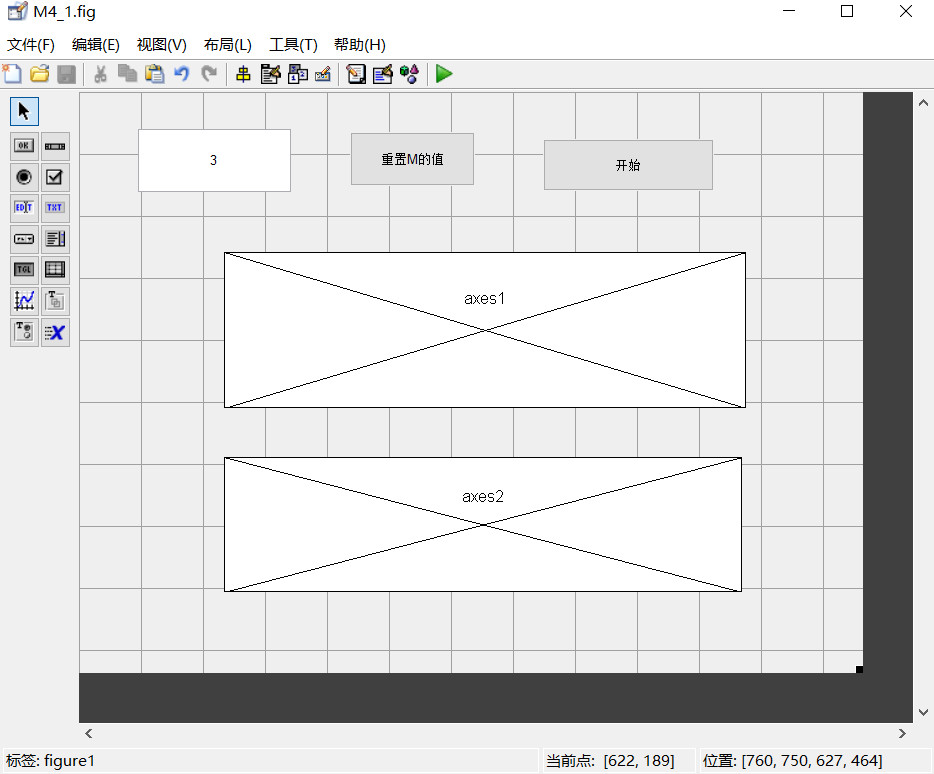
1. 长度为9



1. 利用GUI手动输入长度

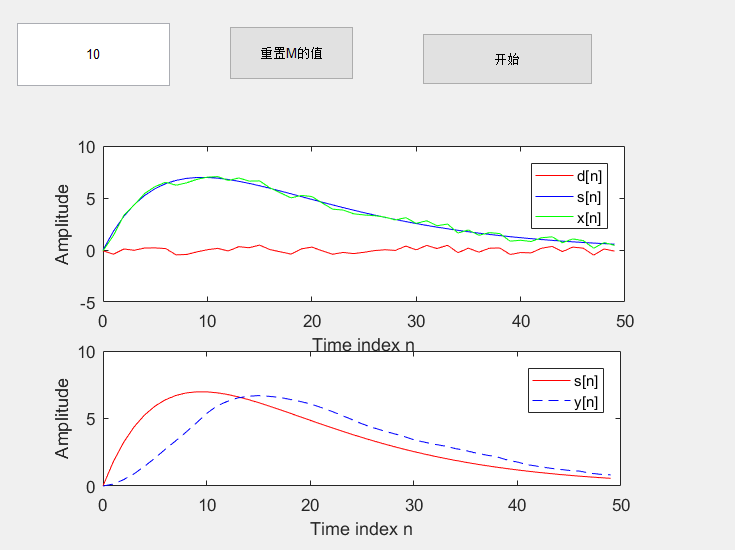
制作GUI

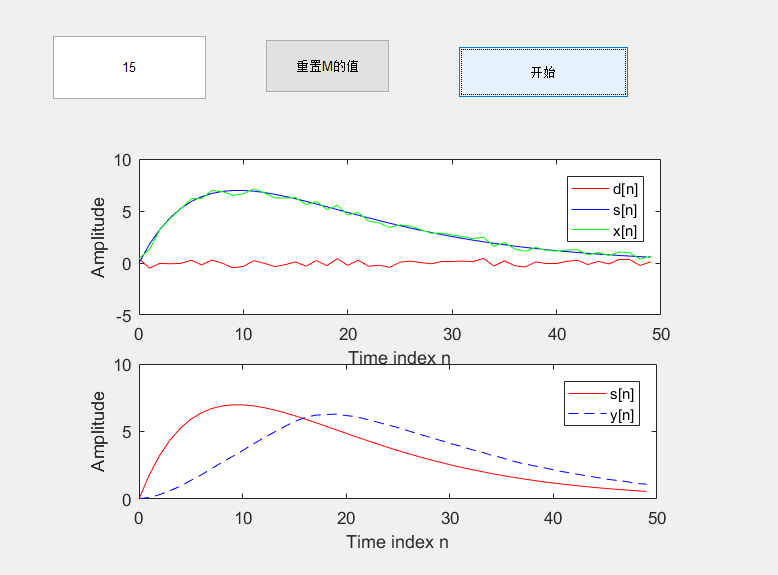
设置如下界面

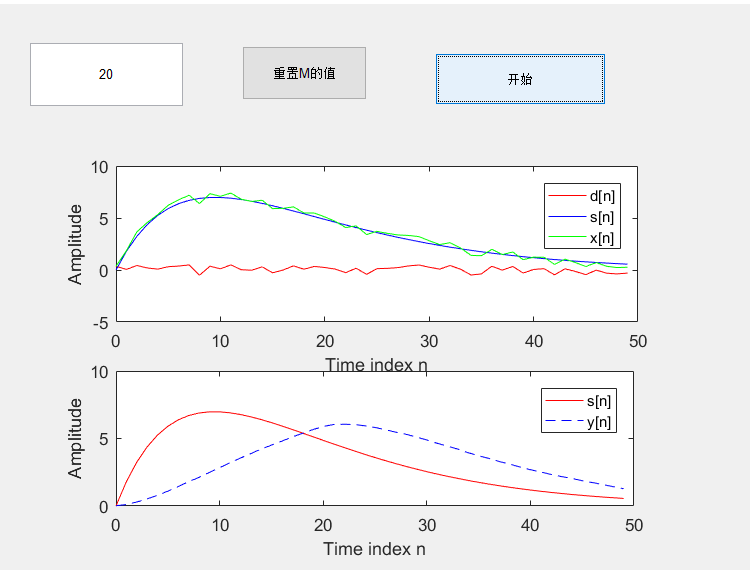


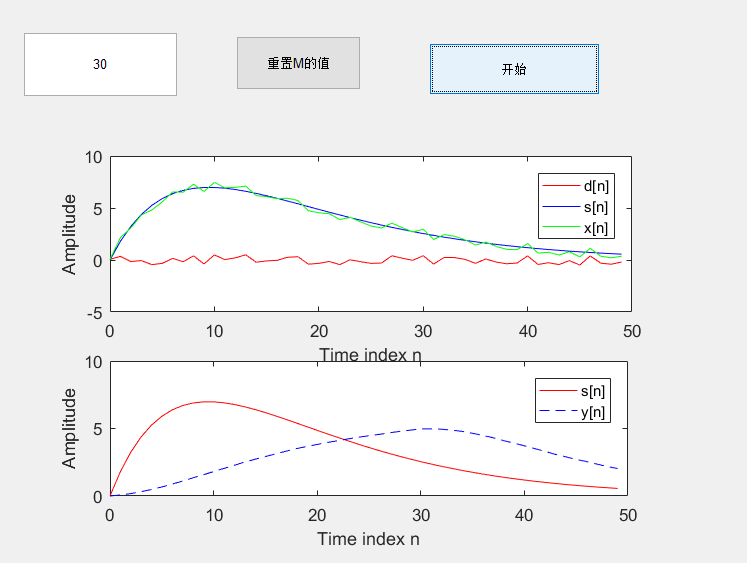
具体的回掉函数请见附录1

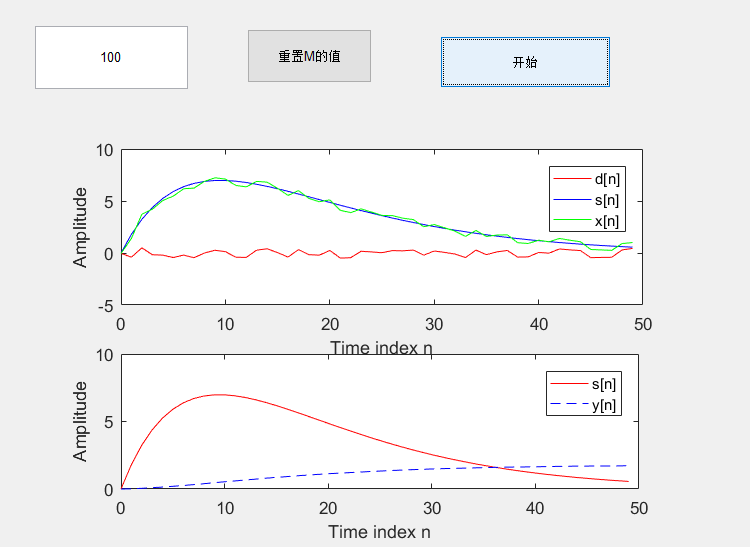
运行结果取值截图如下：











1. 结论

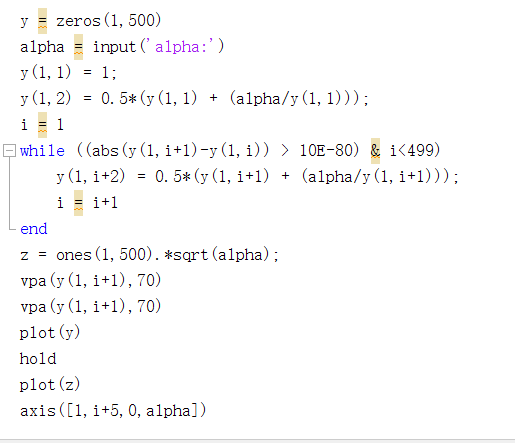
可见当滤波器的长度增加时，信号的平滑效果会得到提升，但是，延迟却会增加。

M4\_2

1. 实现思路

程序大致分为以下几部分：

1. 输入参数alpha，确定初始条件
2. 通过循环，求出y序列的值
3. 检验输出函数是否完成了逐渐逼近平方根的功能
4. 代码实现



1. 输出结果：



当去alpha为800时，输出图像如上图所示，可见确实完成了逼近效果，且非常非常接近（E-80数量级）

1. 实验结论

y[n]确实收敛于根号α

M4\_3

1. 设计思路

与M4\_2类似，分为如下几个步骤：

A． 输入参数alpha，确定初始条件

B． 通过循环，求出y序列的值

C． 检验输出函数是否完成了逐渐逼近平方根的功能

D． 绘制误差函数

1. 代码实现

将代码M4\_2中的差分方程稍作改动即可得到如下代码：

y = zeros(1,500)

alpha = input('alpha:')

c = 2.35;

y(1,1) = c + alpha - c\*c;

y(1,2) = y(1,1) + alpha - y(1,1)\*y(1,1);

i = 1

while ((abs(y(1,i+1)-y(1,i)) > 10E-80) & i<499)

y(1,i+2) = y(1,i+1) + alpha - y(1,i+1)\*y(1,i+1);

i = i+1

end

z = ones(1,500).\*sqrt(alpha);

vpa(y(1,i+1),70)

vpa(y(1,i+1),70)

plot(y)

hold

plot(z)

axis([1,i+5,0,alpha])

1. 实验结果

对该题目的处理存在疑问

一旦y[n]的某个值出现负数，后面的值则必然会一个比一个负，并迅速低至-INF

且该递推式对初始值的近似要求过高

下面是一个成功的输入：

α = 1，初始值0.8



M4\_5

1. 设计思路

先确定该滤波器的传递函数，之后利用freqz函数做出它的幅频特性曲线即可

1. 设计代码

coe = [-4.8788 9.5631 -4.8788]

[H,w] = freqz(coe,1,512);

plot(w/pi, abs(H));

xlabel('\omega/\pi')

ylabel('Magnitude')

title('4.69的滤波器')

1. 运行结果



1. 结果分析

由此可见，这是一个高通滤波器

附录1

M4\_1修改后代码

% Program 4\_1

% Signal Smoothing by a Moving-Average Filter

%通过移动平均滤波器进行信号平滑处理

clear all

R = 50;

d = rand(R,1)-0.5; %噪音

m = 0:1:R-1;

s = 2\*m.\*(0.9.^m); %规整信号

x = s + d'; %加噪音信号

subplot(211)

plot(m,d,'r-',m,s,'b-',m,x,'g-')

xlabel('Time index n'); ylabel('Amplitude')

legend('d[n]','s[n]','x[n]');

% pause

M = 3

b = ones(M,1)/M; %为什么/M

y = filter(b,1,x); %对filter函数的理解

subplot(212)

plot(m,s,'r-',m,y,'b--')

legend('s[n]','y[n]');

xlabel ('Time index n');ylabel('Amplitude')

M4\_1 GUI代码

function varargout = M4\_1(varargin)

% M4\_1 MATLAB code for M4\_1.fig

% M4\_1, by itself, creates a new M4\_1 or raises the existing

% singleton\*.

%

% H = M4\_1 returns the handle to a new M4\_1 or the handle to

% the existing singleton\*.

%

% M4\_1('CALLBACK',hObject,eventData,handles,...) calls the local

% function named CALLBACK in M4\_1.M with the given input arguments.

%

% M4\_1('Property','Value',...) creates a new M4\_1 or raises the

% existing singleton\*. Starting from the left, property value pairs are

% applied to the GUI before M4\_1\_OpeningFcn gets called. An

% unrecognized property name or invalid value makes property application

% stop. All inputs are passed to M4\_1\_OpeningFcn via varargin.

%

% \*See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one

% instance to run (singleton)".

%

% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help M4\_1

% Last Modified by GUIDE v2.5 01-May-2017 20:26:58

% Begin initialization code - DO NOT EDIT

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @M4\_1\_OpeningFcn, ...

'gui\_OutputFcn', @M4\_1\_OutputFcn, ...

'gui\_LayoutFcn', [] , ...

'gui\_Callback', []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

% End initialization code - DO NOT EDIT

% --- Executes just before M4\_1 is made visible.

function M4\_1\_OpeningFcn(hObject, eventdata, handles, varargin)

% This function has no output args, see OutputFcn.

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% varargin command line arguments to M4\_1 (see VARARGIN)

global k

k = get(handles.edit1,'String')

k = str2num(k)

% Choose default command line output for M4\_1

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes M4\_1 wait for user response (see UIRESUME)

% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.

function varargout = M4\_1\_OutputFcn(hObject, eventdata, handles)

% varargout cell array for returning output args (see VARARGOUT);

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure

varargout{1} = handles.output;

function edit1\_Callback(hObject, eventdata, handles)

% hObject handle to edit1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text

% str2double(get(hObject,'String')) returns contents of edit1 as a double

% --- Executes during object creation, after setting all properties.

function edit1\_CreateFcn(hObject, eventdata, handles)

% hObject handle to edit1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

% See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

% --- Executes on button press in pushbutton1.

function pushbutton1\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

global k

M = k

R = 50;

d = rand(R,1)-0.5; %噪音

m = 0:1:R-1;

s = 2\*m.\*(0.9.^m); %规整信号

x = s + d'; %加噪音信号

axes(handles.axes1)

plot(m,d,'r-',m,s,'b-',m,x,'g-')

xlabel('Time index n'); ylabel('Amplitude')

legend('d[n]','s[n]','x[n]');

% pause

b = ones(M,1)/M; %为什么/M

y = filter(b,1,x); %对filter函数的理解

axes(handles.axes2)

plot(m,s,'r-',m,y,'b--')

legend('s[n]','y[n]');

xlabel ('Time index n');ylabel('Amplitude')

% --- Executes on button press in pushbutton2.

function pushbutton2\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

global k

k = get(handles.edit1,'String')

k = str2num(k)