

Лабораторная работа
Разработка приложения с графическим
интерфейсом в среде MATLAB

Подготовил А.В.Бруханский

Разработка приложения с графическим интерфейсом в среде MATLAB

Цель работы: Приобретение навыков разработки графического интерфейса пользователя (GUI) средствами MATLAB и программирование реакций на события, связанными с визуальными компонентами окна приложения.

Запуск среды разработки оконного приложения GUIDE (GUI Development Environment) осуществляется командой **guide**.

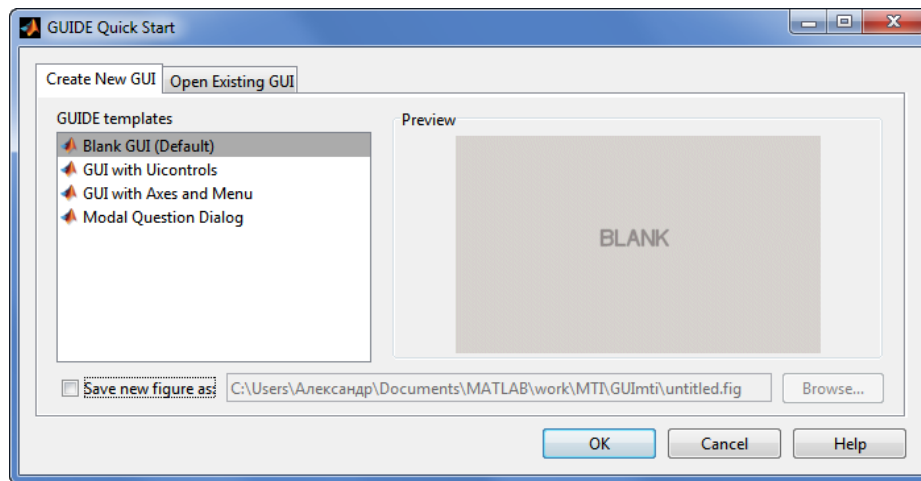


Рисунок 1

Из набора шаблонов (templates) выбираем «Blank GUI (Default)»

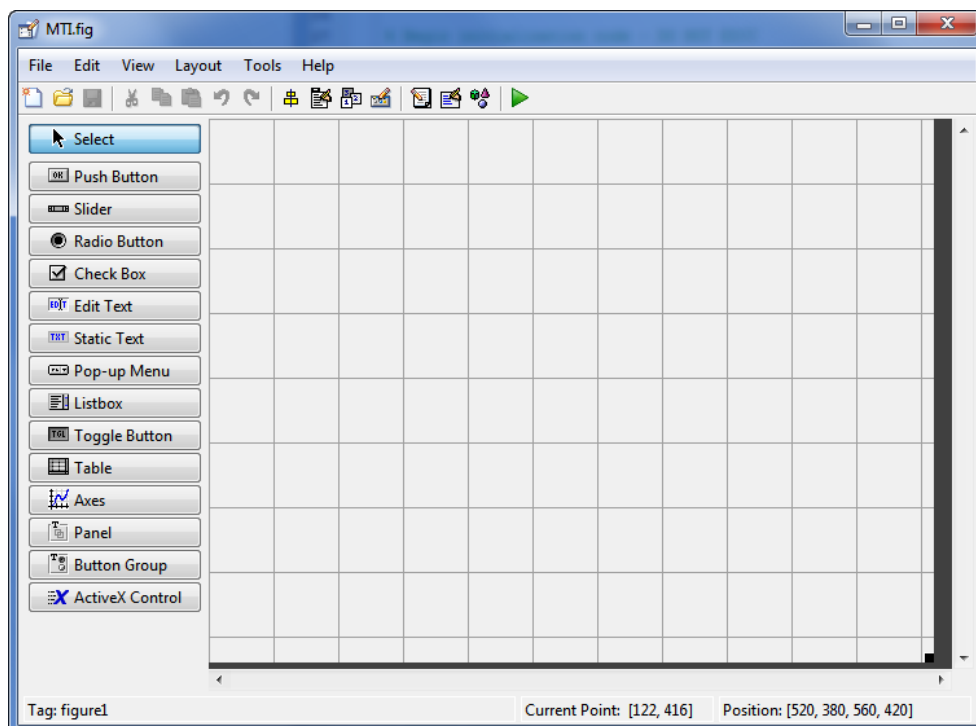


Рисунок 2

После открытия главного окна разработки GUI (рисунок 2) сохраняем пустую форму, например, под именем `MTI.fig`. Одновременно с окном формы MATLAB создает и сохраняет файл с текстом служебных функций, имеющий то же имя, но расширение имени «.m», в данном примере – «`MTI.m`». В этом файле в процессе работы над приложением появятся тексты функций обратного вызова, выполняемые при наступлении каких либо событий в окне работающего приложения. Такими событиями могут быть нажатие на кнопку, установка флажка, изменение теста в окне ввода и т.п.

При необходимости размеры окна формы, как и размеры панели, на которой впоследствии будут размещены элементы управления, можно изменить.

Вид фигуры приложения в редакторе GUI показан на рисунке 3.

Фигура содержит следующие элементы:

`text1`, `text2`, `text3` – статический текст заголовка и меток;

`axes1` – окно графика;

Button Group «Тип режектора» с двумя кнопками `radiobutton1` и `radiobutton2`;

Pop-up Menu выбора цвета линии графика (синий, красный, зеленый);

`edit1` – Edit Text для отображения положения слайдера;

`slider1` – слайдер (ползунок) задания максимального коэффициента корреляции помехи;

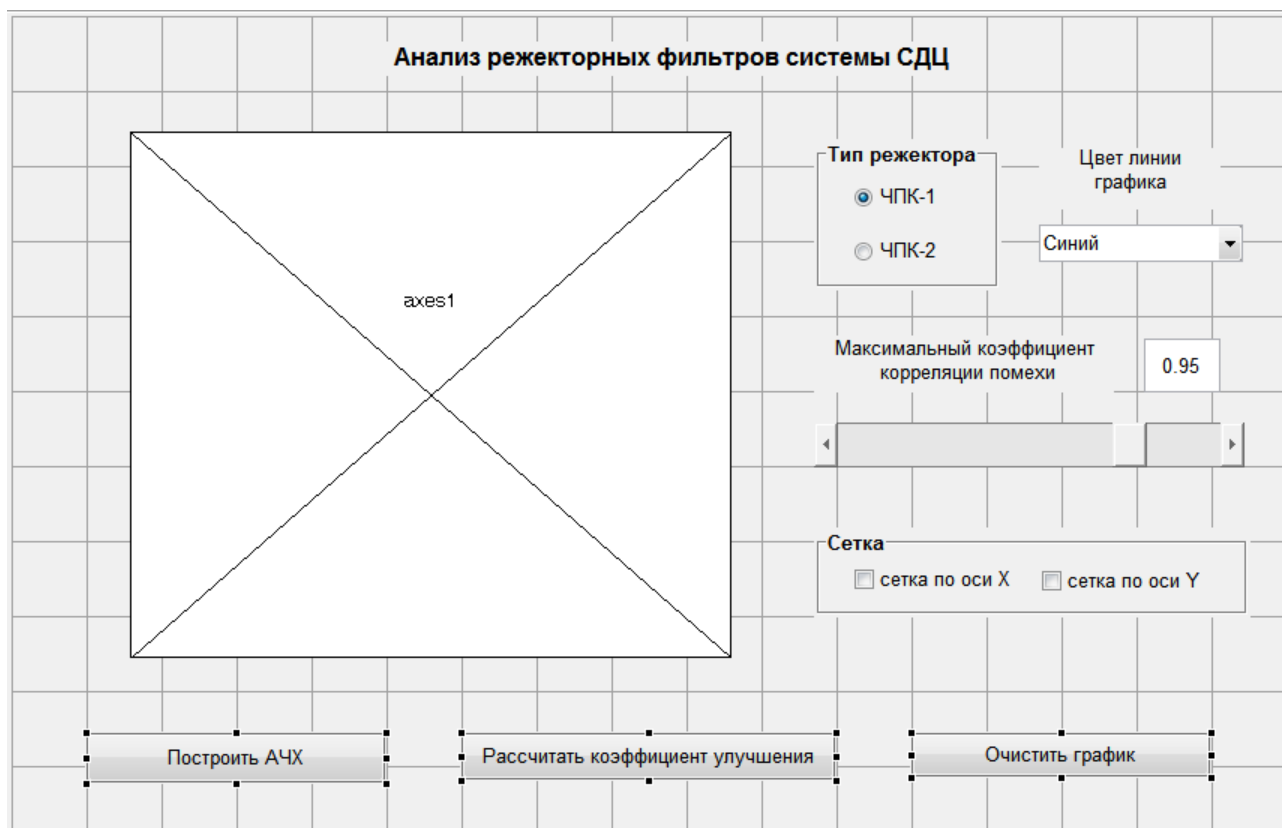


Рисунок 3

Panel – панель «Сетка» с флажками checkbox1 и checkbox2 установки и сброса линий сетки графика, а также неотображаемые на форме главное меню (рисунок 4) и панель инструментов (рисунок 5).

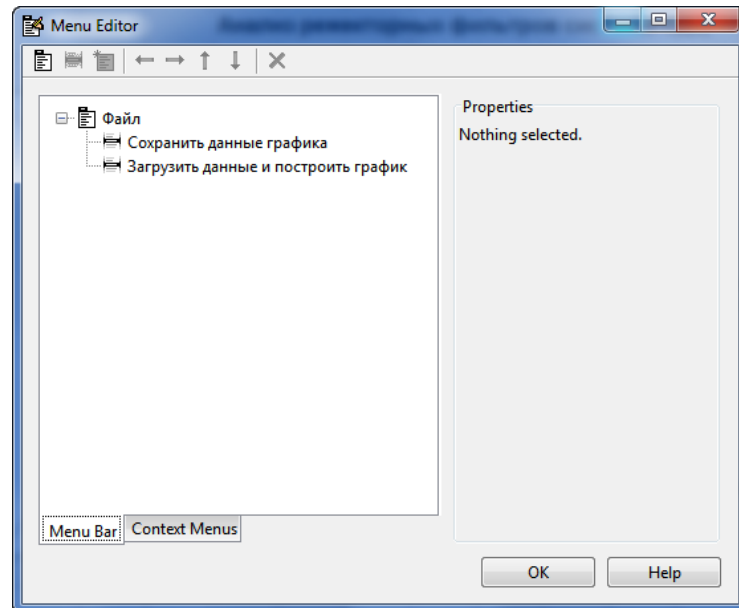


Рисунок 4

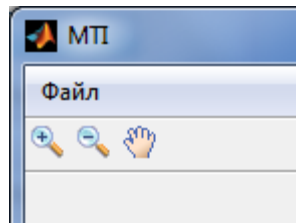


Рисунок 5

Каждому элементу управления окна приложения соответствует функция обратного вызова (callback-function). Исходный код функций обратного вызова приведен далее. В текст программы включены функции, создаваемые средой GUIDE автоматически и не подлежащие изменению.

```
function varargout = MTI(varargin)
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @MTI_OpeningFcn, ...
                  'gui_OutputFcn',  @MTI_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);

if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
```

```

    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% --- Executes just before MTI is made visible.
function MTI_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

title(gca, 'График');

% --- Outputs from this function are returned to the command line.
function varargout = MTI_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
rmax=get(hObject, 'Value');
set(handles.edit1, 'String', num2str(rmax, 2));

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', [.9 .9 .9]);
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
w=0:0.01:2*pi;
if get(handles.radiobutton1, 'Value')
    G=2*sin(w/2);
else
    G=4*sin(w/2).^2;
end
wn=w/2/pi;
hp=plot(wn, G, 'Linewidth', 2);
grid off;
if get(handles.checkbox1, 'Value')
    set(gca, 'XGrid', 'on');
end
if get(handles.checkbox2, 'Value')
    set(gca, 'YGrid', 'on');
end
xlabel('Нормированная частота')
ylabel('Амплитуда')
title('АЧХ режекторного фильтра');
set(handles.pushbutton3, 'Enable', 'on');

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
rm=get(handles.edit1, 'String');
r=0.8:0.01:str2num(rm);
if get(handles.radiobutton1, 'Value')
    Ky=1./(1-r);

```

```

else
    Ky=3./(3-4*r+r.*r);
end
hp=plot(r,Ky, 'Linewidth',2);
grid off;
if get(handles.checkbox1, 'Value')
    set(gca, 'XGrid', 'on');
end
if get(handles.checkbox2, 'Value')
    set(gca, 'YGrid', 'on');
end
xlabel('Коэффициент межпериодной корреляции')
ylabel('Коэффициент улучшения')
title('Коэффициент улучшения режекторного фильтра')
set(handles.pushbutton3, 'Enable', 'on');

function edit1_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

% --- Executes on button press in checkbox1.
function checkbox1_Callback(hObject, eventdata, handles)
if get(hObject, 'Value')
    set(gca, 'XGrid', 'on');
else
    set(gca, 'XGrid', 'off');
end

% --- Executes on button press in checkbox2.
function checkbox2_Callback(hObject, eventdata, handles)
if get(hObject, 'Value')
    set(gca, 'YGrid', 'on');
else
    set(gca, 'YGrid', 'off');
end

% -----
function fmenu2_Callback(hObject, eventdata, handles)
h=get(gca, 'Children');
ydata=get(h, 'YData');
xdata=get(h, 'XData');
save('ydat.txt', 'ydata', '-ASCII');
save('xdat.txt', 'xdata', '-ASCII');
% -----

function fmenu3_Callback(hObject, eventdata, handles)
figure;
xd=load('xdat.txt', '-ascii');
yd=load('ydat.txt', '-ascii');
plot(xd,yd);
grid on

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)

```

```

cla;
set(hObject,'Enable','off');
xlabel(''); ylabel(''); title('');

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
Num = get(hObject,'Value');
h=get(gca,'Children');
switch Num
    case 1
        set(h,'Color','b');
    case 2
        set(h,'Color','r');
    case 3
        set(h,'Color','g');
end

% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

Вид окна, работающего приложения приведен на рисунках 6 и 7

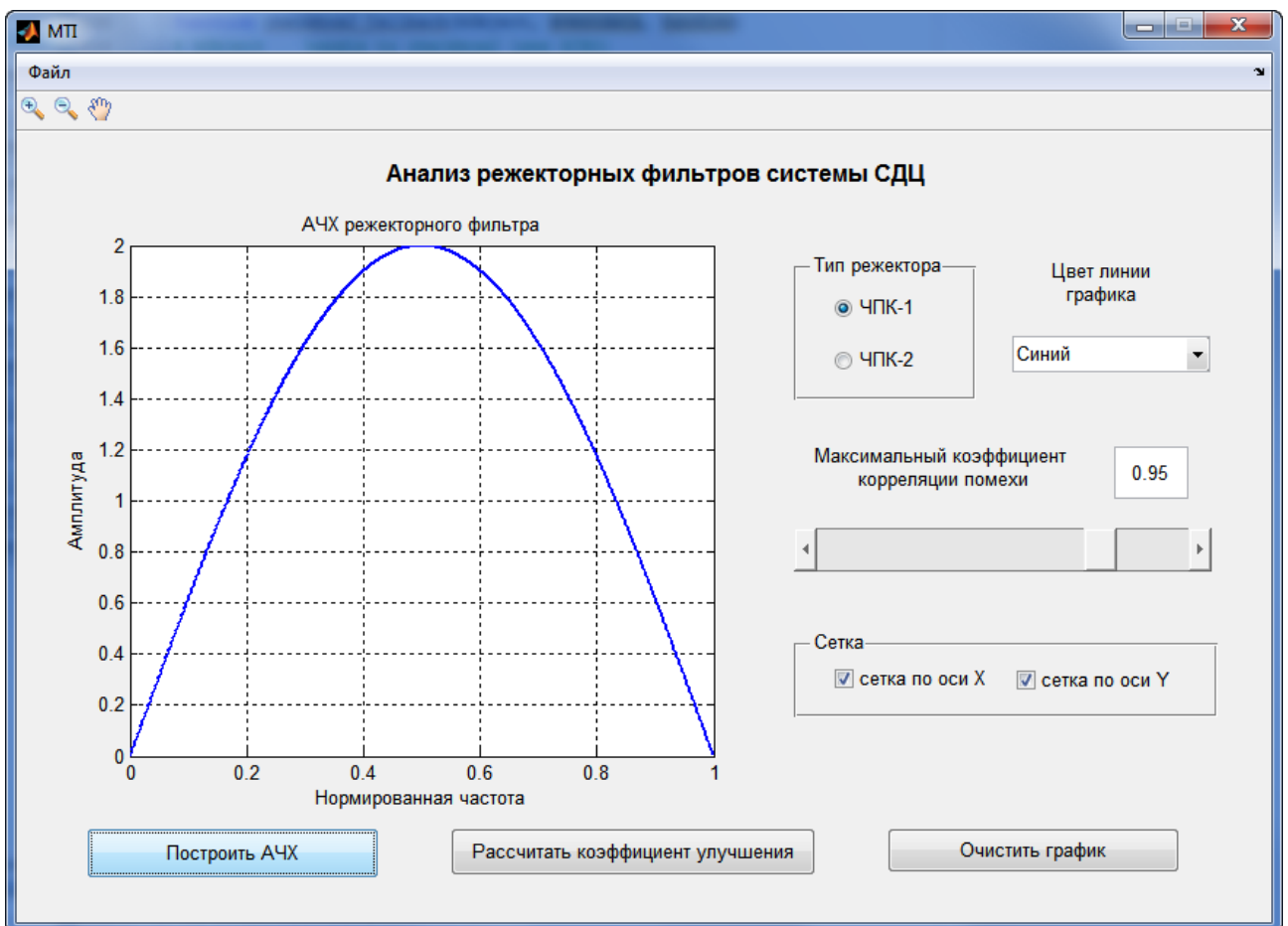


Рисунок 6

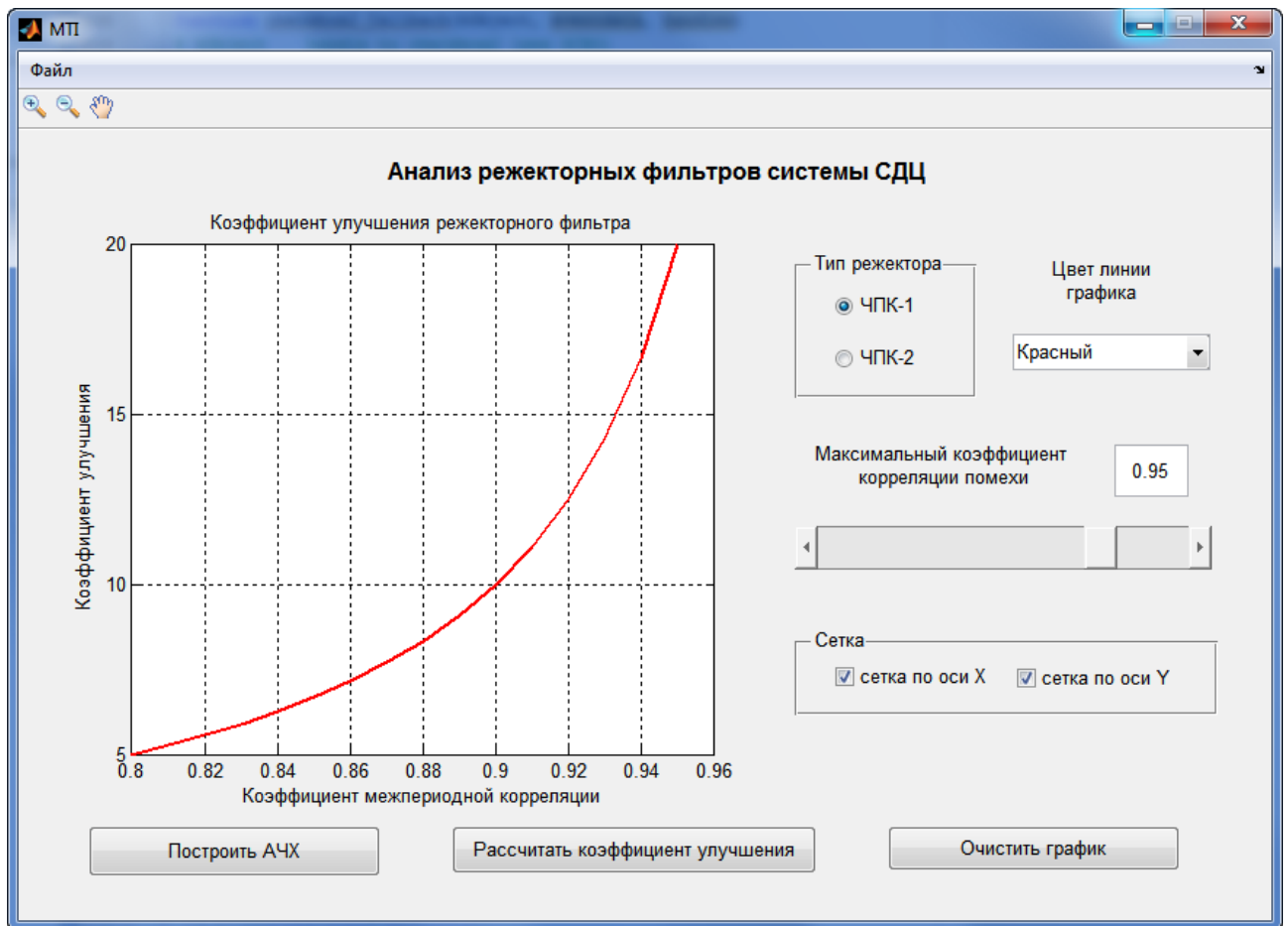


Рисунок 7

График, построенный по сохраненным данным, показан на рисунке 8.

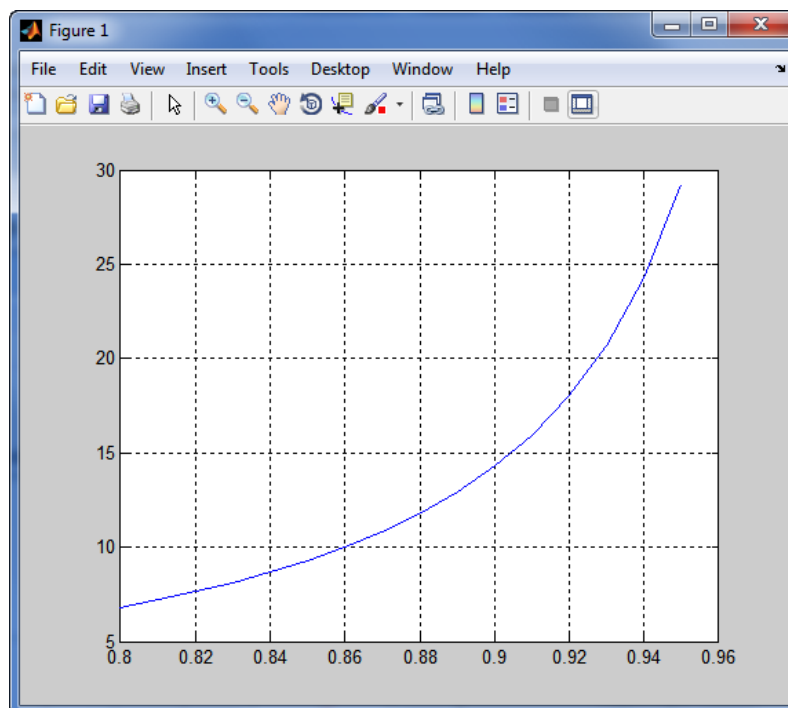


Рисунок 8