

用户界面（或接口）是指：人与机器（或程序）之间交互作用的工具和方法。如键盘、鼠标、跟踪球、话筒都可成为与计算机交换信息的接口。

图形用户界面（Graphical User Interfaces，GUI）则是由窗口、光标、按键、菜单、文字说明等对象（Objects）构成的一个用户界面。用户通过一定的方法（如鼠标或键盘）选择、激活这些图形对象，使计算机产生某种动作或变化，比如实现计算、绘图等。

假如读者所从事的数据分析、解方程、计算结果可视工作比较单一，那么一般不会考虑 GUI 的制作。但是如果读者想向别人提供应用程序，想进行某种技术、方法的演示，想制作一个供反复使用且操作简单的专用工具，那么图形用户界面也许是最好的选择之一。

MATLAB 为表现其基本功能而设计的演示程序 demo 是使用图形界面的最好范例。MATLAB 的用户，在指令窗中运行 demo 打开那图形界面后，只要用鼠标进行选择 and 点击，就可浏览那丰富多彩的内容。

即便比较熟悉 MATLAB 的读者，在他初次编写 GUI 程序时，也会感到棘手。为使读者获得制作自己 GUI 的体验，本章“入门”节提供了一个简单的示例。读者只要输入所提供的程序，就可引出相应的界面。

本章第 2 节叙述图形用户界面的设计原则和一般制作步骤。第 3、4 节分别介绍用户菜单、用户控件的制作。出于“由浅入深”的考虑，前 4 节制作 GUI 是通过 M 脚本文件实现的。利用 M 函数文件制作 GUI，需要解决数据传递问题，为此专设第 5 节给予阐述和示例。MATLAB 5.x 版为方便用户制作图形界面，提供了一个交互式的设计工具 guide。关于该工具的使用方法，被放在第 6 节中，以一个综合例题为目标逐步展

在此提醒读者，假如要比较准确的理解本程序和掌握本章内容，请先阅读第 10 章关于图柄的内容。

11.1 入门

【*例 11.1-1】对于传递函数为 $\frac{1}{s^2+2s+1}$ 的归一化二阶系统，制作一个能绘制该系统单位阶跃响应的图形用户界面。本例演示：（A）图形界面的大致生成过程；（B）静态文本和编辑框的生成；（C）坐标方格控制键的形成；（D）如何使用该界面。

（1）产生图形窗和轴位框：

```
clf reset
H=axes('unit','normalized','position',[0,0,1,1],'visible','off');
set(gcf,'currentaxes',H);
str='\fontname{隶书}归一化二阶系统的阶跃响应曲线';
text(0.12,0.93,str,'fontsize',13);
h_fig=get(H,'parent');
set(h_fig,'unit','normalized','position',[0.1,0.2,0.7,0.4]);
h_axes=axes('parent',h_fig,...
    'unit','normalized','position',[0.1,0.15,0.55,0.7],...
    'xlim',[0 15],'ylim',[0 1.8],'fontsize',8);
```

图 11.1-1 产生坐标轴

（2）在坐标框右侧生成作解释用的“静态文本”和可接受输入的“编辑框”：

```
h_text=uicontrol(h_fig,'style','text',...
    'unit','normalized','position',[0.67,0.73,0.25,0.14],...
```

```

        'horizontal','left','string',{'输入阻尼比系数','zeta ='});
h_edit=uicontrol(h_fig,'style','edit',...
    'unit','normalized','position',[0.67,0.59,0.25,0.14],...
    'horizontal','left',...
    'callback',[...
        'z=str2num(get(gcbo,''string'))';',...
        't=0:0.1:15;','...',...
        'for k=1:length(z);','...',...
        's2=tf(1,[1 2*z(k) 1]);','...',...
        'y(:,k)=step(s2,t);','...',...
        'plot(t,y(:,k));','...',...
        'if (length(z)>1),hold on,end;','...',...
        'end;','...',...
        'hold off,']]);

```

图 11.1-2 在图形界面中添加编辑框和文本框

(3) 形成坐标方格控制按键:

```

h_push1=uicontrol(h_fig,'style','push',...
    'unit','normalized','position',[0.67,0.37,0.12,0.15],...
    'string','grid on','callback','grid on');
h_push2=uicontrol(h_fig,'style','push',...
    'unit','normalized','position',[0.67,0.15,0.12,0.15],...
    'string','grid off','callback','grid off');

```

图 11.1-3 添加了两个按键的图形界面

(4) 输入阻尼比系数，可得单位阶跃响应曲线:

图 11.1-4 输入标量阻尼比所得到的响应曲线

图 11.1-5 输入阻尼比数组所得到的一组响应曲线

11.2 图形用户界面的设计原则和一般步骤

11.2.1 设计原则

11.2.2 一般制作步骤

11.3 界面菜单 (uimenu)

11.3.1 图形窗的标准菜单

【例 11.3.1-1】 本例说明：如何隐藏和恢复标准菜单的显示。

(1) 获得缺省设置的标准菜单

figure

(2) 隐去标准菜单的两种方法

```
set(H_fig, 'MenuBar','none');  
set(gcf,'menubar',menubar);
```

(3) 恢复图形窗上标准菜单

```
set(gcf,'menubar','figure');
```

图 11.3.1-1 含有菜单条的图形窗 图 11.3.1-2 移去菜单条的图形窗

11.3.2 自制的用户菜单

【*例 11.3.2-1】本例演示：如何自制一个带下拉菜单表的 用户菜单（如图 11.3.2-1 所示）。该菜单能使图形窗背景颜色设置为兰色或红色。

```
figure %创建一个图形窗  
h_menu=uimenu(gcf,'label','Color'); %制作用户顶层菜单项 Color <2>  
h_submenu1=uimenu(h_menu,'label','Blue',... %制作下拉菜单项 Blue <3>  
'callback','set(gcf,''Color'',''blue'')'); %<4>  
h_submenu2=uimenu(h_menu,'label','Red',... %制作下拉菜单 Red <5>  
'callback','set(gcf,''Color'',''red'')'); %<6>
```

图 11.3.2-1 创建用户菜单示例

博客石油——精彩转载
更多精彩请登陆
www.blogoil.com

11.3.3 用户菜单的属性

11.3.3.1 回调属性和菜单名

【*例 11.3.3.1-1】本例的目标是：在图形窗上自制一个名为【Test】的“顶层菜单项”；当用鼠标点动该菜单项时，将产生一个带分格的封闭坐标轴。通过本例说明：（A）回调属性的运作机理；（B）用户顶层菜单项的制作（C）uimenu 属性的设置方法；（D）复杂字符串的构成方法和注意事项。

（1）在 MATLAB 指令窗中运行以下程序可产生带分格的封闭坐标轴（见图 11.3.3.1-1）

```
grid on, set(gca,'box','on')
```

图 11.3.3.1-1 带分格的封闭坐标轴

（2）在 MATLAB 指令窗中用以下 eval 指令可产生与图 11.3.3.1-1 相同的界面

```
eval('grid on, set(gca,''box'',''on'')')
```

（3）产生图 11.3.3.1-2 界面的 uimenu 的书写格式一：直接连续表示法

```
uimenu('Label','Test','Callback','grid on, set(gca,''box'',''on'')',')
```

图 11.3.3.1-2 通过顶层菜单 Test 形成的带分格的封闭坐标轴

(4) 产生图 11.3.3.1-2 界面的 uimenu 的书写格式二：方括号续行号表示法

```
uimenu('Label','Test', ...  
    'Callback',['grid on', ...  
        'set(gca, 'box', 'on')'];])
```

(5) 产生图 11.3.3.1-2 界面的 uimenu 的书写格式三：串变量法

```
Lpv='Test';  
Cpv=['grid on', 'set(gca, 'box', 'on')'];  
uimenu('Label', Lpv, 'Callback', Cpv)
```

(6) 产生图 11.3.3.1-2 界面的 uimenu 的书写格式四：构架表示法

```
PS.Label='Test';  
PS.Callback=['grid on;', 'set(gca, 'box', 'on')'];  
uimenu(PS)
```

11.3.3.2 设置简捷键或快捷键

【*例 11.3.3.2-1】本例目标：使图 11.3.2-1 所示菜单成为图 11.3.3.2-1 那样，Color 菜单项及其下拉的 Blue 菜单各带一个简捷键，而另一项下拉菜单 Red 带一个快捷键。

```
[exm11332_1.m]  
figure  
h_menu=uimenu(gcf, 'Label', '&Color'); %带简捷键 C 的用户菜单 Color <2>  
h_submenu1=uimenu(h_menu, 'Label', '&Blue', ... %带简捷键 B 的的下拉菜单 Blue <3>  
    'Callback', 'set(gcf, 'color', 'blue')');  
h_submenu2=uimenu(h_menu, 'label', 'Red', ... %制作另一个下拉菜单 Red  
    'Callback', 'set(gcf, 'color', 'red')', ...  
    'Accelerator', 'r'); %为 Red 菜单设置快捷键 R <7>
```

图 11.3.3.2-1 为用户菜单设置快捷键



11.3.3.3 用户菜单的外观设计

【*例 11.3.3.3-1】本例演示：（A）把用户菜单 'Option' 设置为顶层的第 3 菜单项；（B）下拉菜单被两条分隔线分为三个菜单区；（C）最下菜单项又有两个子菜单组成。

（1）编写程序，生成如图 11.3.3.3-1 所示界面

```
[exm11333_1.m]  
figure  
h_menu=uimenu('label', 'Option', 'Position', 3);  
h_sub1=uimenu(h_menu, 'label', 'grid on', 'callback', 'grid on');  
h_sub2=uimenu(h_menu, 'label', 'grid off', 'callback', 'grid on');  
h_sub3=uimenu(h_menu, 'label', 'box on', 'callback', 'box on', ...
```

```

'separator','on'); %<6>
h_sub4=uimenu(h_menu,'label','box off','callback','box off');
h_sub5=uimenu(h_menu,'label','Figure Color','Separator','on'); %<8>
h_subsub1=uimenu(h_sub5,'label','Red','ForegroundColor','r',... %<9>
    'callback','set(gcf,''Color'',''r'')');
h_subsub2=uimenu(h_sub5,'label','Reset',...
    'callback','set(gcf,''Color'',''w'')');

```

图 11.3.3.3-1

(2) 位置属性的获取

```

Pos_0=get(h_menu,'position'), %查询 Option 菜单位置值
Pos_Box0n=get(h_sub3,'position') %查询 box ob 子菜单位置值
Pos_Red=get(h_subsub1,'position') %查询 red 子菜单的位置值
Pos_0 =
    3
Pos_Box0n =
    3
Pos_Red =
    1

```

[博客石油--精彩转载](#)
[更多精彩请登陆](#)
www.blogoil.com

【*例 11.3.3.3-2】本例演示：当某菜单项选中后，如何使该菜单项贴上检录符“√”。

```

[exml1333_2.m]
figure
h_menu=uimenu('label','Option');
h_sub1=uimenu(h_menu,'label','Grid on',... %<3>
    'callback',[...
        'grid on',...
        'set(h_sub1,''checked'',''on''),'...',...
        'set(h_sub2,''checked'',''off''),'...',...
    ]);
h_sub2=uimenu(h_menu,'label','Grid off',... %<4>
    'callback',[...
        'grid off',...
        'set(h_sub2,''checked'',''on''),'...',...
        'set(h_sub1,''checked'',''off''),'...',...
    ]);

```

图 11.3.3.3-2 Grid on 菜单选中后出现检录符

11.3.3.4 使能 (Enable) 与可见性 (Visible) 属性

【*例 11.3.3.4-1】 本例目标：制作一个带四个子菜单项的顶层菜单项；该下拉菜单分为两个功能区；每个功能区的两个菜单项是相互对立的，因此采用使能属性处理；当图形窗坐标轴消隐时，整个坐标分隔控制功能区不可见。

(1) 编写如下脚本 M 文件 exm11334_1.m

```
[exm11334_1.m]
clf
h_menu=uimenu('label','Option'); %产生顶层菜单项 Option
h_sub1=uimenu(h_menu,'label','Axis on'); %产生 Axis on 菜单项，由缺省设置而使能
h_sub2=uimenu(h_menu,'label','Axis off',...
    'enable','off'); %产生 Axis off 菜单项，但失能
h_sub3=uimenu(h_menu,'label','Grid on',...
    'separator','on','visible','off'); %产生与上分隔的 Grid on 菜单项，但不可见
h_sub4=uimenu(h_menu,'label','Grid off',...
    'visible','off'); %产生 Grid off 菜单项，但不可见
set(h_sub1,'callback',[... %选中 Axis on 菜单项后，产生回调操作
    'Axis on',... %画坐标
    'set(h_sub1,''enable'',''off''),'...',... %Axis on 菜单项失能
    'set(h_sub2,''enable'',''on''),'...',... %Axis off 菜单项使能
    'set(h_sub3,''visible'',''on''),'...',... %Grid on 菜单项可见
    'set(h_sub4,''visible'',''on''),'']); %Grid off 菜单项可见
set(h_sub2,'callback',[... %选中 Axis off 菜单项后，产生回调操作
    'axis off',... %使坐标消失
    'set(h_sub1,''enable'',''on''),'...',... %Axis on 菜单项使能
    'set(h_sub2,''enable'',''off''),'...',... %Axis off 菜单项失能
    'set(h_sub3,''visible'',''off''),'...',... %Grid on 菜单项不可见
    'set(h_sub4,''visible'',''off''),'']); %Grid off 菜单项不可见
set(h_sub3,'callback',[... %选中 Grid on 菜单项后，产生回调
    'grid on',... %画坐标分格线
    'set(h_sub3,''enable'',''off''),'...',... %Grid on 菜单项失能
    'set(h_sub4,''enable'',''on''),'']); %Grid off 菜单项使能
set(h_sub4,'callback',[... %选中 Grid off 菜单项，产生回调
    'grid off',... %消除坐标分格线
    'set(h_sub3,''enable'',''on''),'...',... %Grid on 菜单项使能
    'set(h_sub4,''enable'',''off''),'']); %Grid off 菜单项失能
```

(2) 在 MATLAB 指令窗中运行 exm11334_1，得到图 11.3.3.4-1 所示的界面

图 11.3.3.4-1

(3) 选中【Option】菜单项，界面呈现如图 11.3.3.4-2 所示。

图 11.3.3.4-2

(4) 选中【Option: Axis on】后，界面呈现如图 11.3.3.4-3 所示。

图 11.3.3.4-3

(5) 选中【Option: Grid on】后，界面呈现如图 11.3.3.4-4 所示。

图 11.3.3.4-4

11.3.4 现场菜单的制作

【*例 11.3.4-1】目标：绘制一条 Sa 曲线，创建一个与之相联系的现场菜单，用以控制 Sa 曲线的颜色。

(1) 编写脚本 M 文件 exml134_1.m

```
[exml134_1.m]
t=(-3*pi:pi/50:3*pi)+eps;
y=sin(t)./t;
hline=plot(t,y);          %绘制 Sa 曲线
cm=uicontextmenu;          %创建现场菜单
%制作具体菜单项，定义相应的回调
uimenu(cm,'label','Red','callback','set(hline,''color'',''r''),'')
uimenu(cm,'label','Blue','callback','set(hline,''color'',''b''),'')
uimenu(cm,'label','Green','callback','set(hline,''color'',''g''),'')
set(hline,'uicontextmenu',cm) %使 cm 现场菜单与 Sa 曲线相联系
```

(2) 在指令窗中运行文件 exml134_1.m，得到图 11.3.4-1 所示的（但为蓝色的）Sa 曲线。

图 11.3.4-1 Context 菜单

(3) 将鼠标指针指向线条，点击鼠标右键的同时弹出现场菜单，在选中某菜单项（如 Red）后，Sa 曲线就改变（为红）颜色（如图 11.3.4-1 所示）。

11.4 用户控件 (uicontrol)

11.4.1 控件制作函数

11.4.2 用户控件的种类

11.4.3 控件制作示例

11.4.3.1 双位按键、无线电按键、控件区域框示例

【*例 11.4.3.1-1】目标：创建一个界面包含 4 种控件：静态文本、“无线电”选择开关、双位按键、控件区域框。

```
[exml1431_1.m]
clf reset
set(gcf,'menubar','none')
set(gcf,'unit','normalized','position',[0.2,0.2,0.64,0.32]);
set(gcf,'defaultuicontrolunits','normalized') %设置用户缺省控件单位属性值
h_axes=axes('position',[0.05,0.2,0.6,0.6]);
```

```

t=0:pi/50:2*pi;y=sin(t);plot(t,y);
set(h_axes,'xlim',[0,2*pi]);
set(gcf,'defaultuicontrolhorizontal','left');
h=title=title('正弦曲线');
set(gcf,'defaultuicontrolfontsize',12); %设置用户缺省控件字体属性值
uicontrol('style','frame',... %创建用户控件区 <11>
    'position',[0.67,0.55,0.25,0.25]);
uicontrol('style','text',... %创建静态文本框 <13>
    'string','正斜体图名:',...
    'position',[0.68,0.77,0.18,0.1],...
    'horizontal','left');
hr1=uicontrol(gcf,'style','radio',... %创建“无线电”选择按键 <17>
    'string','正体',... %按键功能的文字标识'正体'
    'position',[0.7,0.69,0.15,0.08]); %按键位置
set(hr1,'value',get(hr1,'Max'));%因图名缺省使用正体，所以小圆圈应被点黑 <20>
set(hr1,'callback',[... % <21>
    'set(hr1,''value'',get(hr1,''max''))',... %选中将小圆圈点黑 <22>
    'set(hr2,''value'',get(hr2,''min''))',... %将“互斥”选项点白 <23>
    'set(htitle,''fontangle'',''normal'')',... %使图名字体正体显示
]);
hr2=uicontrol(gcf,'style','radio',... %创建“无线电”选择按键 <26>
    'string','斜体',... %按键功能的文字标识'斜体'
    'position',[0.7,0.58,0.15,0.08],... %按键位置
    'callback',[...
    'set(hr1,''value'',get(hr1,''min''))',... % <30>
    'set(hr2,''value'',get(hr2,''max''))',... % <31>
    'set(htitle,''fontangle'',''italic'')',... %使图名字体斜体显示
]); % <33>
ht=uicontrol(gcf,'style','toggle',... %制作双位按键 <34>
    'string','Grid',...
    'position',[0.67,0.40,0.15,0.12],...
    'callback','grid');

```

图 11.4.3.1-1 静态文本、选择开关、双位按键及控件区域框

11.4.3.2 静态文本框、滑动键、检录框示例

【*例 11.4.3.2-1】目标：制作演示“归一化二阶系统单位阶跃响应”的交互界面。在该界面中，阻尼比可在[0.02,2.02]中连续调节，标志当前阻尼比值；可标志峰值时间和大小；可标志（响应从0到0.95所需的）上升时间。本例涉及以下主要内容：（A）静态文本的创建和实时改写。（B）滑动键的创建；'Max'和'Min'的设置；'value'的设置和获取。（C）检录框的创建；'value'的获取。（D）受多个控件影响的回调操作。

[exml1432_1.m]

clf reset


```

set(gcf,'unit','normalized','position',[0.1,0.2,0.64,0.35]);
set(gcf,'defaultuicontrolunits','normalized');
set(gcf,'defaultuicontrolfontsize',12);
set(gcf,'defaultuicontrolfontname','隶书');
set(gcf,'defaultuicontrolhorizontal','left');
str='归一化二阶系统阶跃响应曲线';
set(gcf,'name',str,'numbertitle','off'); %书写图形窗名
h_axes=axes('position',[0.05,0.2,0.6,0.7]); %定义轴位框位置
set(h_axes,'xlim',[0,15]); %设置时间轴长度
str1='当前阻尼比=';
t=0:0.1:10;z=0.5;y=step(1,[1 2*z 1],t);
hline=plot(t,y);
htext=uicontrol(gcf,'style','text',... %制作静态说明文本框 <14>
    'position',[0.67,0.8,0.33,0.1],...
    'string',[str1,sprintf('%1.4g',z)]);
hslider=uicontrol(gcf,'style','slider',... %创建滑动键 <17>
    'position',[0.67,0.65,0.33,0.1],...
    'max',2.02,'min',0.02,... %设最大阻尼比为2,最小阻尼比为0.02 <19>
    'sliderstep',[0.01,0.05],... %箭头操纵滑动步长1%,游标滑动步长5% <20>
    'value',0.5); %缺省取阻尼比等于0.5 <21>
hcheck1=uicontrol(gcf,'style','checkbox',... %创建峰值检录框 <22>
    'string','最大峰值',...
    'position',[0.67,0.50,0.33,0.11]);
vchk1=get(hcheck1,'value'); %获得峰值检录框的状态值 <25>
hcheck2=uicontrol(gcf,'style','checkbox',... %创建上升时间检录框 <26>
    'string','上升时间(0->0.95)',...
    'position',[0.67,0.35,0.33,0.11]);
vchk2=get(hcheck2,'value'); %获得上升时间检录框的状态值 <29>
set(hslider,'callback',[... %操作滑动键,引起回调 <30>
    'z=get(gcbo,''value'');',... %获得滑动键状态值 <31>
    'callcheck(htext,str1,z,vchk1,vchk2)']); %被回调的函数文件 <32>
set(hcheck1,'callback',[... %操作峰值检录框,引起回调 <33>
    'vchk1=get(gcbo,''value'');',... %获得峰值检录框状态值 <34>
    'callcheck(htext,str1,z,vchk1,vchk2)']); %被回调的函数文件 <35>
set(hcheck2,'callback',[... %操作峰值检录框,引起回调 <36>
    'vchk2=get(gcbo,''value'');',... %获得峰值检录框状态值 <37>
    'callcheck(htext,str1,z,vchk1,vchk2)']); %被回调的函数文件 <38>

[callcheck.m]
function callcheck(htext,str1,z,vchk1,vchk2)
cla;set(htext,'string',[str1,sprintf('%1.4g',z)]); %更新静态文本框内容 <2>
dt=0.1;t=0:dt:15;N=length(t);y=step(1,[1 2*z 1],t);plot(t,y);
if vchk1 %假如峰值框被选中 <4>
    [ym,km]=max(y);

```

```

if km<(N-3)          %假如在设定时间范围内能插值    <6>
    k1=km-3;k2=km+3;k12=k1:k2;tt=t(k12);
    yy=spline(t(k12),y(k12),tt);          %局部样条插值    <8>
    [yym,kkm]=max(yy);          %求更精确的峰值位置
    line(tt(kkm),yym,'marker','.',...    %画峰值点    <10>
        'markeredgecolor','r','markersize',20);
    ystr=['ymax = ',sprintf('%1.4g',yym)];
    tstr=['tmax = ',sprintf('%1.4g',tt(kkm))];
    text(tt(kkm),1.05*yym,{ystr;tstr})
else          %假如在设定时间范围内不能插值    <15>
    text(10,0.4*y(end),{'ymax --> 1';'tmax --> inf'})
end
end
if vchk2          %假如上升时间框被选中    <19>
    k95=min(find(y>0.95));k952=[(k95-1),k95];
    t95=interp1(y(k952),t(k952),0.95);    %线性插值    <21>
    line(t95,0.95,'marker','o','markeredgecolor','k','markersize',6);
    tstr95=['t95 = ',sprintf('%1.4g',t95)];
    text(t95,0.65,tstr95)
end

```

[博客石油--精彩转载](http://www.blogoil.com)
[更多精彩请登陆](http://www.blogoil.com)
www.blogoil.com



11.4.3.3 可编辑框、弹出框、列表框、按键示例

【*例 11.4.3.3-1】目标：制作一个能绘制任意图形的交互界面。它包括：可编辑文本框、弹出框、列表框。本例的关键内容是：如何使编辑框允许输入多行指令。

```

[exml1433_1.m]
clf reset          %    <1>
set(gcf,'unit','normalized','position',[0.1,0.4,0.85,0.35]);%设置图形窗大小
set(gcf,'defaultuicontrolunits','normalized');
set(gcf,'defaultuicontrolfontsize',11);
set(gcf,'defaultuicontrolfontname','隶书');
set(gcf,'defaultuicontrolhorizontal','left');
set(gcf,'menubar','none');          %删除图形窗工具条
str='通过多行指令绘图的交互界面';
set(gcf,'name',str,'numbertitle','off');    %书写图形窗名
h_axes=axes('position',[0.05,0.15,0.45,0.70],'visible','off');%定义轴位框位置
uicontrol(gcf,'Style','text',...    %制作静态文本框
    'position',[0.52,0.87,0.26,0.1],...
    'String','绘图指令输入框');
hedit=uicontrol(gcf,'Style','edit',...    %制作可编辑文本框    <14>
    'position',[0.52,0.05,0.26,0.8],...

```

```

    'Max',2);          %取 2, 使 Max-Min>1, 而允许多行输入 <16>
hpop=uicontrol(gcf,'style','popup',... %制作弹出菜单 <17>
    'position',[0.8,0.73,0.18,0.12],...
    'string','spring|summer|autumn|winter');%设置弹出框中选项名 <19>
hlist=uicontrol(gcf,'Style','list',... %制作列表框 <20>
    'position',[0.8,0.23,0.18,0.37],...
    'string','Grid on|Box on|Hidden off|Axis off',...%设置列表框中选项名 <22>
    'Max',2);          %取 2, 使 Max-Min>1, 而允许多项选择 <23>
hpush=uicontrol(gcf,'Style','push',... %制作与列表框配用的按键 <24>
    'position',[0.8,0.05,0.18,0.15],'string','Apply');
set(hedit,'callback','calledit(hedit,hpop,hlist)');%编辑框输入引起回调 <26>
set(hpop,'callback','calledit(hedit,hpop,hlist)');%弹出框选择引起回调 <27>
set(hpush,'callback','calledit(hedit,hpop,hlist)');%按键引起的回调 <28>

[calledit.m]
function calledit(hedit,hpop,hlist)
ct=get(hedit,'string'); %获得输入的字符串函数 <2>
vpop=get(hpop,'value'); %获得选项的位置标识 <3>
vlist=get(hlist,'value'); %获得选项位置向量 <4>
if ~isempty(ct) %可编辑框输入非空时 <5>
    eval(ct') %运行从编辑文本框送入的指令 <6>
    popstr={'spring','summer','autumn','winter'}; %弹出框色图矩阵 <7>
    liststr={'grid on','box on','hidden off','axis off'};%列表框选项内容 <8>
    invstr={'grid off','box off','hidden on','axis on'};%列表框的逆指令 <9>
    colormap(eval(popstr{vpop})) %采用弹出框所选色图 <10>
    vv=zeros(1,4);vv(vlist)=1;
    for k=1:4
        if vv(k);eval(liststr{k});else eval(invstr{k});end %按列表选项影响图形
    end
end

```

图 11.4.3.3-1

11.5 由 M 函数文件产生用户菜单和控件

11.5.1 利用全局变量编写用户界面函数文件

【*例 11.5.1-1】目标：利用 M 函数文件创建与例 11.4.3.3-1 相同的用户界面。本例演示：如何依靠全局变量传递控件的图柄，从而保证回调动作正确执行。

(1) 编写 M 函数文件 exml151_1.m 和 calledit1.m

```

[exml151_1.m]
function exml151_1( )
global hedit hpop hlist
(这中间是：原 exml1433_1.m 第<1>行到第<25>行的全部指令)
set(hedit,'callback','calledit1'); %编辑框输入引起回调 <26>

```

```
set(hpop,'callback','calledit1');    %弹出框选择引起回调 <27>
set(hpush,'callback','calledit1');    %按键引起的回调 <28>
```

[calledit1.m]

```
function calledit1()
```

```
global hedit hpop hlist
```

(下面续接内容是：原 calledit.m 第 (2) 行以下的全部指令)

(2) 在 MATLAB 指令窗中运行 exml151_1 就可获得题目所要求的图形用户界面。

11.5.2 利用 'UserData' 属性编写用户界面函数文件

【*例 11.5.2-1】目标：利用 M 函数文件创建与例 11.4.3.3-1 相同的用户界面。本例演示：如何依靠图形窗的 'UserData' 属性传送用户控件的图柄，从而保证回调动作正确执行。

(1) 编写 M 函数文件 exml152_1.m 和 calledit2.m

[exml152_1.m]

```
function exml152_1()
```

(这中间是：原 exml1433_1.m 第 (1) 行到第 (25) 行的全部指令)

```
set(hedit,'callback','calledit2');    %编辑框输入引起回调 <26>
```

```
set(hpop,'callback','calledit2');    %弹出框选择引起回调 <27>
```

```
set(hpush,'callback','calledit2');    %按键引起的回调 <28>
```

```
set(gcf,'UserData',[hedit,hpop,hlist])
```

[calledit2.m]

```
function calledit2()
```

```
H=get(gcf,'UserData');
```

```
ct=get(H(1),'string');    %获得输入的字符串函数 <2>
```

```
vpop=get(H(2),'value');    %获得选项的位置标识 <3>
```

```
vlist=get(H(3),'value');    %获得选项位置向量 <4>
```

(下面续接内容是：原 calledit.m 第 (5) 行以下的全部指令)

(2) 在 MATLAB 指令窗中运行 exml152_1 就可获得题目所要求的图形用户界面。



11.5.3 利用递归法编写用户界面函数文件

【*例 11.5.3-1】目标：利用 M 函数文件创建与例 11.4.3.3-1 相同的用户界面。本例演示：如何依靠图形窗 'UserData' 属性在递归调用中传送用户控件的图柄，保证回调动作正确执行。

(1) 编写 M 函数文件 exml153_1.m

[exml153_1.m]

```
function exml153_1(flag)
```

```
if nargin<1;flag='startup';end %允许在无输入宗量形式下调用该函数 <2>
```

```
if ~ischar(flag);error('flag must be character ''startup''.');end
```

```

switch flag          %切换控制          <4>
case 'startup'       %          <5>
clf reset            %          <6>
set(gcf,'unit','normalized','position',[0.1,0.4,0.85,0.35]);
set(gcf,'defaultuicontrolunits','normalized');
set(gcf,'defaultuicontrolfontsize',11);
set(gcf,'defaultuicontrolfontname','隶书');
set(gcf,'defaultuicontrolhorizontal','left');
set(gcf,'menubar','none');          %删除图形窗工具条
str='通过多行指令绘图的交互界面';
set(gcf,'name',str,'numbertitle','off');    %书写图形窗名
h_axes=axes('position',[0.05,0.15,0.45,0.70],'visible','off');
uicontrol(gcf,'Style','text',...          %制作静态文本框
    'position',[0.52,0.87,0.26,0.1],...
    'String','绘图指令输入框');
hedit=uicontrol(gcf,'Style','edit',...      %制作可编辑文本框  <19>
    'position',[0.52,0.05,0.26,0.8],...      %  <20>
    'Max',2);          %取 2, 使 Max-Min>1, 而允许多行输入  <21>
hpop=uicontrol(gcf,'style','popup',...      %制作弹出菜单  <22>
    'position',[0.8,0.73,0.18,0.12],...      %  <23>
    'string','spring|summer|autumn|winter');%设置弹出框中选项名  <24>
hlist=uicontrol(gcf,'Style','list',...      %制作列表框  <25>
    'position',[0.8,0.23,0.18,0.37],...      %  <26>
    'string','Grid on|Box on|Hidden off|Axis off',...%设置列表框中选项名 <27>
    'Max',2);          %取 2, 使 Max-Min>1, 而允许多项选择  <28>
hpush=uicontrol(gcf,'Style','push',...      %制作与列表框配用的按键  <29>
    'position',[0.8,0.05,0.18,0.15],'string','Apply');
set(hedit,'callback','exml153_1(''set'')'); %编辑框输入引起回调 <31>
set(hpop,'callback','exml153_1(''set'')'); %弹出框选择引起回调 <32>
set(hpush,'callback','exml153_1(''set'')'); %按键引起的回调 <33>
set(gcf,'UserData',[hedit,hpop,hlist]); %向'UserData' 存放图柄  <34>
case 'set'          %以下是回调函数          <35>
H=get(gcf,'UserData'); %从'UserData' 获取图柄          <36>
ct=get(H(1),'string'); %获得输入的字符串函数          <37>
vpop=get(H(2),'value'); %获得选项的位置标识          <38>
vlist=get(H(3),'value'); %获得选项位置向量          <39>
if ~isempty(ct)
    eval(ct')          %运行从编辑文本框送入的指令
    popstr={'spring','summer','autumn','winter'}; %弹出框色图矩阵
    liststr={'grid on','box on','hidden off','axis off'};%列表框选项内容
    invstr={'grid off','box off','hidden on','axis on'};%列表框的逆指令
    colormap(eval(popstr{vpop})) %采用弹出框所选色图
    vv=zeros(1,4);vv(vlist)=1;
    for k=1:4

```

```

        if vv(k);eval(liststr{k});else eval(invstr{k});end %按列表选项影响图形
    end
end
% <50>
end

```

(2) 在 MATLAB 指令窗中运行 exml153_1 就可获得题目所要求的图形用户界面 (即图 11.4.3.3-1 无图形时的初始界面)。

【*例 11.5.3-2】目标：利用 M 函数文件创建与例 11.4.3.3-1 相同的用户界面。本例演示：如何依靠 'Tag' 属性与 findobj 指令的配合使用获取回调操作所必须的控件图柄，保证回调动作正确执行。

本例的程序可由 exml153_1.m 做如下修改而得：

(1) 删去 exml153_1.m 的指令<34><36>。

(2) 在 exml153_1.m 的<20>和<21>行之间增添一行

```
'Tag','H_edit',...
```

(3) 在 exml153_1.m 的<23>和<24>行之间增添一行

```
'Tag','H_popup',...
```

(4) 在 exml153_1.m 的<27>和<28>行之间增添一行

```
'Tag','H_list',...
```

(5) 把 exml153_1.m 的<31><32><33>条指令中的 exml153_1 改为 exml153_2。

(6) 在 exml153_1.m 的<35>和<37>行之间增添以下三条指令。

```

H(1)=findobj(gcf,'Tag','H_edit');
H(2)=findobj(gcf,'Tag','H_popup');
H(3)=findobj(gcf,'Tag','H_list');

```

(7) 把 exml153_1.m 的函数头修改为

```
function exml153_2(flag)
```

(8) 把修改后的文件“另存为” exml153_2.m，就完成了新文件的编写。

(9) 在 MATLAB 指令窗中运行 exml153_2 就可获得题目所要求的图形用户界面。

11.6 图形用户界面设计工具

图 11.6-1

11.6.1 界面设计工具的结构和调用指令

11.6.1.1 界面设计工具的结构

11.6.1.2 图形窗的激活态和受控态

11.6.1.3 启动交互式编辑工具的指令

11.6.2 交互式用户界面设计工具应用示例

图 11.6.2-1 待制作的用户界面

11.6.2.1 工序一：窗口初始位置和大小设计

【例 11.6.2.1-1】本例演示：界面设计工具 guide 的启动和用户界面窗口初试几何制作。

图 11.6.2.1-1 属性编辑工具界面

11.6.2.2 工序二：对象的几何布局

【例 11.6.2.2-1】整个用户界面的几何布局：“轴”、控件种类、相对位置及大致尺寸。本例演示：(A) 设计工具控制面板上“新对象模块区”的图标的使用；(B) 几何布局时不必太多考虑各对象的精细位置和

大小。

图 11.6.2.2-1 “第二道工序” 构成控件布局的界面

11.6.2.3 工序三：新建对象的属性设置

【例 11.6.2.3-1】控件关键属性的设置。本例演示：（A）属性编辑工具的使用。（B）当 ‘Callback’ 属性值可用比较简单的 MATLAB 语句表达时，则直接填写；如果语句较多，表达复杂，那么就应采用一个待写的 M 函数名填写。本例中的回调都借助 M 函数文件实现。（C）当控件上有字符串标识时，应注意文字的对齐方式和注意字体大小，使外观上与对象大小协调。（D）控件的 ‘String’ 属性字符串的输入格式。在这过程中，可能还要适当调整对象几何尺寸，使字符表现清晰醒目。（E） ‘Units’ 采用 ‘normalized’，使得所有新建对象随所在图形窗按比例缩放。

11.6.2.4 工序四：用户菜单的制作

用户菜单制作工序比较独立，因此该工序可前可后，也可以与 “工序一” 相合并。

【例 11.6.2.4-1】用户菜单的制作。本例演示：菜单编辑工具的使用。在本例中菜单引起的回调都是直接、简单的 MATLAB 语句。

图 11.6.2.4-1 用户菜单编辑器界面

11.6.2.5 工序五：新建图形对象的齐整化

【例 11.6.2.5-1】控件的齐整化。本例演示：演示 “对齐编辑工具” 的使用。

图 11.6.2.5-1

11.6.2.6 工序六：回调函数的编写

【例 11.6.2.6-1】回调函数的编写。本例演示：从处理方便出发编写回调函数。（关于回调函数的详细讨论，请看第 11.5 节的三个算例）

弹出框的回调函数 Mycolormap.m

```
[Mycolormap.m]
function Mycolormap
popstr={' spring',' summer',' autumn',' winter'}; %弹出框色图矩阵
vpop=get(findobj(gcf,' Tag',' PopupMenu1'),' value'); %获得选项的位置标识
colormap(eval (popstr{vpop})) %采用弹出框所选色图
```

（2）列表框和 “Apply” 按钮配合的回调函数 Myapply.m

```
[Myapply.m]
function Myapply
vlist=get(findobj(gcf,' Tag',' Listbox1'),' value'); %获得选项位置向量
liststr={' grid on',' box on',' hidden off',' axis off'}; %列表框选项内容
invstr={' grid off',' box off',' hidden on',' axis on'}; %列表框的逆指令
vv=zeros(1,4);vv(vlist)=1;
for k=1:4
    if vv(k);eval(liststr{k});else eval(invstr{k});end %按列表选项影响图形
end
```

（3）动态编辑框的回调函数 Myedit.m

```
[Myedit.m]
function Myedit
```

```
ct=get(findobj(gcf,'Tag','EditText1'),'string');
eval(ct')
```

11.6.2.7 工序七：界面功能的全面测试

11.6.2.8 为读者提供的配套文件和数据

(1) 机器自动生成的主控文件

```
[Myguil.m]
function fig = Myguil()
% This is the machine-generated representation of a Handle Graphics object
% and its children. Note that handle values may change when these objects
% are re-created. This may cause problems with any callbacks written to
% depend on the value of the handle at the time the object was saved.
% This problem is solved by saving the output as a FIG-file.%
% To reopen this object, just type the name of the M-file at the MATLAB
% prompt. The M-file and its associated MAT-file must be on your path.%
% NOTE: certain newer features in MATLAB may not have been saved in this
% M-file due to limitations of this format, which has been superseded by
% FIG-files. Figures which have been annotated using the plot editor tools
% are incompatible with the M-file/MAT-file format, and should be saved as
% FIG-files.
load Myguil
h0 = figure('Units','normalized', ...
'Color',[0.8553876799929505 0.8553876799929505 0.8553876799929505], ...
'Colormap',mat0, ...
'FileName','F:\99\m5\Myguil.m', ...
'MenuBar','none', ...
'Name','GUI 工具设计的界面', ...
'PaperPosition',[18 180 575.9999999999999 432], ...
'PaperUnits','points', ...
'Position',[0.1484375 0.5291666666666667 0.8 0.35], ...
'Tag','Fig1', ...
'ToolBar','none');
h1 = uimenu('Parent',h0, ...
'Label','Zoom', ...
'Tag','M_Z');
h2 = uimenu('Parent',h1, ...
'Callback','zoom on', ...
'Label','On', ...
'Tag','M_Zon');
h2 = uimenu('Parent',h1, ...
'Callback','zoom off', ...
'Label','Off', ...
'Tag','M_Zoff');
h1 = uicontrol('Parent',h0, ...
```



```

'Units','normalized', ...
'BackgroundColor',[0.7529411764706 0.7529411764706 0.7529411764706], ...
'Callback','Mycolormap', ...
'ListboxTop',0, ...
'Position',[0.80859375 0.773809523809524 0.16 0.130952380952381], ...
'String',['spring ','summer ','autumn ','winter '], ...
'Style','popupmenu', ...
'Tag','PopupMenu1', ...
'value',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Max',2, ...
'Position',[0.80859375 0.327380952380952 0.16 0.398809523809524], ...
'String',['grid on ','box on ','hidden off','axis off '], ...
'Style','listbox', ...
'Tag','Listbox1', ...
'value',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'BackgroundColor',[.752941176470588 .752941176470588 .752941176470588], ...
'Callback','Myapply', ...
'ListboxTop',0, ...
'Position',[0.80859375 0.12 0.16 0.15], ...
'String','apply', ...
'Tag','Pushbutton1');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Callback','Myedit', ...
'FontName','Times New Roman', ...
'FontSize',10, ...
'HorizontalAlignment','left', ...
'ListboxTop',0, ...
'Max',2, ...
'Position',[.55078125 .3273809523809524 .232421875 .4523809523809523], ...
'Style','edit', ...
'Tag','EditText1');
h1 = axes('Parent',h0, ...
'CameraUpVector',[0 1 0], ...
'CameraUpVectorMode','manual', ...
'Color',[1 1 1], ...
'ColorOrder',matl, ...
'Position',[0.05 0.15 0.45 0.65], ...

```

```

'Tag','Axes1', ...
'XColor',[0 0 0], ...
'YColor',[0 0 0], ...
'ZColor',[0 0 0]);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.4978165938864628 -0.222222222222221 9.160254037844386], ...
'Tag','Axes1Text4', ...
'VerticalAlignment','cap');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[-0.1353711790393013 0.4907407407407408 9.160254037844386], ...
'Rotation',90, ...
'Tag','Axes1Text3', ...
'VerticalAlignment','baseline');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','right', ...
'Position',mat2, ...
'Tag','Axes1Text2', ...
'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.4978165938864628 1.064814814814815 9.160254037844386], ...
'Tag','Axes1Text1', ...
'VerticalAlignment','bottom');
set(get(h2,'Parent'),'Title',h2);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontName','隶书', ...
'FontSize',13, ...
'HorizontalAlignment','left', ...
'ListboxTop',0, ...
'Position',[0.55 0.8 0.2 0.12], ...

```

```

'String','输入绘图指令', ...
'Style','text', ...
'Tag','StaticText1');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'BackgroundColor',[.752941176470588 .752941176470588 .752941176470588], ...
'Callback','close(gcf)', ...
'ListboxTop',0, ...
'Position',[0.586992187499999 0.12 0.16 0.15], ...
'String','Close', ...
'Tag','Pushbutton2');
if nargin > 0, fig = h0; end

```

(2) 配套数据文件

[Myguizzy.m]

function Myguizzy

%假如 Myguil.m 所在目录不是 d:\matbook5\mdisk，那么第<10>条就应做相应的改变。

%一定要保证本函数生成的 Myguil.mat 与 Myguil.m 在同一目录。

mat0=jet(64);

```

mat1=[ 0      0      1.0000
0      0.5000      0
1.0000      0      0
0      0.7500      0.7500
0.7500      0      0.7500
0.7500      0.7500      0
0.2500      0.2500      0.2500];

```

mat2=[-0.1179 1.3056 9.1603];

save d:\matbook5\mdisk\Myguil %<10>

(3) 如何利用本节所提供的文件产生图 11.6.2-1 所示的界面

1 把本节提供的 Myguil.m，Myguizzy.m，Mycolormap.m，Myapply.m，Myedit.m 五个文件放在 MATLAB 的搜索路径上。

1 先运行 Myguizzy.m，创建数据文件 Myguil.mat。

1 运行 Myguil.m，就可得到符合要求的界面。

摘自：精通 Matlab 综合辅导和指南