

Introduction to MATLAB

Violeta Ivanova, Ph.D.

Office for Educational Innovation & Technology

violeta@mit.edu

<http://web.mit.edu/violeta/www>

[Topics]

- MATLAB Interface and Basics
- Calculus, Linear Algebra, ODEs
- Graphics and Visualization
- Basic Programming
- **Programming Practice**
- Statistics and Data Analysis

[Resources]

■ Class materials

<http://web.mit.edu/acmath/matlab/IAP2007>

- Previous sessions: [InterfaceBasics](#),
[Graphics](#), [Programming](#)
- This session: [Practice <.zip, .tar>](#)

■ Mathematical Tools at MIT

<http://web.mit.edu/ist/topics/math>

[MATLAB Help Browser]

■ MATLAB

- + Programming
 - + M-File Programming
 - + Programming Tips
- + Creating Graphical User Interfaces
 - + Getting Started with GUIDE
 - + Creating a GUI
 - + Laying Out GUIs and Setting Properties
 - + Programming GUIs
 - + GUI Applications
 - + Working with Callbacks

MATLAB GUI Programming

GUIDE, FIG-Files & M-Files
Graphical User Interface
Widgets and Callbacks

MATLAB GUI Demo



■ 1.010 Uncertainty in Engineering: MonteCarlo

Daniele Veneziano, Andreas Langousis, Violeta Ivanova

The screenshot shows a MATLAB GUI window titled 'simulationsGUI'. The main title bar is black with white text. Below the title bar, the GUI has a light beige background. It contains several input fields and a 'Run Simulations' button. The inputs are: 'Mean Customer Interarrival Time' (slider set to 5.0 min), 'Mean Service Time Per Customer' (slider set to 10.0 min), 'Number of Open Counters' (spinner set to 5), 'Max. Number of Simulation States' (spinner set to 10000), and 'Accuracy' (spinner set to 0.01). At the bottom, there is a black bar with white text describing three simulation systems: SYSTEM 1: Random Allocation, SYSTEM 2: Shortest Queue, and SYSTEM 3: Single Queue.

simulationsGUI

Monte Carlo Simulations of Three Supermarket Checkout Systems

Mean Customer Interarrival Time: 5.0 min

Mean Service Time Per Customer: 10.0 min

Number of Open Counters: 5

Max. Number of Simulation States: 10000

Accuracy: 0.01

Run Simulations

SYSTEM 1: Random Allocation. Customers select a counter at random with equal probability.
SYSTEM 2: Shortest Queue. Upon arrival at checkout customers select the shortest queue.
SYSTEM 3: Single Queue. The first customer in the queue goes to the first available counter.

[Graphical User Interface]

■ Widgets

- Text fields: static text, edit text
- Buttons: toggle, push, and radio buttons
- Popup menus
- Check boxes
- Sliders
- Axes
- Panels

■ Callbacks

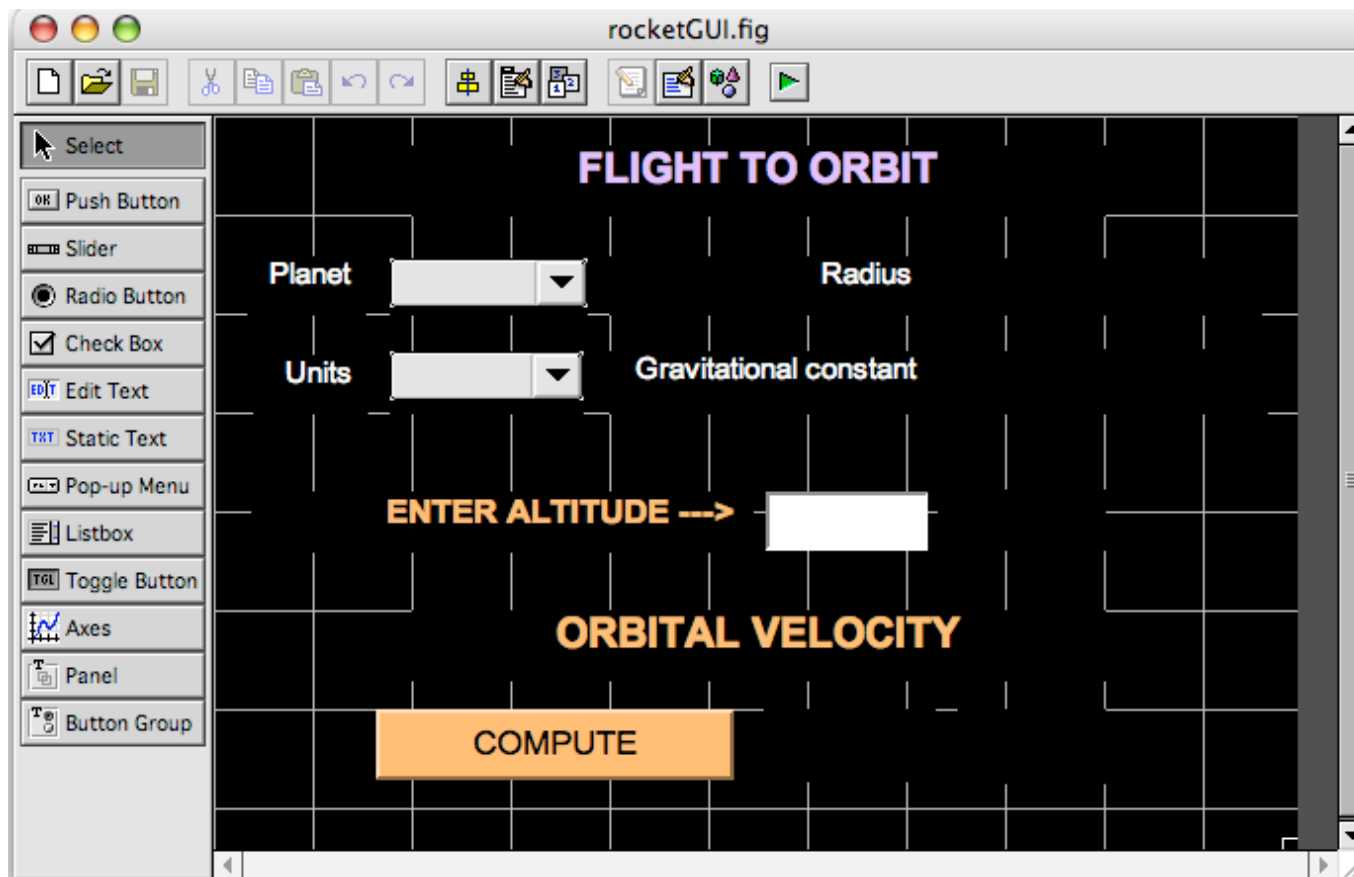
... are functions called by widgets.

[MATLAB Programs with GUI]

- FIG-File(s)
 - Created with **GUIDE**
 - GUI graphical layout (widgets)
- M-Files
 - Created in **Editor/Debugger**
 - The code behind a GUI
 - Function M-file to open the GUI
 - Function M-files with callbacks
 - Other function M-files

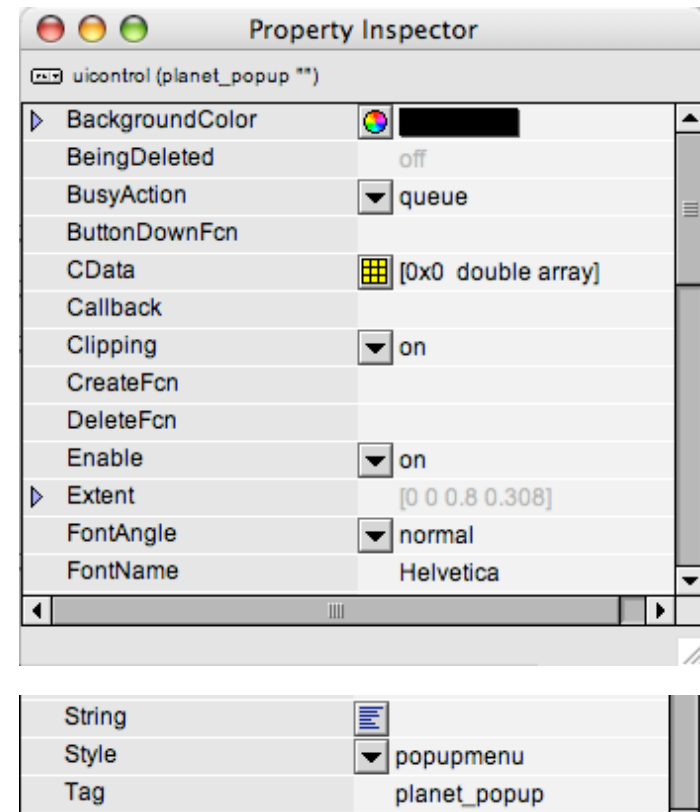
[GUIDE Editor: Layout Design]

```
>> guide GUIfilename
```



[GUIDE: Property Inspector]

- Double click on a widget to open the Property Inspector
- Set widget attributes
 - Tag
 - String
 - Fonts (optional)
 - Colors (optional)
- Do **NOT** set **Callback** function here



[GUI FIG-Files and M-Files]

- In **GUIDE**, save the graphical layout to a **FIG file**
 - Note: you may ignore the generated M-file.
- In **Editor/Debugger**, create M-Files
 - One **M-File** for every **GUI Window**
 - One **M-File** for every **Callback** function that is executed when the user interacts with widgets
 - Note: different widgets may call the same callback function.

[GUI M-Files]

■ Functions:

- To open the GUI: `openfig`, `figure`
- To get access to widgets: `findobj`
- To set attributes to widgets: `set`
- To get attributes of widgets: `get`

■ Widget attributes:

- Widgets work with `STRINGS`, not numbers
- `'String'` and `'Value'`
- `'Tag'`
- `'Callback'`

[GUI M-Files: Example]

```
function rocketprogram
% -----
fig1 = openfig('rocketGUI.fig', 'reuse')
figure(fig1)
% -----
R_units_txt = findobj( fig1, 'Tag', 'R_units_txt' );
units_popup = findobj( fig1, 'Tag', 'units_popup' );
% -----
set( R_units_txt, 'String', 'km' );
set( units_popup, 'String', {'Metric', 'English'} );
set( units_popup, 'Value', 1);
set( units_popup, 'Callback', velocitycallback);
```

[GUI M-Files: Example (continued)]

```
function velocitycallback
% -----
units_popup = findobj( fig1, 'Tag', 'units_popup' );
H_input_txt = findobj( fig1, 'Tag', 'H_input_txt' );
% -----
units = get( units_popup, 'Value' );
strH  = get( H_input_txt, 'String' );
numH  = str2num( strH );
% -----
if units == 1
    V = orbitalvelocity( R, g0, numH, 'm' )
```

[MATLAB Practice Exercises]

- **Exercise One:** `rocketGUI.fig`,
`velocitycallback.m`,
`orbitalvelocity.m` and
`rocketprogram.m`
 - Creating widgets for a GUI with GUIDE
 - Callbacks: writing code behind a GUI

Follow instructions in m-files ...

MATLAB Code Database

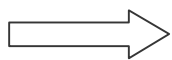
MATLAB Central
User File Exchange
Writing a Program

[MATLAB File Exchange]



An open exchange for the MATLAB and Simulink user community

<http://www.mathworks.com/matlabcentral>



File Exchange

■ Downloads

- `units.m`
- `sun_position.m`
- `sudoku.m`

[MATLAB Practice Exercises]

- **Exercise Two:** `SunBostonNow.m`, `sun_position.m` **and** `units.m`
 - Using M-files from MATLAB Central
 - Writing a program from scratch

Follow instructions in m-files ...